Trademarks

Telebit, NetBlazer, and TrailBlazer are trademarks of Telebit Corporation.

Other brands or product names are trademarks or registered trademarks of their respective holders.

Software Copyright Notice

Printed in U.S.A.

How to Use This Manual 1

This guide provides instructions for configuring the NetBlazer to the operational requirements of the network environment in which the NetBlazer will be used. It is intended as a general guide to the system administrator in performing both initial configuration, network "fine-tuning," and network management activities.

This guide assumes that all NetBlazer hardware has been preconfigured and installed in the NetBlazer chassis, and that all cabling to the network and telephone lines has been completed. This manual consists of eleven sections:

- Chapter 1, Introduction, provides an overview of the NetBlazer followed by a description of the various roles that the NetBlazer plays in a TCP/IP environment. Planning guidelines are also covered in this chapter.
- Chapter 2, Using the NetBlazer Commands, overviews the NetBlazer's command environment and its facilities, reviewing the commands by functional group.
- Chapter 3, Getting Started, covers making a working copy of the NetBlazer System distribution diskette, and performing the initial log-on and setup of the NetBlazer including, setting the time and date, and setting the root password.
- Chapter 4, Local IP Router Configuration, provides instructions on how to configure the NetBlazer to route IP packets between local Ethernet and/or Token Ring networks.
- Chapter 5, Dynamic Interface Configuration, provides instructions on how to configure the NetBlazer to route IP packets over a dynamic dial-up interface.
- Chapter 6, Synchronous Interface Configuration, covers setting up the syn0 synchronous port for use with DSU/CSUs or with an ISDN terminal adapter.
- Chapter 7, Asynchronous Lines Usage, provides information on using the NetBlazer's asynchronous lines including setting up line characteristics, dedicated SLIP or PPP links, and setting up and controlling a raw TCP link.
- Chapter 8, Dailout Destinations, covers setting up the remote destination list, which is used to establish a connection with a remote system.

- Chapter 9, User IDs and Destination Groups, covers setting up the user ID records. These records are required for each user logging into the NetBlazer, be it through a telnet session or by calling in from a remote terminal.
- Chapter 10, IP Packet Filtering, explains the use of the NetBlazer's IP filter commands to restrict IP packet routing to and from designated networks or host addresses through a particular interface.
- Chapter 11, Network Management, describes the NetBlazer's SNMP support facilities and their use, and monitoring and reporting the NetBlazer's activity.
- NetBlazer file definitions and software update procedures are described in the Appendix.

Audience

This document is intended as a procedural guide for the System Administrator in configuring the NetBlazer.

Related Documentation

Hardware installation of the NetBlazer products are detailed in the *NetBlazer Installation Guides*.

Once the NetBlazer has been configured, the system administrator can turn his or her attention to getting the users up to speed with the NetBlazer. The *NetBlazer User's Guide* provides a user-oriented overview of the NetBlazer operation, its user-level commands, and shows step-by-step procedures for performing basic user functions.

A complete listing of the NetBlazer's commands, along with comprehensive descriptions and examples of usage, is contained in the *NetBlazer Command Reference*.

NetBlazer Training

Telebit's Customer Services group offers excellent NetBlazer training on topics such as "Introduction to Internetworking and TCP/IP," and "NetBlazer Technical Training." This comprehensive course combines easy-to-read training materials with insightful lecture and labs. For more information and class schedules, contact Telebit Education and Training Services at 1-800-TELEBIT.

Table of Contents

Introduction 1

This chapter starts with a functional overview of the NetBlazer followed by a description of the various roles that the NetBlazer plays in a TCP/IP environment. It then goes on to explain some of the unique characteristics of the NetBlazer, and ends with some general planning guidelines along with a summary of the information that you will need to configure the NetBlazer.

Functional Overview

The NetBlazer is a special purpose computer primarily designed to route data between TCP/IP network segments and serial lines. Physically, it consists of a CPU motherboard, a high-density diskette drive, a power supply, and several adapter boards, which provide the physical connections to the interconnected LANs and the serial lines.

The NetBlazer's proprietary operating system and control software provides the following functions:

- **Command Interpreter** – provides the user interface to the NetBlazer that allows the users to control the NetBlazer operation.
- **Disk Management** – includes a set of utilities used to manage the files on the NetBlazer's diskette.
- **User Authentication and Security** – prevents unauthorized access to the NetBlazer and its associated facilities.
- **TCP/IP Control and Services** – allows the NetBlazer to participate in a TCP/IP environment.
- **Serial Line Control** – regulates the interaction of the NetBlazer with devices attached to the NetBlazer's serial lines.
- **Network Management and Diagnostics** – provides the facilities used by a system administrator to oversee the operation of the NetBlazer on the network.

The Role of the NetBlazer

The NetBlazer is a multipurpose TCP/IP device that serves a variety of functions in a TCP/IP environment. The primary function of the NetBlazer is to act as a dial-up router for transferring IP packets over the Public

Switched Telephone Network (PSTN). It also can be used as a terminal server that allows remote users access to the network, or as a modem server to allow users on the network to gain access to outside services.

Using the NetBlazer as a Router

As shown in Figure 1-1, the NetBlazer serves as a dial-up router to transfer IP packets between geographically distant sites through the telephone network. It also can serve as a gateway between Ethernet or Token Ring LANs that are connected directly to the NetBlazer.

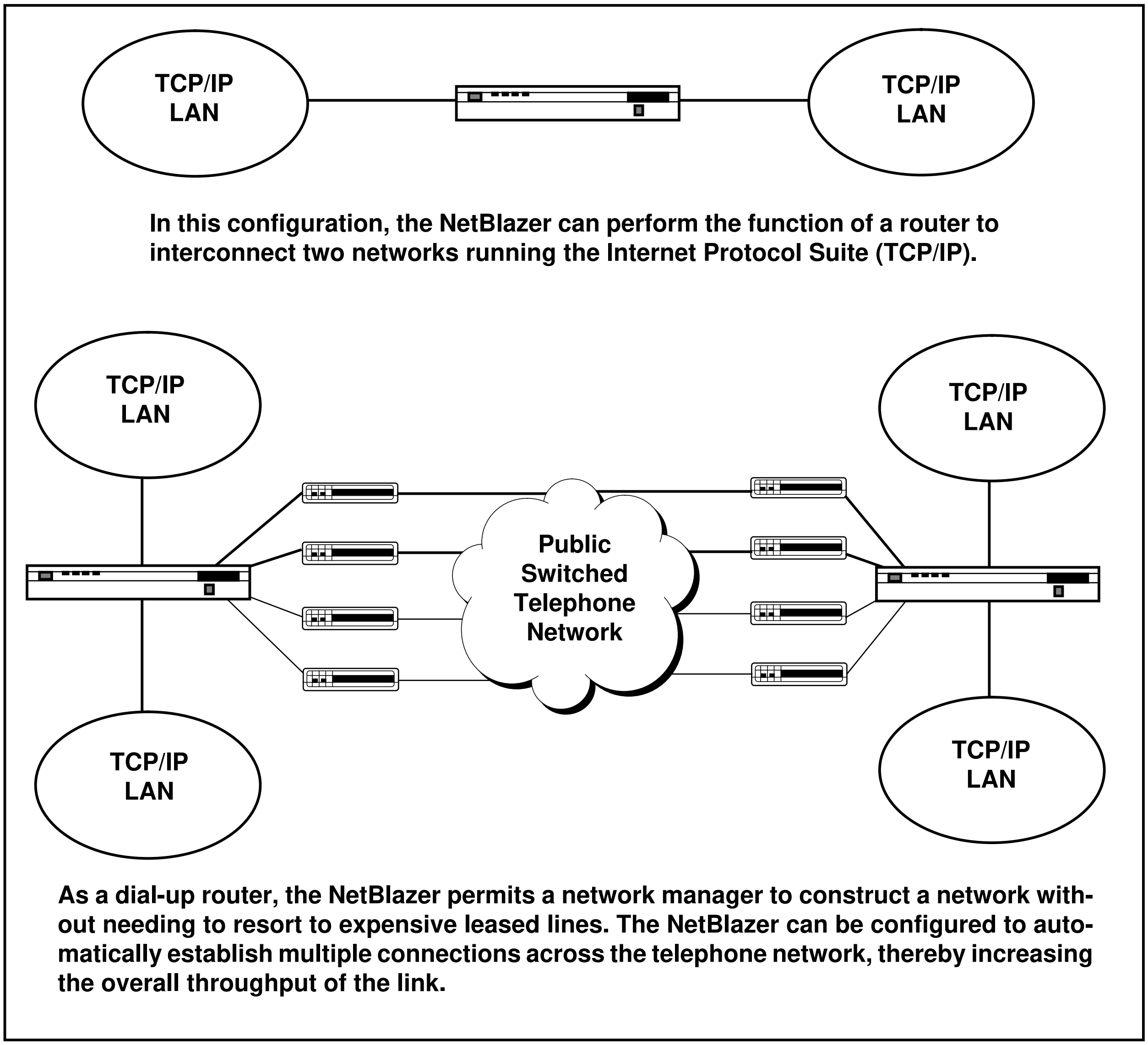


Figure 1-1. Using the NetBlazer as an IP Router

When a packet arrives at the NetBlazer, the NetBlazer determines which interface to use based on the destination IP address. If the interface is one of the Ethernet or Token Ring ports, the NetBlazer encapsulates the IP packet in a physical network frame and sends it to the destination host or to another gateway on the network for further routing.

If the interface is a dynamic dial-out interface, the NetBlazer checks for an existing telephone link to the remote site. If a telephone link does not exist, the local NetBlazer gets the telephone number for the interface from its dial-out record, and calls the remote NetBlazer.

When a modem connection is established, the answering NetBlazer starts a caller authentication process to verify that this is an authorized connection. If the calling NetBlazer responds with a valid user ID, password and crypto responses, the link is established.

When a link exists, the sending NetBlazer encapsulates IP packets in SLIP or PPP frames and sends them to the receiving NetBlazer over the telephone network. The dynamic interface record tells the NetBlazer whether to use SLIP or PPP framing.

Using the NetBlazer as a Terminal Server

As a terminal server, the NetBlazer provides the means for remote users to gain access to the IP networks by calling into the NetBlazer through the PSTN as illustrated in Figure 1-2. The remote user can use an asynchronous terminal (or computer with communications software) and a modem to call into the NetBlazer.

Caller Verification

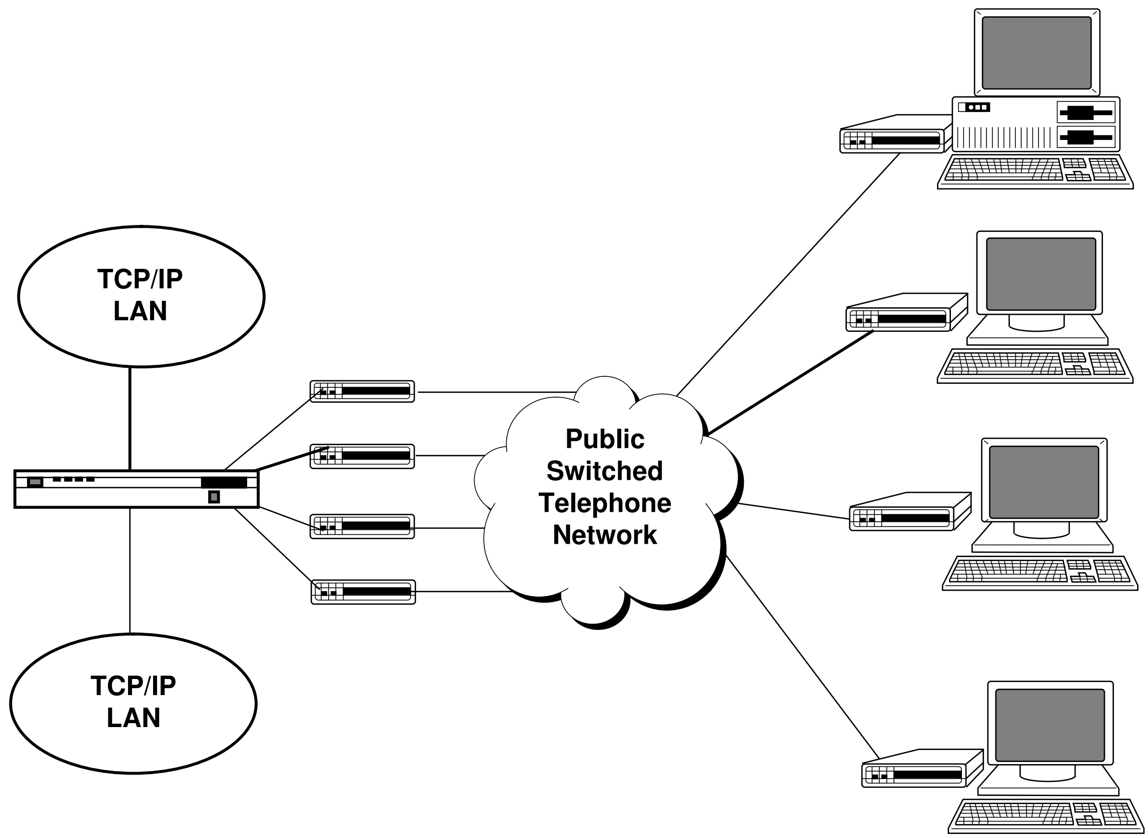
Upon receiving a call from a remote user, the NetBlazer prompts the caller for a login ID and password, which are used by the NetBlazer to verify that the caller is an authorized user. If the caller responds with a valid user ID and password, he or she is connected to the NetBlazer in command mode. The user can then enter a command to start a telnet session with a TCP/IP host on the network.

Once a telnet session is established, the NetBlazer packetizes data sent by the remote user and routes it to the TCP/IP host. Packetized data received from the TCP/IP host is reassembled into a serial character stream before being transmitted to the remote user.

Automatically Connecting to a Host

The NetBlazer can be configured to bypass the login process and start a telnet session with a designated host by assigning a telnet command to an asynchronous line. Then, whenever a call is received on that line, the caller is automatically connected with the specified host as if he or she had called the host directly.

An alternative to the previous method of automatically connecting a caller to a host is to create a special command alias for users who always connect to the same host that automatically starts a telnet or rlogin session with the host when the user logs on to the NetBlazer. The advantage of using this method is that it does not require dedicating an asynchronous line to a single host.



As a terminal server, the NetBlazer provides the means by which users with terminals and modems gain access to the resources of a larger network. This eliminates the need for special hardware on the host computer to manage and control pools of modems for incoming calls.

In addition, a personal computer running compatible networking software can use the NetBlazer as a gateway to the larger network. The PC then has access to all the network services.

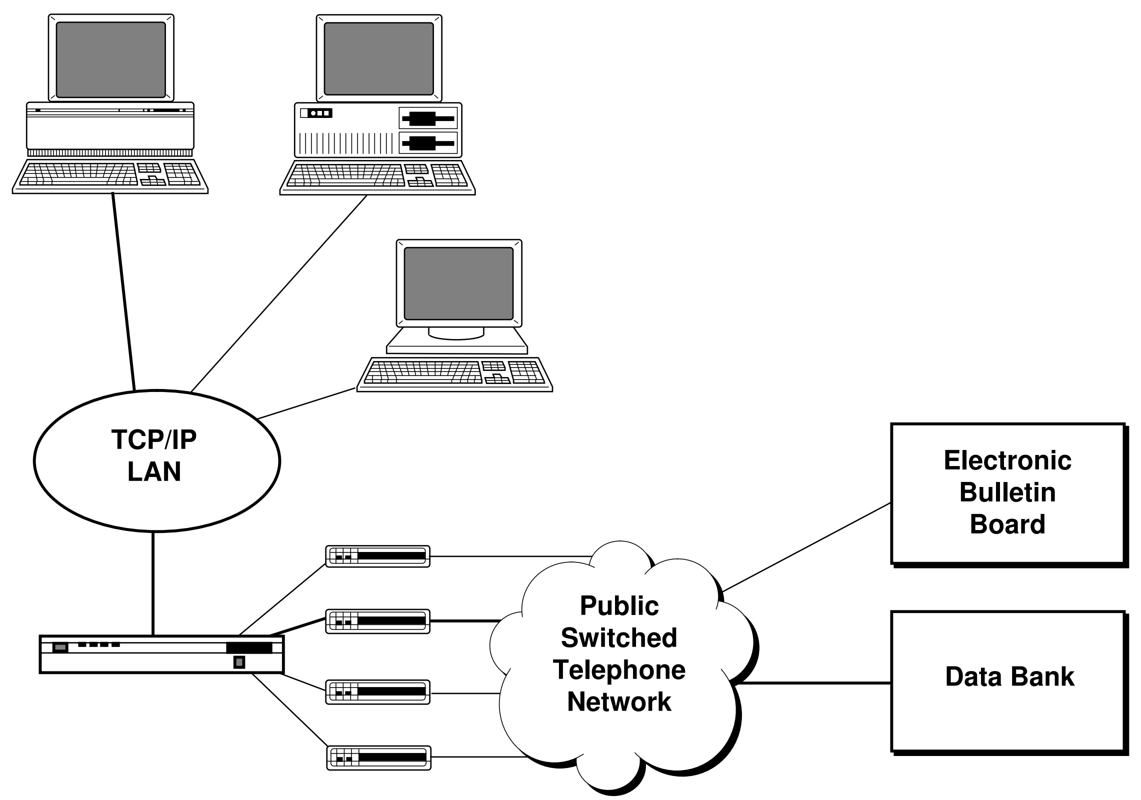
Figure 1-2. Using the NetBlazer as a Terminal Server

Using Callback Security

To provide additional security, you can configure the NetBlazer to call back a remote user at a predetermined destination. When callback security is used, the remote user calls the NetBlazer at the telephone number associated with the callback request line. The NetBlazer then prompts the user for the dialout destination to callback, then disconnects the original connection and calls the user back at the predetermined destination.

Using the NetBlazer as a Modem Server

As illustrated in Figure 1-3, using the NetBlazer as a modem server provides the means for TCP/IP hosts or end-users to access outside services through a modem connected to the NetBlazer.



As a modem server, the NetBlazer provides the means for hosts and end-users to get access to a pool of modems by which they can access services unavailable on the local network. This greatly simplifies cabling, configuration, and modem arrangement.

Figure 1-3. Using the NetBlazer as a Modem Server

To use the NetBlazer as a modem server, a user on an IP network should first establish a telnet session with the NetBlazer. When the telnet session is started, the NetBlazer prompts the user for a login ID and password. After the login is validated, the user can issue a command to the NetBlazer to place a call to any remote location that has been previously defined in the dialout list. Data on the network side of the NetBlazer is packetized, and data on the telephone line side of the NetBlazer is in a serial character stream format.

IP Network Addressing and Routing Technique

This sections describes a couple of techniques used by the NetBlazer to route IP packets between network segments that differ from the routing technique used by most routers.

Using Symbolic Named Interfaces to Route IP Packets

Unlike traditional routers, the NetBlazer uses a symbolic name to specify which interface to use to route the packet. The NetBlazer employs three types of interfaces: static, dynamic, and dedicated SLIP or PPP interfaces.

The Ethernet interfaces (en0, en1, and en2), Token Ring (tr0, tr1, and tr2) interface and the synchronous interface (syn0) are static interfaces. These interfaces are automatically set up when the system is booted.

Dynamic interfaces, used to route IP packets over the switched telephone network, are created by the **ipdial** or **dynamic add** commands. See Chapter 5, Dynamic Interface Configuration for instructions on how to set up a dynamic interface.

Dedicated SLIP or PPP interfaces are used to send IP packets through an RS-232 serial connection. These interfaces are created by attaching an interface name to a serial line as described in Chapter 7, Asynchronous Lines Usage.

The advantage of the NetBlazer's routing technique over that of traditional routers is that the NetBlazer requires fewer IP addresses by eliminating the need for an intermediate network address on a point-to-point serial link as shown in Figure 1-4.

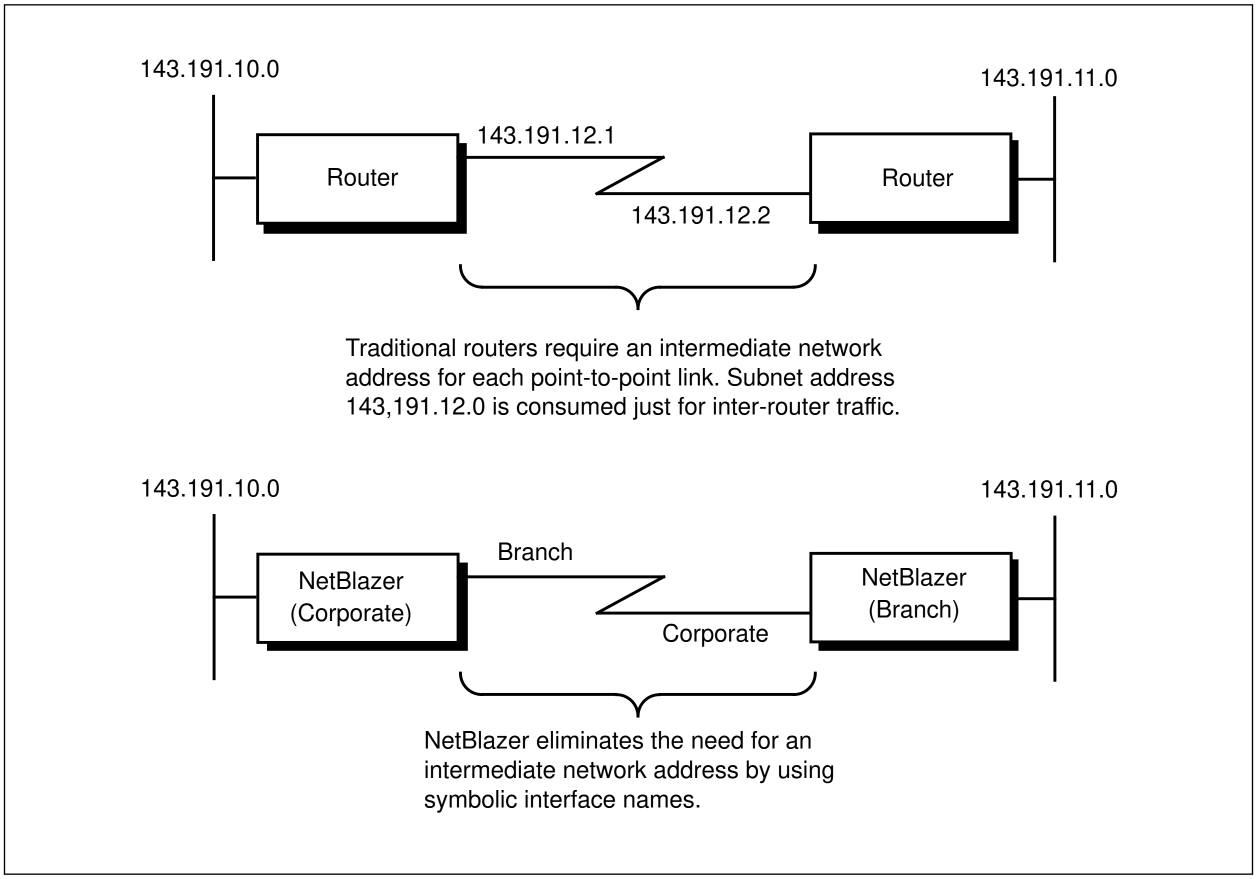


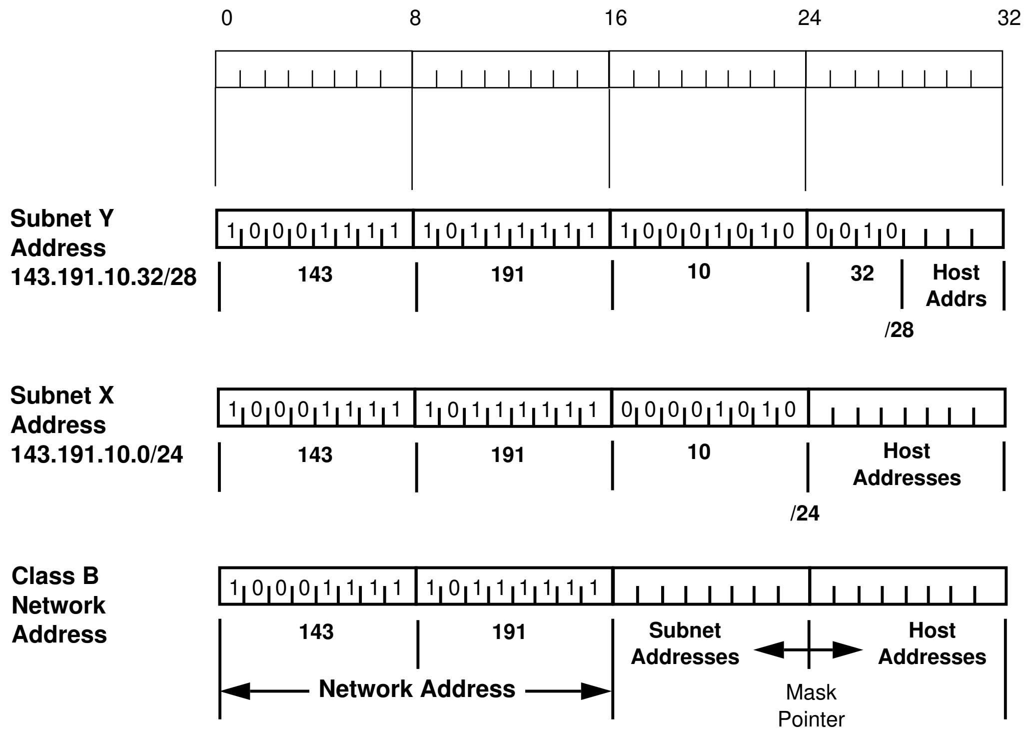
Figure 1-4. NetBlazer Routing Technique

Setting Up Subnetwork Addresses

The NetBlazer expects network addresses, entered in the routing table, to be accompanied by a subnet bit mask which specifies the number of high-order bits allocated to the network portion of the IP address. A typical network IP address would be entered as 143.191.10.0/24 where the /24 indicates that the first 24 bits are to be treated as the network part of the IP address.

Routing addresses with larger subnet bit masks take precedence over other routes, and appear in the routing table first. In other words, routes with a 32-bit subnet mask appear in the routing table first because they have the highest order of precedence. Routes with a 24-bit mask appear after 32-bit routes, but before routes with a 16-bit subnet mask.

Figure 1-5 illustrates how noncontiguous address space is allocated between two networks. Since Subnet Y has a larger subnet bit mask than Subnet X, destination addresses ranging from 143.191.10.32 through 143.191.10.47 will be routed to Subnet Y even though these addresses also fall within the range of addresses designated for Subnet X.



Host Addresses	Route to:
143.191.10.1 - 143.191.10.31	Subnet X
143.191.10.32 - 143.191.10.47	Subnet Y
143.191.10.48 - 143.191.10.255	Subnet X

Note:

Routes with more significant bits in their subnet bit masks take precedence over other routes.

Figure 1-5. NetBlazer Subnet Addressing Technique

Serial Line Controls

When the NetBlazer is turned on or rebooted, it scans the asynchronous lines to automatically detect the type of modem attached to each line, and resets the lines and modems to a known state based on information contained in the MODEMCAP.TXT file. Any line that does not have a modem attached to it is treated as a login terminal port.

The default line and modem settings are satisfactory for most applications. However, if you have special requirements, a set of commands is provided that allow you to modify the operating characteristics of a line. See Chapter 7, Asynchronous Lines Usage, in this manual and the description of the **line** commands in the *NetBlazer Command Reference Manual* for additional information.

General Configuration Planning

This section should help you determine the information you will need prior to actually configuring the NetBlazer.

While performing the NetBlazer configuration, you may need to refer to the *NetBlazer Command Reference Manual* for additional information on using the various configuration commands.

Tips: *Use the NetBlazer default configuration values if you do not have a reason to change them.*

Save your current configuration often, especially when doing something complicated. Saving before you start a complicated process, then rebooting will bring you back to where you were should you run into trouble.

Sometimes its easier to start over from the beginning than to make several changes, so don't be afraid to do so. To start over, copy the DEFAULT.CNF *file to the* STARTUP.CNF *file and reboot.*

The **history** *mechanism is a useful tool for entering commands based on earlier commands.*

The **grep** *command is useful in quickly locating specific information from a file. For example you may wish to locate all the routes that use a specific gateway in the* STARTUP.CNF *file.*

Using **list commands | grep** *<desired command> is a quick way to find the location of a desired command.*

Refer to the Chapter 2, Using the NetBlazer Commands, for details on using the **history** *and* **grep** *commands.*

First Things First

Before starting the NetBlazer configuration, make a working copy of the NetBlazer system diskette, and put the distribution diskette in a safe place. Under no circumstances should you use the distribution diskette as a work diskette.

Next log in to the NetBlazer as root, set the NetBlazer date and time and assign a password to root.

Refer to Chapter 3, Getting Started, for instructions.

Local IP Router Configuration Requirements

When configuring the NetBlazer as a gateway between LANs, you will need the following information:

- The host name which is assigned to the NetBlazer.
- A unique IP address for each Ethernet or Token Ring interface board installed in the NetBlazer. The NetBlazer automatically assigns each board an interface name, i.e. en0 – en2 for Ethernet and tr0 – tr2 for Token Ring.
- Routing information for any LAN segment which is not directly connected to the NetBlazer.

Tip: *Using RIP (Routing Information Protocol) to dynamically maintain the NetBlazer's routing table allows the NetBlazer to automatically learn the routes to indirect LAN segments from other routers.*

- The IP address of any domain name server which can be used by the NetBlazer to resolve a domain name to its corresponding IP address.

Dial-up IP Router Configuration Requirements

When configuring the NetBlazer as a dial-up IP router between the local network(s) and one or more remote locations, you will need the following information for each remote site:

- An interface name for the dial-up IP interface. Using the host name of the remote NetBlazer as the interface name simplifies setting up a dial-up interface.
- The maximum transmission unit (MTU) for this interface.
- The serial link framing protocol to use with this interface. You can choose either SLIP or PPP framing to encapsulate IP packets sent over the serial link. Both ends of the link must agree.
- The maximum number of lines to use when sending IP packets through this interface.
- Login authentication information, i.e. login ID, password and crypto key, for dial-in and/or dial-out connection through this interface.

Tips: *The login ID sent from the remote site <u>must</u> match the local interface name. The host name of the local NetBlazer is typically used by default as the login ID sent to the remote NetBlazer when dialing out to a remote site.*

Using the same password and/or crypto key in both directions simplifies setting up the user authentication records.

- The phone number(s) and characteristics group(s) used to establish one or more dial-out connections with the remote NetBlazer.

- The network address and subnet mask for each network at the remote location which can be accessed through this interface. Any network which you do not want advertised by RIP should be designated as private.

Terminal Server Configuration Requirements

To configure the NetBlazer as a terminal server, you simply need to create a user ID for each user who will be calling into the NetBlazer from a remote site. However, you may want to set up user destination groups before assigning user IDs.

You can configure a NetBlazer asynchronous line to automatically establish a telnet session with a specific host. This allows a remote user to transparently start a telnet session with a designated host as if he or she had called the host directly.

The NetBlazer can be set up to automatically call back a remote user at a designated dialout destination. To implement call back security, you will need at least two modems, one to receive a call back request, and the other to call the remote user back. A dialout destination record must also be set up for each destination site that will be using the call back feature.

Refer to Chapter 9, User IDs and Destination Groups, for instructions on configuring the NetBlazer as a terminal server.

Modem Server Configuration Requirements

To configure the NetBlazer as a modem server, you need to add a dialout destination record to the dialout destination list for each remote system which may be called by the NetBlazer, and create a user ID for each user on the network who will be logging into the NetBlazer via telnet, and dialing out to a remote system.

Refer to Chapter 8, Dialout Destinations, and Chapter 9, User IDs and Destination Groups, for instructions.

Using the NetBlazer Commands 2

NetBlazer commands are arranged in a tree-structured hierarchy, starting at the top command level, and working down through several subcommand levels as illustrated in Figure 2-1.

Moving Around in the Command Tree

To go to a new command level, enter the name of the new command level. Usually only the first few characters of the command need to be entered to uniquely distinguish it from other commands. For example, to get to the Configure IP Route command level from anywhere in the command tree, enter either **route** or **ro** at the command prompt.

When a command is entered, the NetBlazer first checks the current command level for the command. If it is a valid command for that level, the NetBlazer invokes the function associated with the command at that level. If not, the NetBlazer searches the command tree for the specified command. To illustrate, assume that you are currently at the Configure User command level. Entering **list** displays a list of users; whereas entering **route list** displays the list of routes without leaving the current command level.

The **up** command takes you up to the next higher level in the command tree. For instance, if you are currently at the Configure IP command level, entering an **up** command moves you up one level to the Configure command level.

The **top** command takes you to the top level of the command tree.

Getting Help

The **?** command displays a list of possible commands at the current command level. If you enter a command before the **?**, the syntax for that command is displayed.

Using the History Function

The NetBlazer history mechanism saves a list of recent commands. You can use this list to avoid repetitive typing by using a shorthand notation to enter commands based on previous commands.

The history mechanism uses the exclamation mark (!) followed by a variable to reference a previous command. The reference variable may be a second exclamation mark (!) to reference the previous command, a number to reference a recent command by its event number, or a character string to reference the most recent command that begins with or contains the character string.

Table 2-11 at the end of this chapter summaries the various options of the history mechanism. For a detailed description of the **history** command, see the description in the *NetBlazer Command Reference Manual*.

Using the More and Grep Commands

The **more** and **grep** commands are two useful commands for displaying the contents of a file or the piped output of a list command.

The **|** symbol (vertical bar) is used in a NetBlazer command line to indicate that you want to pipe (or pass) the output or the command on the left of the vertical bar to the command on the right of the vertical bar.

The **more** command displays the content of a file or the piped output of a command one screenful (23 lines) at a time. When the screen is full you are prompted to press any key except the “q” key to display the next screenful. Pressing the “q” key stops the display and returns the command prompt.

To display the contents of a file use the **more** <file-name> command. For example, entering **more passwd** displays the contents of the PASSWD file one screenful at a time.

To display the output of a NetBlazer command one screenful at a time, use the **|more** command. For example, entering **route list | more** pipes the output of the **route list** command through **more**.

The **grep** command, located at the Configure command level, searches a file or the piped output of a command for all occurrences of a specified string.

To display all lines in a file that contain a specified string, use the **grep** <string> <filename> command . For example, entering **grep route startup.cnf** displays a list of all the route commands in the STARTUP.CNF file.

To scan the output of a NetBlazer command for lines containing a specified string, use the **|grep** <string> command. For example entering **route list |grep en0** displays a list of routes that use the en0 interface.

Command Summary

This remainder of this chapter summarizes the NetBlazer commands by functional group.

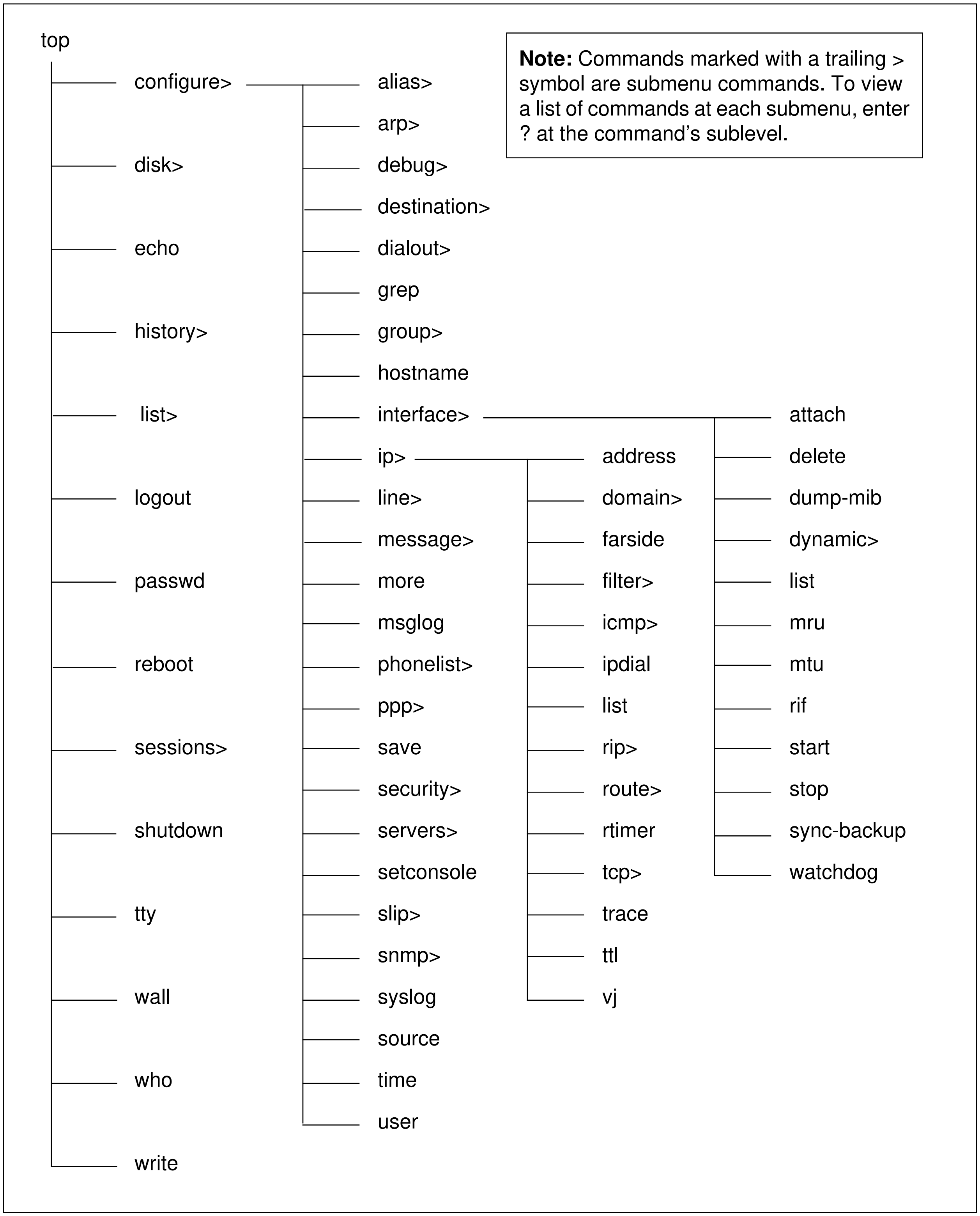


Figure 2-1. NetBlazer Command Tree

Disk Maintenance Commands

The **disk** commands are used to perform functions similar to the MS-DOS disk utilities, such as listing a directory of contents, copying a diskette, renaming a file, etc. Although the NetBlazer uses its own operating system, the file format of the diskettes are MS-DOS compatible.

The NetBlazer disk commands are listed in Table 2-1.

Table 2-1. Disk Commands Summary

Command	Function
chmod	Disable/enable write protection for a specified file.
copy	Copy a specified file.
delete	Delete a specified file.
dir	List a directory of files on the diskette.
diskcopy	Create a duplicate copy of the NetBlazer diskette.
format	Initialize a blank diskette for use with the NetBlazer.
rename	Rename a specified file.

IP Packet Routing

Commands used to configure the NetBlazer interfaces and control routing packets between network segments are listed in Table 2-2.

Refer to Chapter 4, Local IP Router Configuration, for instructions on how to configure the NetBlazer as a local IP router. Also see Chapter 10, IP Packet Filtering, if you want to use packet filtering to restrict packet routing through a particular interface.

Table 2-2. IP Packet Routing Commands Summary

Command	Function
hostname	Assign a host name to the NetBlazer.
ip address	Assign a global IP address to the NetBlazer, and assign an IP address to each of the static interfaces. Entering an **ip address** <ip-addr> command assigns a global IP address to the NetBlazer. Entering an **ip address** <ip-addr>/<bits> <interface> command assigns an IP address to a static interface, adds an ARP broadcast address to the ARP table, and adds a network route to the routing list.
interface commands	Interface control and maintenance.
domain commands	Configure the NetBlazer as a domain name server client.

Table 2-2. IP Packet Routing Commands Summary (Continued)

Command	Function
arp commands	ARP table maintenance.
route commands	Route table control and maintenance.
rip commands	Control and maintain destination list for RIP advertisements sent by the NetBlazer. Also, maintain a list of source addresses from which the NetBlazer will not accept RIP information.
filter commands	Maintain a list of IP packet filters used to restrict IP packet routing through a particular interface.

Synchronous Interface

Commands listed in Table 2-3 control routing IP packets through the synchronous interfaces.

Refer to Chapter 6, Synchronous Interface Configuration, for instructions on configuring and using the synchronous interface.

Table 2-3. Synchronous Interface Commands Summary

Command	Function
interface commands	Synchronous interface (syn0) control. Commands at this level allow you to check the status of the synchronous interface, stop or start the interface, and set up a watchdog which monitors the syn0 interface and should the link go down redirects the syn0 interface traffic through dynamic backup interface.
route commands	Control routing IP packets through the syn0 interface.
ppp commands	Control the point-to-point framing protocol used to transfer packets over the synchronous link.

Dedicated SLIP or PPP Links

Commands listed in Table 2-4 control the configuration and operation of dedicated serial line links.

For instructions on how to configure an asynchronous line as a dedicated SLIP or PPP link, see Setting Up a Dedicated SLIP or PPP Link on page 7-2.

Table 2-4. Dedicated SLIP and PPP Commands Summary

Command	Function
interface attach	Attach a packet mode interface to an asynchronous line.
ppp commands	Control the point-to-point framing protocol used to transfer packets over the dedicated PPP link.

Dynamic Dial-Up Interfaces

Commands listed in Table 2-5 establish and maintain dynamic interface which are used to transfer IP packets between a local and remote site using the public telephone network.

Instructions for setting up a dynamic interface between the local and remote site are provided in Chapter 5, Dynamic Interface Configuration.

Table 2-5. Dynamic Interface Commands Summary

Command	Function
ipdial	Start a dialog that steps you through the process of creating a dynamic dial-up interface to a remote site. The NetBlazer use information obtained from ipdial to create: • a dynamic interface record for the remote site. • one or more dialout records used to call the remote site. • a user record to authenicate a dial-in connection from the remote site. • add routes for the remote site.
dynamic commands	Set up and maintain dynamic interfaces, used to route IP packets over the public telephone network.
route commands	Maintain routes to remote networks through the dynamic interfaces.
dialout commands	Maintain a list of dial-out destinations used to establish a connection with a remote site.
users commands	Maintain a list of users authorized to log into the NetBlazer.

Users and Destination Groups

Commands listed in Table 2-6 are used to set up and maintain user accounts for those user who will be logging in to the NetBlazer.

Refer to Chapter 9, User IDs and Destination Groups, for instructions on setting up user records and destination groups.

Table 2-6. Users and Destination Groups Command Summary

Command	Function
user commands	Set up and maintain list of authorized users.
destination commands	Create destination groups, which allows authorized user to easily start a telnet or dial-out session with a predefined destination.
alias commands	Set up a special command alias that automati cally connects a user to a pre-defined host when the user logs in to the NetBlazer.

Sessions

Commands listed in Table 2-7 allow authorized users to start, stop and control communication sessions with local or remote machines.

Refer to the *NetBlazer User's Guide* for instructions on how to log in to the NetBlazer and initiate a session with a TCP/IP host or remote system.

Table 2-7. Sessions Command Summary

Command	Function
bang	Equivalent to the ping command except instead of printing a line of statistics for each response, only a single ! character is displayed.
binary	Turn binary mode on or off for a specified telnet session.
close	End a telnet session with a graceful, negotiated termination.
connect	Start a telnet or dialout session t using a predefined destination name.
continue or fg	Reactivate a session after it has been suspended.
dial	Establish a character mode connection with a remote site specified by a dial-out record.
disconnect	Terminate a character mode dial-out connection.
eol	Unix end-of-line marker that is used when a remote Unix host does not respond to a standard EOL sequence.
finger	Get information about a specific user on the network.
ftp	Transfer files between the NetBlazer and a remote host that supports FTP.
list or jobs	Display a list of sessions currently active on the NetBlazer.
kill	Force an abrupt termination of a session.
ping	Send ICMP echo requests to a specified host, then measures the time taken for the host to respond.
rlogin	Automatically logs a user onto a remote host by providing the NetBlazer user ID and password information to the remote host.
send	Issue a predefined character sequence to a remote host, such as an escape sequence.
telnet	Establishes an interactive session with an TCP/IP host.
terminal	Send terminal information to a UNIX host.

Network Management and Diagnostics

Commands listed in Table 2-8 are used to monitor and control the activity of the NetBlazer.

Table 2-8. Network Management and Diagnostics Command Summary

Command	Function
snmp commands	A group of Simple Network Mangement Protocol (SNMP) commands used to obtain statistical and other network management information.
syslog commands	Send NetBlazer activity information to a specified UNIX host, to a terminal, to a buffer in memory, or to the console.
icmp list	Display a list of error and control statistics maintained by the NetBlazer.
tcp commands	Set up and manage specific TCP (Transmission Control Protocol) parameters used by the NetBlazers.
list commands	Display various NetBlazer status information.
debug commands	A suite of network and NetBlazer troubleshooting facilities.

Servers

The NetBlazer servers are listed in Table 2-9.

Table 2-9. Server Commands Summary

Command	Function
discard	Accepts and acknowledges data received on TCP port 9. Data is discarded.
echo	Accepts and acknowledges TCP segments received at port 7. Data is echoed back to the original source.
finger	Allows other hosts to query the NetBlazer for information about NetBlazer users using the finger protocol.
raw	Allows a raw TCP connection from a host to the modem or other asynchronous device attached to a NetBlazer line.
ftp	Enables using FTP (File Transfer Protocol) to transfer new software and configuration files to the NetBlazer from a remote host.
rip	Allows the NetBlazer to learn routes to other networks from other routers on the network.
snmp	Used for network management tasks.
telnet	Allows users to log into the NetBlazer from a TCP/IP host.
v25bis	Must be turned on to use the synchronous line with a ISDN terminal adapter.

Productivity Tools

This group of commands provide a variety of network management tools and conveniences to the System Administrator.

Table 2-10. Productivity Commands Summary

Command	Function
alias	Sets up and maintain command aliases.
grep	Search a file for a specified string. Can also pipe output of a list to grep.
history commands	Maintain and display a list of the last 20 commands entered into the NetBlazer. Provides a “shorthand” method of entering commands based on the history list.
message commands	Enter and maintain messages displayed at startup, login and shutdown.
more	Display the contents of NetBlazer files on diskette a page (24 lines) at a time. The output of a list can also be piped to more.
source	Run NetBlazer commands stored in a file.
who	List all users currently logged onto the NetBlazer.
write	Write a message to a user currently logged onto the NeBlazer.
wall	Write to all users currently logged onto the NetBlazer.

Table 2-11 summarizes the facilities of the history command.

Table 2-11. History Commands Summary

Command Syntax	Description
history on [<number of lines>]	Turns on the history mechanism and optionally sets the maximum number of lines in the history list.
history off	Turns off the history mechanism and clears the history list.
history prompt	Displays an event number at the end of each command prompt when history is on.
history status	Displays the current settings of the history mechanism.
history list [<number of lines>]	Displays a list of recent commands (events).
!!	Invokes the last command.
!<event number>	Invokes the command specified by the event number.
!<string>	Invokes the most recent command beginning with <string>.
!?<string>**?**	Invokes the most recent command containing <string>.
!<event>**:p**	Displays and makes current the specified event, but does not invoke the command.
!<event>**:**<number>	References field <number> of the specified event.

Table 2-11. History Commands Summary (Continued)

Command Syntax	Description
!<event>**:**<start>**–**<end>	References fields ranging from <start> to <end> of specified event.
^<old string>**^**<new string>**^**	Replaces <old string> with <new string> in last command.

Getting Started 3

Before you begin working with the NetBlazer, you should make a backup copy of the system diskette, set the NetBlazer's date and time, and assign a password to the root user ID.

To accomplish these tasks, you must have a dial-in workstation that you can use to call in to the NetBlazer, or an asynchronous terminal connected to the NetBlazer through a null modem cable. Refer to your *NetBlazer Installation Guide* for instructions on setting up an administrative workstation.

If setting up an administrative workstation is not feasible and you need to access the NetBlazer from a LAN workstation, you can preconfigure the NetBlazer system diskette by using a text editor on an MS-DOS system to edit the STARTUP.CNF file, and then telnet to the NetBlazer from a host on the Ethernet network attached to the NetBlazer's en0 interface to complete the configuration. See Using MS-DOS to Preconfigure the STARTUP.CNF File for instructions.

Using MS-DOS to Back Up the System Diskette

Although the NetBlazer has its own operating system and disk utilities, its diskette file format is compatible with MS-DOS. Therefore as an alternative to using the NetBlazer's **disk** commands, you can use the **DISKCOPY** utility on an MS-DOS system that supports 3.5 inch high-density diskettes to make a backup copy of the system diskette. Refer to your MS-DOS Reference Manual for instructions on copying a diskette.

Use the duplicate diskette as a working copy and put the original NetBlazer system diskette away in a safe place.

Using MS-DOS to Preconfigure the STARTUP.CNF File

If you have a terminal attached to one of the NetBlazer's asynchronous ports or a dial-in workstation that you can use to call into the NetBlazer, you do not need to preconfigure your STARTUP.CNF file. Skip ahead to Starting the NetBlazer and Logging in as Root on page 3-3.

If you do not have a way to connect to the NetBlazer through one of the asynchronous ports, you can use an MS-DOS editor to preconfigure the STARTUP.CNF file so that the NetBlazer can be accessed from a LAN workstation the first time it is powered up.

Caution: *When editing the STARTUP.CNF file, use a text editor or word processor that does not truncate or break lines that wrap around, and be sure to save the file in ASCII text format.*

Edit the STARTUP.CNF file to include an **ip address** and **route** command as shown in **bold** in Figure 3-1.

```
configure setconsole asy00
configure hostname NetBlazer
configure ip address 143.191.63.1
configure ip route add 143.191.63/24 en0
configure group add dialout
configure group modify dialout add dialout
configure group add v32dialout
configure group modify v32dialout add dialout
configure group modify v32dialout add v32
configure group add request
configure group modify request add request
configure group add pepdialout
configure group modify pepdialout add dialout
configure group modify pepdialout add pep
configure group add raw
configure group modify raw add raw
configure group add raw_dial
configure group modify raw_dial add raw
configure group modify raw_dial add dialout
configure group add v25dialout
configure group modify v25dialout add v25bis
configure ip route addprivate 127.0.0.1 loopback
configure server start telnet
configure server start ftp
configure server start rip
configure server start snmp
configure server start finger
configure dialout add tbupdate 1-408-745-3700 dialout 10
configure interface dynamic add tbupdate "tbupdate" "netblazer" #0 "ics" slip
1
1500
configure ip vj tbupdate auto
configure ip route addprivate 143.191.20.1/32    tbupdate
configure message hello  nlmsgnl  nlmsgnl "Telebit's NetBlazer Version %s"
nlmsg
nl
configure alias add update ftp 143.191.20.1
```

Figure 3-1. Preconfiguring the STARTUP.CNF File

Note: *The IP addresses shown in this and other examples throughout this manual are for illustrative purposes only.*

These two commands provide the minimum information required to access the NetBlazer from a LAN workstation on the Ethernet LAN that is attached to the NetBlazer's en0 interface. The **ip address** command assigns a global IP address to the NetBlazer, and the **route add** command creates a route to the subnetwork which is attached to the NetBlazer through the en0 interface. The /24 in the **route add** command's IP address indicates that the route is for a subnetwork with the first 24 bits being the network portion of the address.

If your NetBlazer is connected to a Token Ring network instead of an Ethernet network, use tr0 in place of en0 in the above example.

You should now be able to power up the NetBlazer using the modified STARTUP.CNF file, and log in to the NetBlazer through a telnet session from any host which is on the network attached to the NetBlazer through the en0 interface.

Starting the NetBlazer and Logging in as Root

Unless you have preconfigured the NetBlazer system diskette as described in the previous section, you must use a dial-in workstation or terminal connected to the NetBlazer the first time that you log in to the NetBlazer. After you have configured the NetBlazer as an IP router, you can use telnet to reach the NetBlazer over the network.

1. Verify that all the modems connected to the NetBlazer are turned on and ready. The MR (Modem Ready) and CTS (Clear to Send) lights on the front of each modem should be lit.
2. Insert your working copy of the NetBlazer system diskette (or the write-protected original system diskette) into the NetBlazer. The diskette label should face upward when you insert the diskette.
3. To log into the NetBlazer from a dial-in workstation, you must first establish a connection with the NetBlazer by using your modem (or communications software) dial command. For example, to call a NetBlazer at 555-1234 using a Telebit modem dial command, enter:

   atdt 555–1234

   If you are using a terminal connected directly to the NetBlazer through a null modem cable, the NetBlazer login prompt is displayed at your terminal when the system finishes loading.
4. When a connection is established, the NetBlazer displays a login prompt. Enter **root** at the login prompt.

```
Telebit's NetBlazer Version 1.5

NetBlazer login: root

NetBlazer:Top>
```

The NetBlazer then displays its command prompt. This indicates that the NetBlazer is ready and waiting for you to enter a command.

The NetBlazer commands are arranged in a tree-structured hierarchy, starting at the Top command level, and working down through several subcommand levels. Instructions for moving around and using the various commands within the NetBlazer's command structure are provided in the previous chapter.

Using NetBlazer's Diskcopy to Backup the System Diskette

If you have already made a backup copy of your NetBlazer system diskette on an MS-DOS system, skip ahead to the next section, Setting the Current Date and Time, on page 3-5. Otherwise, perform the following steps to make a backup copy of your NetBlazer system.

1. Go to the Disk command level and enter the **format** command as shown below.

```
NetBlazer:Top> disk
NetBlazer:Top>Disk> format
Format disk [n]? y
```

2. Enter **y** to start formatting the diskette.

```
Insert diskette to be formatted and press return
```

3. Insert a new high-density diskette into the NetBlazer disk drive and press the return key. The following messages are displayed on the screen as the formatting routine progresses.

```
Formatting cylinder 79
Verifying cylinder 79
No bad sectors on disk
Format another disk [n]?
```

4. Unless you want to format another diskette, press return or enter **n** to the "Format another disk" prompt.

```
Insert system diskette and press return
```

5. Insert your write-protected original system diskette in the NetBlazer disk drive and press the return key.

6. Enter a **diskcopy** command at the Disk command level.

```
NetBlazer:Top>Disk> diskcopy
0% done...Insert source diskette and press return
```

7. Your original system diskette should already be in the NetBlazer disk drive. Press the return key.

```
Insert destination diskette and press return
```

8. Insert your newly formatted diskette into the NetBlazer disk drive, and press the return key.

9. Continue following the instructions from the NetBlazer until the following message is displayed.

```
Disk copy complete
```

10. Use the duplicate diskette as a working copy and put the original NetBlazer system diskette away in a safe place.

```
Insert system diskette and press return
```

11. Make sure that you have your working copy of the system diskette inserted in the disk drive, then press the return key.

12. Enter a **dir** command at the Disk command level to display a list of files that are contained on the system diskette.

```
NetBlazer:Top>Disk> dir
Directory of A:/

TELEBIT  SYS     14844  07-22-91   15:08
FRED            604160  02-25-92   16:25
MODEMCAP TXT     17095  02-25-92   16:26
CHAT     TXT      1588  02-25-92   16:26
DEFAULT  CNF      1265  02-25-92   16:26
STARTUP  CNF      1265  03-12-92   22:14
6 File(s) (640217 bytes) 816128 bytes free
```

Setting the Current Date and Time

To set the NetBlazer's date and time, enter **time** at the command prompt.

```
NetBlazer:Top> time
```

The NetBlazer prompts you for the current date and time as shown below.

```
Current time is Thu Mar 12 22:38:38 1992
Enter the year [92]:
Enter the month number (1 to 12) [3]:
Enter the day of the month [12]:
Enter the hour (0 to 23) [22]:
Enter the minute [38]:
Enter the current second [38]:
Verify: Thu Mar 12 22:38:38 1992
```

Parameters shown in square brackets indicate the current setting. If the value shown is correct, press the return key to accept the current value; otherwise enter the correct value.

Entering a **time list** command displays the NetBlazer's date and time.

Setting the Root Password

1. Go to the Configure User command level by entering **user** at the command prompt.

```
NetBlazer:Top> user
NetBlazer:Top>Configure>User>
```

2. Set the root password by entering the following command.

```
NetBlazer:Top>Configure>User> password root <password>
```

Choose a <password> that you can remember but is not easily guessed by other users.

3. Enter **list** at the Configure User command level to display a list of current users.

```
NetBlazer:Top>Configure>User> list
USERID      PASSWORD        CRYPTO          USE       PRIVS   DEST-GROUP
root        abiO2vUZhAKHo                   login     SC
```

Only the root user ID record is displayed at this time. The password is encrypted to prevent unauthorized viewing. Also notice that the root user ID has full system status and configuration privileges as indicated by the "SC" in the PRIVS field.

4. Verify that you can log into the NetBlazer using the new root password by issuing a **logout** command and then logging back into the NetBlazer. If you cannot log back in, reboot the NetBlazer by pressing the reset button on the NetBlazer front panel.

5. Make sure that your NetBlazer system diskette is inserted in the diskette drive, and enter **save** at the Configure User command level to save the root user ID and password:

```
NetBlazer:Top>Configure>User> save
```

6. The user information is saved in a file named PASSWD on the NetBlazer system diskette. You can view the contents of this file by entering **more passwd** at the command prompt.

```
NetBlazer:Top>Configure>User> more passwd
root:abiO2vUZhAKHo:0::,,,::
```

For a description of the PASSWD file format, refer to NetBlazer PASSWD File on page 9-6.

Local IP Router Configuration 4

This chapter provides instructions on how to configure the NetBlazer to route IP packets between local networks. You need to know the network addresses of each Ethernet and/or Token Ring LAN that is attached directly to the NetBlazer, and you need to choose a host name for the NetBlazer.

You also need to decide how the NetBlazer will learn the routes to other LANs, which are not directly accessible to the NetBlazer. You have two choices: manually add static routes to the NetBlazer's routing table, or configure the NetBlazer to dynamically learn the routes to other indirect networks using RIP (Routing Information Protocol).

Tip: *Static routes work best for small stable networks with just a few routes. For complex networks with many paths, like the network shown in Figure 4-1, RIP provides a more flexible and easier way to maintain the list of routes used by the NetBlazer.*

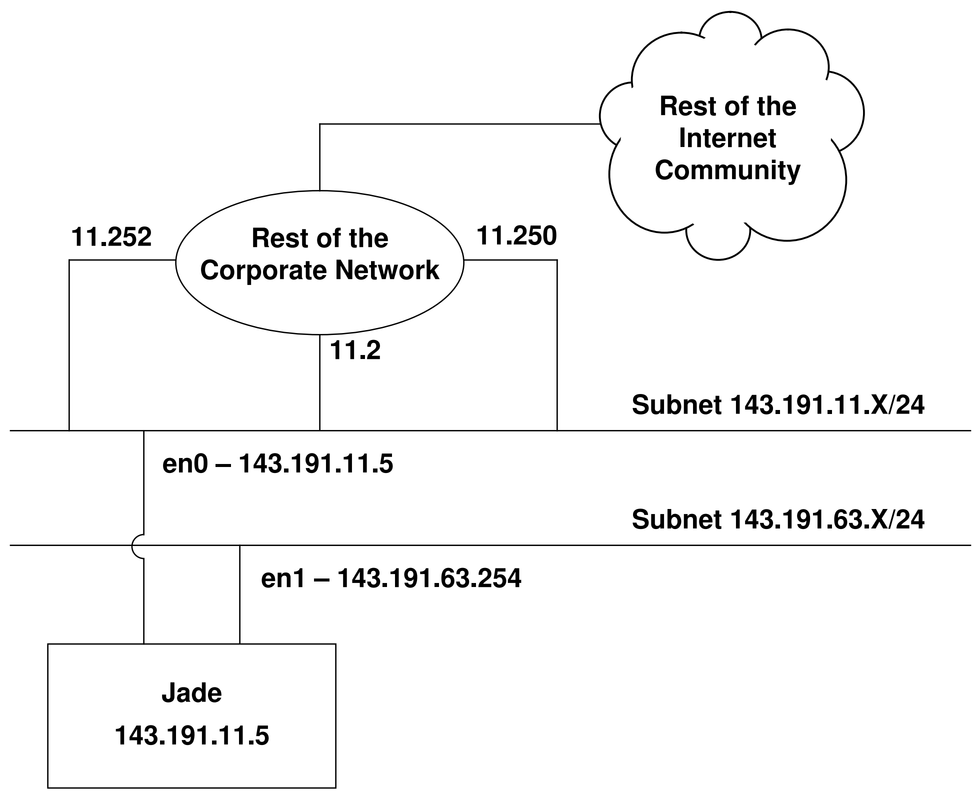


Figure 4-1. Local IP Network Example

The remainder of this chapter describes how to configure the NetBlazer (Jade, shown in Figure 4-1) to route IP packets between LANs.

Assigning a Hostname and IP Address to the NetBlazer

Go to the Configure command level and assign a hostname to the NetBlazer as shown below:

```
NetBlazer:Top>Configure>User> configure
NetBlazer:Top>Configure> hostname Jade
Jade:Top>Configure>
```

The host name you choose, Jade in the above example, is displayed as part of the command prompt.

Go to the Configure IP command level, and enter the IP address assigned to the NetBlazer:

```
Jade:Top>Configure> ip
Jade:Top>Configure>IP> address 143.191.11.5
Set IP address(es) of 3 interface(s)
Set IP address(es) of 10 lines(s)
```

Note: *The IP address used in this example and other examples throughout this manual are for illustrative purposes only. You should use your own IP addresses when configuring the NetBlazer.*

Assigning an IP Address to Each LAN Interface

When the NetBlazer is turned on, it automatically creates an interface for each Ethernet and/or Token Ring board installed in the system. Before an interface may be used for routing IP packets, it must be assigned an IP address and static route. When using RIP to broadcast routing information to other routers, an ARP broadcast address must also be assigned to each network.

Fortunately, only one command per interface provides the NetBlazer with enough information to automatically configure each interface as illustrated below.

```
Jade:Top>Configure>IP> address 143.191.11.5/24 en0
Set IP address of interface 'en0'
Jade:Top>Configure>IP> address 143.191.63.254/24 en1
Set IP address of interface 'en1'
```

These commands assign IP addresses 143.191.11.5 and 143.191.63.254 to the en0 and en1 interfaces, respectively.

The **/24** in each of the IP addresses is a subnet bit mask indicating that the first 24 bits are to be treated as the network part of the IP address. The NetBlazer uses this information to automatically create a static route and ARP broadcast address for each interface. If a subnet bit mask is not entered, the NetBlazer will not create a static route or ARP broadcast entry for the specified interface.

To see the effects of the above two **ip address** commands produce listings of the interface, local routes and ARP entries as shown below.

```
Jade:Top>Configure>IP> interface list
Name               IP address   MRU   MTU type    line      pkts in   pkts out
tbupdate         143.191.11.5  1500  1500 Dialout Down            0          0
en1            143.191.63.254  1500  1500 Ether                   0          3
en0              143.191.11.5  1500  1500 Ether                  28          3
loopback            127.0.0.1 65535 65535                         0          0
Jade:Top>Configure>IP> route listlocal
Dest            Len Interface    Gateway          Metric  P Timer  Use
127.0.0.1       32  loopback                      1       P 0      0
143.191.20.1    32  tbupdate                      1       P 0      0
143.191.11.0    24  en0                           1         0      0
143.191.63.0    24  en1                           1         0      0
Jade:Top>Configure>IP> arp list
received 0 badtype 0 bogus addr 0 reqst in 0 replies 0 reqst out 6
IP addr         Type  Interface   Time Q Addr
143.191.63.255  ether any         0      ff:ff:ff:ff:ff:ff
143.191.11.255  ether any         0      ff:ff:ff:ff:ff:ff
```

To view the Ethernet address of an interface, go to the Configure Interface command level and enter the **list** command with an interface name.

```
Jade:Top>Configure>IP> interface
Jade:Top>Configure>Interface> list en0
Name               IP address   MRU   MTU type    line      pkts in   pkts out
en0              143.191.11.5  1500  1500 Ether                  61          3
        Adapter address 00:80:ad:02:24:9e
```

Configuring the NetBlazer to Use RIP and/or Static Routes

In order to send IP packets to networks which are not connected directly to one of the NetBlazer's interfaces, the NetBlazer needs to know which interface and gateway to use to reach a particular destination network. This information is kept in the NetBlazer's routing table which includes two types of entries: static routes manually entered by an operator, and dynamic routes added by RIP.

Using RIP (Routing Information Protocol)

The NetBlazer is shipped with the RIP server turned on which allows it to accepts routing information from other routers on the network. If your NetBlazer is connected to a network with another router that is broadcasting RIP information, entering a **route list** command shows entries in the routing table that were dynamically added to the table by RIP.

If you do not want the NetBlazer to accept routing information from other routers on the network(s), you can turn the RIP server off by issuing a **server stop rip** command. Use a **rip refuse** <IP-address> command to stop the NetBlazer from accepting RIP information from one specific router.

If you want the NetBlazer to broadcast RIP information to other routers on the network(s), go to the Configure IP RIP command level and add a RIP broadcast address for each local network.

```
Jade:Top>Configure>Interface> rip
Jade:Top>Configure>IP>RIP> add 143.191.11.255 30 yes
Jade:Top>Configure>IP>RIP> add 143.191.63.255 30 yes
```

These commands cause the NetBlazer to send RIP information to each network's broadcast address in 30 second intervals. The last parameter (yes) in each entry tells the NetBlazer to use split horizons which tends to make the routing tables converge more rapidly when a gateway/router is no longer in service.

Enter **list** at the Configure IP RIP command level to verify your entries:

```
Jade:Top>Configure>IP>RIP> list
RIP: sent 22 rcvd 542 reqst 0 resp 542 unk 0 refused 0
Active RIP output interfaces:
Dest Addr       Interval Split
143.191.63.255  30       1
143.191.11.255  30       1
```

Setting Up Static Routes to Indirect Networks

You normally only need to set up static routes to indirect networks (those networks that are not connected directly to the NetBlazer) if you are not using RIP to dynamically maintain the routing table.

The **route add** or **route addprivate** commands are used to set up static routes. If you are using RIP and do not want a particular route advertised by RIP use **route addprivate**. Otherwise, use the **route add** command. These two commands use the following syntax:

route add <dest_addr>[/<bits>] <interface> [<gateway> <metric>]

route addprivate <dest_addr>[/<bits>] <interface> [<gateway> <metric>]

Both commands require a destination IP address, a subnet bit mask and an interface name. A gateway IP address and metric is also required if the destination network is more than one hop count away from the NetBlazer.

The following example shows you how to enter an indirect route to the 143.191.10.X subnetwork, which is accessible through the 143.191.11.252 gateway.

```
Jade:Top>Configure>Interface> route
Jade:Top>Configure>IP>Route> add 143.191.10/24 en0 143.191.11.252 2
```

Use the **route lookup** command to verify that you have enter the static route correctly.

```
Jade:Top>Configure>IP>Route> lookup 143.191.10/24
Dest            Len Interface    Gateway          Metric  P Timer  Use
143.191.10.0    24  en0          143.191.11.252   2         0      0
```

The new entry shows that network 143.191.10.X is 2 hops away from the NetBlazer, and IP packets destined for this network are sent to gateway 143.191.11.252 through the en0 interface.

Setting Up a Default Route

To keep the routing table small, you can create a default route that is used to route IP packets that do not have a designated route.

Caution: *If you are using RIP, only one router should have a static default route that is advertised by RIP. The remaining routers should learn the default route from RIP or use a private default route that is not advertised to the rest of the network.*

Use the **route addprivate default** command to set up a default route as shown in the next example.

```
Jade:Top>Configure>IP>Route> addprivate default en0 143.191.11.250 4
```

List the static route entries by entering a **route listlocal** command.

```
Jade:Top>Configure>IP>Route> listlocal
Dest            Len Interface    Gateway          Metric  P Timer  Use
127.0.0.1       32  loopback                      1       P 0      0
143.191.20.1    32  tbupdate                      1       P 0      0
143.191.10.0    24  en0          143.191.11.252   2         0      0
143.191.11.0    24  en0                           1         0      157
143.191.63.0    24  en1                           1         0      318
default         0   en0          143.191.11.250   4       P 0      0
```

Notice that a default route has been added to the table which routes all IP packets that do not have a designated route through interface en0 to gateway 143.191.11.250.

Configuring the NetBlazer as a Domain Name System Client

If your network includes one or more domain name servers, go to the Configure IP Domain command level, and add the IP address for each name server to the domain server list as illustrated below:

```
Jade:Top>Configure>IP>Route> domain
Jade:Top>Configure>IP>Domain> add 143.191.10.100
Jade:Top>Configure>IP>Domain> list
Server address      srtt    mdev    timeout queries responses
143.191.10.100      5000    0       5030    0       0
```

As a domain name system client, the NetBlazer accepts a host name provided by a user in a **telnet**, **ftp** or **ping** command, and queries the name server(s) identified in the domain server list for the IP address of the specified host. This IP address is then used to establish the desired connection. This allows users to enter domain names instead of IP address.

The **domain suffix** command, shown in the following example, allows you to define a suffix which is added to the end of any domain name supplied by the user that does not contain a dot (period). The NetBlazer treats domain names that include a dot as a complete domain name and will not add the suffix to the supplied name before querying a name server for the host IP address.

```
Jade:Top>Configure>IP>Domain> suffix telebit.com
```

If the domain suffix is set to telebit.com, as shown above, and a user issues a **telnet marketing** command to the NetBlazer, the NetBlazer will expand the domain name to marketing.telebit.com before requesting the IP address from a name server on the network.

Entering the domain suffix command without a suffix string displays the current domain name suffix.

Saving the NetBlazer System Configuration

Make sure that you have your working copy of the NetBlazer system diskette inserted in the NetBlazer, and enter **configure save** at the command prompt to save the NetBlazer configuration to diskette.

```
corp:Top> configure save
```

Anytime you modify the NetBlazer configuration, you should use the **configure save** command to update the STARTUP.CNF file on your NetBlazer system diskette.

Dynamic Interface Configuration 5

This chapter provides instructions on how to configure the NetBlazer to route IP packets over a dynamic dial-up interface. Setting up a dynamic interface is a fairly complex procedure involving four separate processes which must be done at both the local and remote site:

- Creating the dynamic interface record.You can configure a dynamic interface to only allow dial-out or dial-in connections, or to allow both dial-out and dial-in connections.
- Creating one or more dialout records to initiate a dial-out connection to the remote site.
- Creating a user record to authenticate a dial-in connection initiated from the remote site.
- Adding dynamic interface routes to the routing table.

To allow dial-out connections through a dynamic interface, the NetBlazer must have one or more dialout records with the same name as the interface. The NetBlazer uses the line characteristics and telephone number contained in the interface's dialout record(s) to select an appropriate asynchronous line and call the remote NetBlazer when a dynamic connection to the remote site is required.

To allow dial-in connections through a dynamic interface, the NetBlazer must have a user record with the same name as the interface. The NetBlazer uses the interface's user record to authenticate a connection by checking the remote site's response to a password request and/or crypto challenge. The remote NetBlazer's dynamic interface record must include a password and/or crypto key that matches the local NetBlazer's user record.

In order for the NetBlazer to know how to reach the hosts located at a remote site, you must add dynamic interface routes to the NetBlazer's routing table. If you configure the dynamic interface to allow dial-out connections, then you must add static routes for the remote site to the NetBlazer's routing table. However, if you configure the dynamic interface to only allow dial-in connections, you can choose to either use static routes or configure the dynamic interface to automatically add its routes to the NetBlazer's routing table when a connection is established from the remote site.

Auto routes differ from static routes in that they are only available while a dynamic connection is established, and are removed from the routing table when the connection is dropped.

An example of how to set up auto routes for a remote site is provided at the end of this chapter on page 5-13.

The remainder of this chapter explains how to set up the dynamic interfaces between the two NetBlazers, named Jade and Ruby, shown in Figure 5-1. Before attempting to set up your own dynamic interfaces, read through this chapter and make sure that you have all the necessary information.

Using Ipdial to Set Up a Dynamic Interface

The **ipdial** command simplifies the process of setting up a dynamic interface by prompting you for information required to complete each of the four phases discussed above.

Start by setting up a dynamic interface on the local NetBlazer, named Jade. Enter **ipdial** at Jade's command prompt.

```
Jade:Top> ipdial
```

This routine steps you through the process of setting up a dynamic interface, a dialout record, a user account and one or more routes to the remote site.

```
Name of remote (other) system: Ruby
```

Enter the name of the remote system. This name is assigned to the dynamic interface. It is also the named used to create the dial-out and/or user record.

```
Allow dial-out from this interface [y]?
Allow dial-in to this interface [y]?
```

To allow an IP connection with the remote site in either direction, dial-out or dial-in, press the return key or enter "y" in response to each of the above prompts. If you do not want to allow a dial-out or a dial-in connection, enter "n" to the appropriate prompt. You must answer "y" to at least one of these prompts, otherwise there is no point in creating the interface.

The remainder of this discussion assumes that you have entered "y" to both of the above prompts. If you answer "n" to either of the prompts, the dialog associated with that function will be omitted.

Creating the Dynamic Interface Record

At this point, the NetBlazer starts prompting you for information that it needs to create the dynamic interface record.

```
Begin adding interface 'Ruby' ...
User ID to send to remote system [Jade]:
```

Assuming you are allowing dial-outs from this interface, the above prompt allows you to enter the user ID which must be supplied to the remote system.

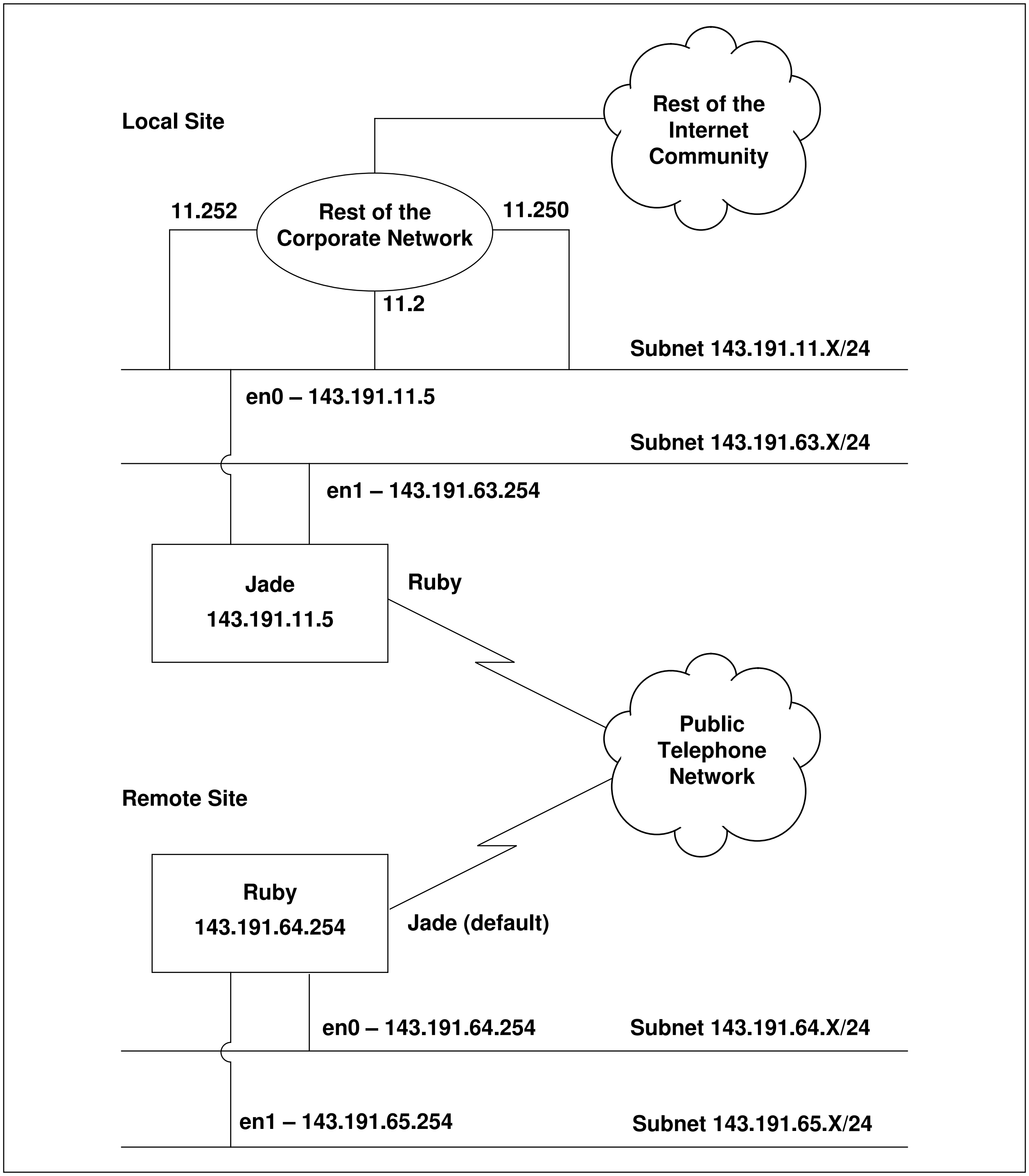


Figure 5-1. Dynamic Interface Example

Pressing the return key in response to the user ID prompt assigns the hostname of your NetBlazer as the user ID which is sent to the remote system.

The remote system must have a corresponding user ID in its user records to allow a connection. If you enter a password and/or crypto key as described below, the user record at the remote system must also include a matching password and/or crypto key.

```
Want to enter a password [y]?
```

If the remote system will be issuing a crypto challenge, entering a password is not necessary. In which case, you can enter “n” to the above prompt. However, if you want to use a password to authenticate the connection, press the return key or enter “y” to the prompt.

```
Enter the password for remote system (no echo):
Re-enter the password:
```

The password is not displayed on the screen as you type it, so you are asked to enter the password twice to ensure that you entered it correctly.

```
Will remote system issue a crypto handshake [n]? y
```

Crypto handshaking adds another level of security when attempting to connect with the remote system. Enter “y” to use crypto handshaking.

```
Enter the crypto key:
Re-enter the crypto key:
```

The above two prompts are used to enter the crypto key which is used to encrypt a crypto response sent to the remote system when a crypto challenge is received. The crypto key is not displayed on the screen as you type it, so you are asked to enter the crypto key twice to ensure that you entered it correctly.

```
Chat script [ics]:
```

This prompt allows you to assign a chat script to the interface. In most cases, you should press the return key to accept the default chat script supplied with the NetBlazer.

For more information about creating and using custom chat scripts, see CHAT.TXT File Definition in Appendix A, NetBlazer File Definitions.

```
SLIP or PPP (s|p) [s]:
```

You can use either SLIP or PPP to send packets. Press the return key or enter "s" to choose SLIP framing. Enter "p" to select PPP framing. *Both the local and remote NetBlazers must use the framing protocol.*

```
Maximum lines to connect to remote system [1]:
```

If both the local and remote NetBlazers have more than one telephone line connected to the NetBlazer, IP packets can be distributed over several telephone connections thereby reducing the time required to deliver the user's data. Enter the maximum number of lines to use when sending IP packets through this interface.

```
What is the Maximum Receive Unit (MRU) [1006]:
```

Unless you require a different MRU, press the return key to accept the default MRU of 1006.

```
Add dynamic interface=Ruby, user=Jade, chat=ics, using SLIP
    with crypto handshaking, max-lines=1, MRU=1006
Okay (yes|no|quit) [y]?
```

Press the return key or enter "y" to create the specified interface record. Otherwise enter "n" to discard this interface record and start a new interface record or "q" to discard the interface record and return to the command prompt.

Creating the Dialout Record(s)

Assuming you answered "y" to the "Allow dial-out from this interface?" prompt, the NetBlazer begins a dialog to obtain the information required to create one or more dialout records for the remote site.

```
Begin adding dial-out 'Ruby' ...
Phone number: 9-555-6789
```

Enter the telephone number of the remote connection. Refer to the Dial command description in your modem's reference manual for a description of acceptable dial strings.

```
Line characteristics [dialout]:
```

Enter an existing characteristics group name or press the return key to accept the default characteristics group (dialout). This entry is used to select a serial line with the same characteristics when placing a call to the remote site. If no line is available with the specified characteristics, the call is not placed.

```
Dialout: name=Ruby, phone=9-555-6789, characteristics=dialout
Okay (yes|no|quit) [y]?
```

Press the return key or enter “y” to create the specified dialout record. Otherwise enter “n” to discard this dialout record and start a new dialout record or “q” to discard the dialout record and return to the command prompt.

```
Add another dial-out [n]? y
```

If the remote site has more than one telephone line connected to the NetBlazer system and is not using rotary switching equipment, enter “y” and enter the next phone number and line characteristics.

```
Phone number: 9-555-4321
Line characteristics (dialout):
Dialout: name=Ruby, phone=9-555-4321, characteristics=dialout
Okay (yes|no|quit) [y]?
Add another dial-out [n]?
```

When you have finished entering a dialout record for each telephone line connected to the remote NetBlazer system, press return or enter “n” at the “Add another dial-out [n]?” prompt.

Creating the User Record

Assuming you answered “y” to the “Allow dial-in to this interface?” prompt, the NetBlazer begins a dialog to obtain the information required to set up a user ID account for the remote site.

```
Please add the information that the remote system 'Ruby'
        will use to contact this NetBlazer
Want to enter a password [y]?
```

If you intend to have the NetBlazer issue a crypto challenge, entering a password is not necessary. In this case, you can enter “n” to the above prompt. However, if you want to use a password to authenticate a dial-in connection from the remote site, press the return key or enter “y” to the prompt.

```
Enter the password (no echo):
Re-enter the password:
```

The above password prompts are used to specify the password that is required from the remote system. The password is not displayed on the screen as you type it, so you are asked to enter the password twice to ensure that you entered it correctly.

```
Issue a crypto handshake [n]? y
```

Enter “y” to issue a crypto challenge to a remote system that is attempting to establish a connection through this interface.

```
Enter the crypto key:
Re-enter the crypto key:
```

The above prompts are used to enter the crypto key that is used to decode a crypto response sent by the remote system when a crypto challenge is issued. The remote system must have an identical crypto key in its interface record to allow a successful connection. The crypto key is not displayed on the screen as you type it, so you are asked to enter the crypto key twice to ensure that you entered it correctly.

```
User: Ruby              privs:     dest-group:
Okay (yes|no|quit) [y]?
```

Press the return key or enter “y” to create the specified user record. Otherwise enter “n” to discard this user record and start a new user record or “q” to discard the user record and return to the command prompt.

Adding Routes for the Remote Site

Before using this interface, you must set up a route for each network which is reachable at the remote site.

Since this dynamic interface allows dial-out connections, you must add static routes for the remote site to the NetBlazer’s routing table. However, if you set up a dynamic interface that only allows dial-in connections and you want the routes to the network to be available only when a connection is established from the remote site, you can choose to not add static routes by answering “n” to the following prompt, and use the **dynamic auto-route** command to set up temporary routes to the remote network(s) instead.

See Setting Up Auto Routes to a Remote Site on page 5-13 for instructions on setting up automatic routes for a dynamic interface.

```
Add routes for interface [y]?
```

To enter the routes now, press the return key or enter “y” at the prompt. Enter “n” if you intend to add temporary routes later through the **dynamic auto-route** command.

```
Begin adding route thru 'Ruby' ...
Destination name or address/bits (Return to finish routes): 143.191.64/24
```

Enter the IP address and subnet bit mask for the remote network. The **/24** in the above example is a subnet bit mask indicating that the first 24 contiguous bits are to be treated as the network part of the IP address.

```
Metric [1]:
```

Enter the number of hops required to reach the destination network.

```
Private [n]:
```

If you do not want this network advertised through RIP enter “y” at the prompt. Otherwise, press the return key or “n” to designate the route not private.

```
Add route destination=143.191.64.0, bits=24, metric=1, not private
Okay (yes|no|quit) [y]?
```

Press the return key or enter “y” to create the route. Otherwise enter “n” to discard this route and start a route or “q” to discard the route and return to the command prompt.

Repeat the above process for all other networks at the remote site. For the example shown in Figure 5-1, a second route should be entered for network 143.191.65/24.

```
Route added
Begin adding route thru 'Ruby' ...
Destination name or address/bits (Return to finish routes): 143.191.65/24
Metric [1]: 2
Private [n]:
Add route destination=143.191.65.0, bits=24, metric=2, not private
Okay (yes|no|quit) [y]?
```

When you have finished entering routes for all networks at the remote site, press the return key to exit the **ipdial** routine.

```
Route added
Begin adding route thru 'Ruby' ...
Destination name or address/bits (Return to finish routes):
... Done adding dial IP
For proper communications, remote system should have:
1) Dynamic interface=Jade, user=Ruby, chat=ics, using SLIP
   with crypto handshaking, max-lines=1, MTU<=1006
2) Dialout=Jade
3) User=Jade
4) Route destination=143.191.11.5, interface=Jade, gateway=none, metric=1
```

The NetBlazer system at the remote site must be configured as described above. You also may need to provide additional routing information to the remote site if your site has more than one network.

You can view the above information about the remote site again at a later time by issuing a **farside** command to NetBlazer as illustrated below.

```
Jade:Top> farside Ruby
For proper communications, remote system should have:
1) Dynamic interface=Jade, user=Ruby, chat=ics, using SLIP
   with crypto handshaking, max-lines=1, MTU<=1006
2) Dialout=Jade
3) User=Jade
4) Route destination=143.191.11.5, interface=Jade, gateway=none, metric=1
```

Viewing the Results of the Ipdial Routine

To see the effects of running the **ipdial** routine, enter the following four commands to produce listings of the interface, routes, dialout and users.

```
Jade:Top> interface list Ruby
Name              IP address   MRU   MTU type    line      pkts in   pkts out
Ruby            143.191.11.5  1006  1006 Dialout Down           0          0
Idle: In 2:01:52:49   Out 2:01:52:49                IP:         0          0
Will save 'Ruby' vj: auto
Current vj: not active
Dynamic SLIP interface, Uname=Jade, Password=scarlet
Dial-out=Yes, Dial-in=Yes
Chat script=ics, Crypto key=scarlet, Max-lines=1, MRU=1006, MTU=1006
Trigger level=2, Timeout=180 s, Queue-max=8
cost = 0        bytespertick = 0

Jade:Top> route list |grep Ruby
143.191.65.0    24  Ruby                          1         0      0
143.191.64.0    24  Ruby                          1         0      0

Jade:Top> dialout list |grep Ruby
Ruby            9-555-6789         dialout
Ruby            9-555-4321         dialout

Jade:Top> user list |grep Ruby
Ruby        abT4SHMDvbaSY     lucky           dial SLIP
```

Note that the dialout record shown above can be used to establish a character mode connection as well as a packet mode connection.

If everything looks OK, save the current configuration to disk.

```
Jade:Top> save
```

At this point you are halfway finished. You must set up a dynamic interface on the remote NetBlazer to complete the job.

Setting Up a Dynamic Interface at the Remote NetBlazer

After setting a dynamic interface on the local NetBlazer, you must do the same thing on the remote NetBlazer. If the remote NetBlazer is installed and ready to accept calls, you can use the **dial** command to call the remote NetBlazer and run **ipdial** from your local NetBlazer as described below.

```
Jade:Top> dial Ruby
Trying Ruby(9-555-6789):dialout on line10 ... connected

Type ^Z (decimal 26) <CR> to return to NetBlazer
Connected to Ruby


Telebit's NetBlazer Version 1.5

Ruby login: root
Password:
Ruby:Top>
```

When a character mode connection is established, log in as root or some other user with system configuration privileges and start **ipdial**. When you are prompted for the name of remote (other) system, enter the hostname of your local NetBlazer (i.e. Jade in our example).

```
Ruby:Top> ipdial
Name of remote (other) system: Jade
Allow dial-out from this interface [y]?
Allow dial-in to this interface [y]?
```

If you allowed dial-out calls on the local interface, then you must allow dial-in calls on this interface. Likewise, if you allowed dial-in calls on the local interface, then it only makes sense to allow dial-out calls from this interface.

Next, set up the dynamic interface record on the remote system. Entering the correct password and/or crypto key may be a little confusing. The password and/or crypto key for the *interface record on the remote NetBlazer* must be identical to the password and/or cypto key entered for the *user record on the local NetBlazer*.

```
Begin adding interface 'Jade' ...
User ID to send to remote system [Ruby]:
Want to enter a password [y]?
Password to send to remote system (no echo):
Re-enter the password:
Will remote system issue a crypto handshake [n]? y
Enter the crypto key (no echo):
Re-enter the crypto key:
Chat script [ics]:
SLIP or PPP (s|p) [s]:
Maximum lines to connect to remote system [1]:
What is the Maximum Receive Unit (MRU) [1006]:
Add dynamic interface=Jade, user=Ruby, chat=ics, using SLIP
    with crypto handshaking, max-lines=1, MRU=1006
Okay (yes|no|quit) [y]?
```

Enter the dialout records used to call the local NetBlazer.

```
Begin adding dial-out 'Jade' ...
Phone number: 9-555-9012
Line characteristics (dialout):
Dialout: name=Jade, phone=9-555-9012, characteristics=dialout
Okay (yes|no|quit) [y]?
Add another dial-out [n]? y
Phone number: 9-555-6678
Line characteristics (dialout):
Dialout: name=Jade, phone=9-555-6678, characteristics=dialout
Okay (yes|no|quit) [y]?
Add another dial-out [n]?
```

Next, enter the user record information that is used by the remote NetBlazer to authenticate a call from the local NetBlazer. This time, the password and/or crypto key for the *user record on the remote NetBlazer* must match the password and/or crypto key entered for the *interface record on the local NetBlazer*.

```
Please add the information that the remote system 'Jade'
        will use to contact this NetBlazer
Want to enter a password [y]?
Enter the password (no echo):
Re-enter the password:
Issue a crypto handshake [n]? y
Enter the crypto key (no echo):
Re-enter the crypto key:
User: Jade              privs:     dest-group:
Okay (yes|no|quit) [y]?
```

Finally, add the routes to the remote NetBlazer that are used to reach the networks located at the local site.

```
Add routes for interface [y]?
Begin adding route thru 'Jade' ...
Destination name or address/bits (Return to finish routes): default
Metric [1]:
Private [n]:
Add route destination=0.0.0.0, bits=0, metric=1, not private
Okay (yes|no|quit) [y]?
Route added
```

Using a default route, as shown in the above example, allows the hosts at the remote site to access all or the networks that can be reached from the local side without knowing the specific layout of the local networks. However, remember that only one default route should be used. Therefore if the remote network has multiple paths to the outside, using a default route for this interface may not be the best choice.

When you are done adding routes, press the return key at the "Destination name or address/bits" prompt to exit the **ipdial** dialog.

```
Begin adding route thru 'Jade' ...
Destination name or address/bits (Return to finish routes):
... Done adding dial IP
For proper communications, remote system should have:
1) Dynamic interface=Ruby, user=Jade, chat=ics, using SLIP
   with crypto handshaking, max-lines=1, MTU<=1006
2) Dialout=Ruby
3) User=Ruby
4) Route destination=143.191.64.254, interface=Ruby, gateway=none, metric=1
```

Verify that the remote network is set up correctly, and save the current configuration.

```
Ruby:Top> interface list Jade
Name              IP address   MRU   MTU type    line      pkts in   pkts out
Jade           143.191.64.254  1006  1006 Dialout Down           0          0
Idle: In 9:09:23:54   Out 9:09:23:54                 IP:         0          0
Will save 'Jade' vj: auto
Current vj: not active
Dynamic SLIP interface, Uname=Ruby, Password=lucky
Dial-out=Yes, Dial-in=Yes
Chat script=ics, Crypto key=lucky, Max-lines=1, MRU=1006, MTU=1006
Trigger level=2, Timeout=180 s, Queue-max=8
cost = 0        bytespertick = 0

Ruby:Top> route list |grep Jade
default         0   Jade                          1         0      49

Ruby:Top> dialout list |grep Jade
Jade            9-555-9012         dialout
Jade            9-555-6678         dialout

Ruby:Top> user list |grep Jade
Jade        ab0tmXfH1a/DI     scarlet         dial SLIP

Ruby:Top> save
```

Log out of the current character mode session.

```
Ruby:Top> logout
3
Connection terminated
Jade:Top>
```

The interfaces between the two NetBlazers (Jade and Ruby in the example) are now ready to be used for packet mode connections. You can confirm this by running a telnet session from the local to remote site.

Setting Up Auto Routes to a Remote Site

If you want the local NetBlazer's routing table to include routes to the remote site only when a dial-in connection is established from the remote site, use the **dynamic auto-route** command to assign one or more auto routes to a dynamic interface instead of adding static routes to the routing table.

To illustrate how auto routes are used, assume that you want the remote site, shown in Figure 5-2, only to be accessible when a dial-in connection has been established from the remote site through the dynamic interface named Ruby.

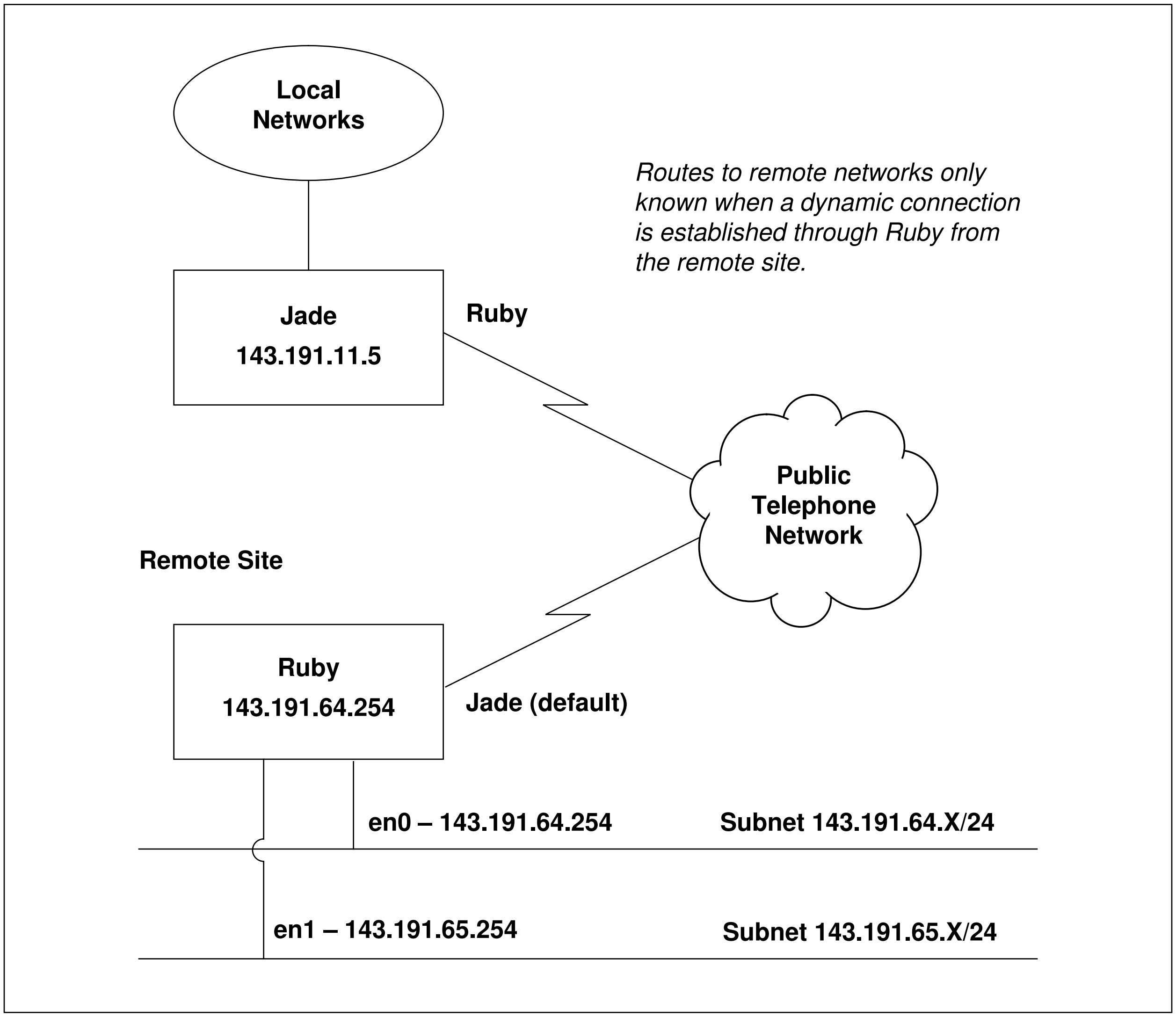


Figure 5-2. Auto Route Example

First, use **ipdial** to create dynamic interfaces for Ruby and Jade as described earlier in this chapter. *Do not enter any static routes through Ruby on the local NetBlazer named Jade.*

Next, go to the Dynamic command level on Jade, and added auto routes for the networks at the remote site to the dynamic interface named Ruby.

```
Jade:Top> dynamic
Jade:Top>Configure>Interface>Dynamic> auto-route Ruby add 143.191.64/24
Jade:Top>Configure>Interface>Dynamic> auto-route Ruby add 143.191.65/24
Jade:Top>Configure>Interface>Dynamic> list Ruby
Name               IP address   MRU   MTU type    line      pkts in   pkts out
Ruby             143.191.11.5  1006  1006 Dialout Down        3872       3879
Idle: In 0:05:56:18   Out 0:05:56:18                IP:       3872       3879
Auto IP's :143.191.64.0/24     143.191.65.0/24
Will save 'Ruby' vj: auto
Current vj: not active
Dynamic SLIP interface, Uname=Jade, Password=scarlet
Dial-out=Yes, Dial-in=Yes
Chat script=ics, Crypto key=scarlet, Max-lines=1, MRU=1006, MTU=1006
Trigger level=2, Timeout=180 s, Queue-max=8
cost = 0        bytespertick = 9
```

At this point, the local NetBlazer named Jade should not have any routes to the remote site through the interfaced named Ruby. You can confirm this by entering the following **route list** command.

```
Jade:Top>Configure>Interface>Dynamic> route list |grep Ruby
Jade:Top>Configure>Interface>Dynamic>
```

When a dynamic interface connection is initiated from the remote site, the local NetBlazer (Jade) automatically adds temporary routes to the remote networks to its routing table as shown below.

```
Jade:Top>Configure>Interface>Dynamic> route list |grep Ruby
143.191.65.0    24  Ruby                           1       A 0      0
143.191.64.0    24  Ruby                           1       A 0      0
```

If Jade is advertising its routes using RIP, these routes to the remote site are learned by the rest of the local network through RIP.

When the interface named Ruby is allowed to timeout and disconnects, Jade discards the temporary routes, and the remote site is no longer accessible to the local network.

Synchronous Interface Configuration 6

When the NetBlazer is turned on, it automatically creates a synchronous interface named syn0. In most cases, the synchronous interface is connected to a DSU/CSU. However, it can be used with an ISDN terminal adapter. Using the synchronous interface with a terminal adapter requires a special configure, which is described under Using the Synchronous Interface with an ISDN Terminal Adapter on page 6-6.

Note: *If you attach a DSU/CSU or an ISDN terminal adapter to the NetBlazer after it is booted, you must issue an* **interface start syn0** *command before the syn0 interface can be used.*

Before using the synchronous interface, both ends of the synchronous link must be configured to route packets to the network(s) on the opposite end of the link.

Routing IP Packets through the Synchronous Interface

To configure the NetBlazer to route IP packets to the network(s) at the remote end of the link perform the following steps. A similar procedure must be perform at the other end of the link in order to allow data to travel in both directions across the synchronous interface.

1. Log into the NetBlazer as root or some other user with configuration privileges.
2. Go to the Configure Interface command level, and verify that the syn0 interface exists and that it has been assigned a valid IP address.

```
Pearl:Top> interface
Pearl:Top>Configure>Interface> list syn0
Name               IP address   MRU   MTU type    line      pkts in   pkts out
syn0             143.191.63.1  1500  1500 PPP     Up            826        921

•
•
•
```

The syn0 interface is automatically assigned the global IP address of the NetBlazer. If the syn0 IP address is 0.0.0.0, you probably need to set the NetBlazer's global IP address by entering an **ip address** <address> command.

If you get an “Interface syn0 unknown” message in response to the **interface list syn0** command, the NetBlazer is not recognizing the synchronous interface board. Open up the top cover and make sure that the synchronous interface board is configured correctly and securely seated in its edge connector.

3. Add a route for the remote end of the synchronous link as illustrated below.

```
Pearl:Top>Configure>Interface> route
Pearl:Top>Configure>IP>Route> add 143.191.65/24 syn0
Pearl:Top>Configure>IP>Route> list |grep syn0
143.191.65.0    24  syn0                          1         0      3
```

The above example adds a route to network 143.191.65.X through the syn0 interface.

4. If the syn0 interface will be using RIP to send routing information to the remote end, add the IP address of the NetBlazer at the other end of the synchronous link to the RIP address table as shown below.

```
Pearl:Top>Configure>IP>Route> rip
Pearl:Top>Configure>IP>RIP> add 143.191.65.1 30 yes
Pearl:Top>Configure>IP>RIP> list
RIP: sent 458 rcvd 875 reqst 0 resp 875 unk 0 refused 0
Active RIP output interfaces:
Dest Addr       Interval Split
143.191.65.1    30       1
143.191.63.255  30       1
```

This allows the NetBlazer to send RIP information to IP address 143.191.65.1 in 30 second intervals.

Note that the NetBlazer at the remote end of the synchronous link must also have a high speed synchronous interface configured to route IP packets to this NetBlazer over the syn0 interface.

Setting Up a Dynamic Backup for the Synchronous Interface

You can configure the NetBlazer to monitor the synchronous interface, and if the link is dropped, automatically establish a dynamic link and redirect the synchronous link traffic over the dynamic link.

When the synchronous interface (syn0) goes down, and the NetBlazer switches to the backup interface, all routes using syn0 are changed to the backup interface. The NetBlazer continues to test the synchronous link, and when the link comes back, the NetBlazer starts using the synchronous link again. The dynamic interface connection is dropped when the interface times out.

To set up a dial-up backup for the synchronous interface as shown in Figure 6-1, perform the following steps:

1. Use the **ipdial** command to create a dynamic interface on the local end of the link. *Do not assign any routes to the dynamic interfaces.*

   See Chapter 5, Dynamic Interface Configuration, for instructions on how to set up a dynamic interface.

2. Set up the sync backup and watchdog parameters for the syn0 interface as illustrated below.

```
Pearl:Top> interface
Pearl:Top>Configure>Interface> sync-backup Opal
Pearl:Top>Configure>Interface> watchdog syn0 on 180
```

   This example configures the NetBlazer to test the synchronous link every 180 seconds. If the NetBlazer determines that the synchronous link has gone down, it automatically calls the remote side of the link using the dynamic interface named Opal, and sends the synchronous link traffic through the dynamic interface until the synchronous link is returned to service.

3. Repeat steps 1 and 2 above for the remote end of the link.

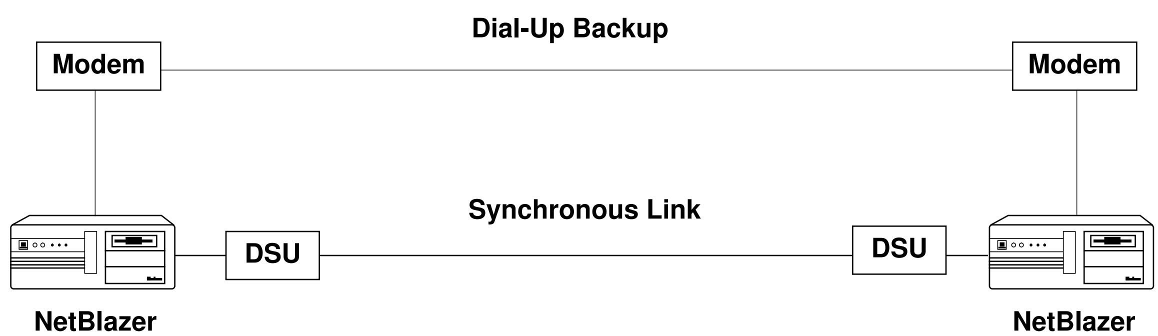


Figure 6-1. Dial-Up Backup for Synchronous Interface

Controlling the Synchronous Interface

The NetBlazer uses PPP (Point-to-Point Protocol) to control data transmissions over the synchronous interface. When the NetBlazers at both ends of the serial link are powered up, a PPP connection is automatically negotiated which allows IP packets to travel across the link.

If you need to temporarily stop IP packet transmissions through the syn0 interface, you can issue a **ppp close syn0** command. This places the local side of the PPP link in a closed state as shown below.

```
Pearl:Top>Configure>PPP> close syn0
Pearl:Top>Configure>PPP> list syn0
Name              IP address   MRU   MTU type    line      pkts in   pkts out
syn0            143.191.63.1  1500  1500 PPP     Down        11385      15292
Backed up by: Opal            Watchdog: on               interval: 60

Will save 'syn0' vj: off
Current vj: IN direction off
            OUT direction off

Termination Phase
    838856 In          0 Flags      0 ME       0 FE        0 CSE     62 other
                      49 Lcp        0 Pap     34 IPcp      1 Unknown
   4144771 Out         0 Flags      0 ME     628 Fail
                     173 Lcp        0 Pap     34 IPcp
LCP Closed
                 MRU     ACCM            AP      PFC  ACFC Magic
        Local:   1500    0x00000000     *None    Yes  Yes  0x0f6b17fd
        Remote:  1500    0xffffffff     *None    No   No   0x0862df06
IPCP Listening -- waiting for remote host to attempt open
        local IP address: 143.191.63.1  remote IP address: 0.0.0.0
syn0        11434 Rxi             4 over           44 CSE      185 abort
                0 unexpct         3 spurious        0 nobuf      0 dropped
V.35        15315 Txi             4 underrun     6649 split      0 queued
                0 hard reset      0 soft reset  [15377,255 Esi]
```

To reactivate the link, you must issue a **ppp activeopen syn0** command. Entering **list syn0** shows that the link is again open.

```
Pearl:Top>Configure>PPP> activeopen syn0
Pearl:Top>Configure>PPP> list syn0
Name              IP address   MRU   MTU type    line      pkts in   pkts out
syn0            143.191.63.1  1500  1500 PPP     Up          11505      15412
Backed up by: Opal            Watchdog: on               interval: 60

Will save 'syn0' vj: off
Current vj: IN direction off
            OUT direction off

Network Protocol Phase  (open for 0:00:00:04)
    841436 In          0 Flags      0 ME       0 FE        0 CSE     62 other
                     163 Lcp        0 Pap     36 IPcp      1 Unknown
   4146231 Out         0 Flags      0 ME     995 Fail
                     287 Lcp        0 Pap     36 IPcp
LCP Opened
                 MRU     ACCM            AP      PFC  ACFC Magic
        Local:   1500    0x00000000     *None    Yes  Yes  0x0feac332
        Remote:  1500    0xffffffff     *None   +Yes +Yes +0x0e0fc129
IPCP Opened
        local IP address: 143.191.63.1  remote IP address: 0.0.0.0
syn0        11554 Rxi             4 over           44 CSE      185 abort
                0 unexpct         3 spurious        0 nobuf      0 dropped
V.35        15435 Txi             4 underrun     6765 split      0 queued
                0 hard reset      0 soft reset  [15497,255 Esi]
```

Another way to control the synchronous interface is by using the **interface stop** and **start** commands.

The **interface stop syn0** command turns off the synchronous interface and sets PPP to a passive state.

```
Pearl:Top>Configure>PPP> interface
Pearl:Top>Configure>Interface> stop syn0
Pearl:Top>Configure>Interface> list syn0
Name                 IP address   MRU   MTU type    line      pkts in   pkts out
syn0               143.191.63.1  1500  1500 PPP     Down        11678      15593
Backed up by: Opal            Watchdog: on               interval: 60

Will save 'syn0' vj: off
Current vj: IN direction off
            OUT direction off

Physical Line Dead
    854530 In          0 Flags      0 ME       0 FE        0 CSE     62 other
                     163 Lcp        0 Pap     36 IPcp       1 Unknown
   4165417 Out          0 Flags      0 ME    1007 Fail
                     287 Lcp        0 Pap     36 IPcp
LCP Listening -- waiting for remote host to attempt open
                 MRU     ACCM            AP      PFC  ACFC Magic
        Local:   1500    0x00000000     *None    Yes  Yes  0x106ac465
        Remote:  1500    0xffffffff     *None    No   No   0x0e0fc129
IPCP Listening -- waiting for remote host to attempt open
        local IP address: 143.191.63.1  remote IP address: 0.0.0.0
syn0        11727 Rxi              4 over            44 CSE      185 abort
                0 unexpct          3 spurious         0 nobuf      0 dropped
V.35        15616 Txi              4 underrun      6780 split      0 queued
                0 hard reset       0 soft reset  [15678,255 Esi]
```

To restart the interface, enter an **interface start syn0** command.

```
Pearl:Top>Configure>Interface> start syn0
Pearl:Top>Configure>Interface> list syn0
Name                 IP address   MRU   MTU type    line      pkts in   pkts out
syn0               143.191.63.1  1500  1500 PPP     Up          11718      15634
Backed up by: Opal            Watchdog: on               interval: 60

Will save 'syn0' vj: off
Current vj: IN direction off
            OUT direction off

Network Protocol Phase  (open for 0:00:01:10)
    858062 In          0 Flags      0 ME       0 FE        0 CSE     62 other
                     165 Lcp        0 Pap     38 IPcp       1 Unknown
   4169927 Out          0 Flags      0 ME    1430 Fail
                     289 Lcp        0 Pap     38 IPcp
LCP Opened
                 MRU     ACCM            AP      PFC  ACFC Magic
        Local:   1500    0x00000000     *None    Yes  Yes  0x10eba846
        Remote:  1500    0xffffffff     *None   +Yes +Yes +0x14c017f9
IPCP Opened
        local IP address: 143.191.63.1  remote IP address: 0.0.0.0
syn0        11767 Rxi              4 over            44 CSE      185 abort
                0 unexpct          3 spurious         0 nobuf      0 dropped
V.35        15657 Txi              4 underrun      6789 split      0 queued
                0 hard reset       0 soft reset  [15719,255 Esi]
```

Using the Synchronous Interface with an ISDN Terminal Adapter

To configure the synchronous interface line for use with an ISDN terminal adapter, perform the following steps:

1. Make sure that the synchronous interface board is configure for RS-232 operation, and that a terminal adapter which is supported by the NetBlazer is connected to the synchronous interface board.

   Refer to the *NetBlazer Installation Guide* for instructions on setting up the synchronous interface board for RS-232 operation. Contact Telebit's Technical Support for a current list of supported terminal adapters.

2. Go to the Configure Line command level, and set up the syn0 line to use the modem type ta and the line characteristic group v25dialout.

```
Pearl:Top>Configure>Line> modem syn0 ta
Pearl:Top>Configure>Line> characteristics syn0 v25dialout
Pearl:Top>Configure>Line> list
Line     Modem Type Char's   Use        Command   Flags
line00   <<None>>   <<None>> login      <<None>>   5200 <AF,CON,BSY>
line01   <<None>>   <<None>> login      <<None>>   5200 <AF,CON,BSY>
syn0     ta         v25dialo <<None>>   <<None>>   4008 <DSR,BSY>
```

3. Start the v25bis server using the **server start** command.

```
Pearl:Top>Configure>Line> server start v25bis
```

   Now any dialout record that is assigned the characteristic v25dialout will use the syn0 line to attempt the connection.

4. To use the syn0 interface for dynamic packet mode connections with a remote site, set up a dynamic interface as described in Chapter 5, Dynamic Interface Configuration. *When you are prompted for the dialout record's line characteristics, enter **v25dialout**.*

5. If you want to back up a dynamic interface that uses the syn0 line, set up a watchdog and backup interface as illustrated below.

```
Pearl:Top> interface
Pearl:Top>Configure>Interface> watchdog Opal 143.191.63.1 180
Pearl:Top>Configure>Interface> sync-backup Opal.backup
```

   In this example the watchdog checks the dynamic interface, named Opal, by sending ICMP echo requests to 143.191.63.1 every 180 seconds of idle time. If the watchdog determines that the syn0 line is down, the NetBlazer redirects Opal's traffic through the backup interface name Opal.backup until the syn0 link returns.

6. The NetBlazer allows you to screen incoming calls by comparing the caller’s phone number, which is detected by the terminal adapter, to a list of permitted or rejected telephone numbers. If you want the NetBlazer to screen incoming calls, refer to the **phonelist** command description in your *NetBlazer Command Reference Manual* for information on setting up the phone number list.

Asynchronous Lines Usage 7

This chapter provides information on using the NetBlazer's asynchronous lines. It overviews the procedures for setting up line characteristics, dedicated SLIP or PPP links, and setting up and controlling a raw TCP link. It also covers setting up an auto telnet or rlogin line for automatic login to a remote host, and setting up callback security over an asynchronous line.

Controlling the Asynchronous Lines

When the NetBlazer is turned on or rebooted, it scans the asynchronous lines to automatically detect the type of modem attached to each line, and resets the lines and modems to a known state based on information contained in the MODEMCAP.TXT file. Any line that does not have a modem attached to it is treated as a login terminal port.

If a line's DSR signal is dropped and then raised (e.g. the modem is turned off and then on) or a line is reset, the NetBlazer queries the line again to determine the modem type and autoconfigures the line and modem.

The default line and modem settings are satisfactory for most applications. However, if you have special requirements, a set of commands is provided at the Configure Line command level that allow you to modify the operating characteristics of a line. These commands are summarized in Table 7-1. For a complete description of the **line** commands, refer to the *NetBlazer Command Reference Manual*.

Table 7-1. Line Commands Summary

Command	Function
access	Sets a line to only allow packet mode connections or to allow all connections from the outside. *Default: allow all connections*.
auto	Disables/enables the automatic modem detection and configuration process for a specified line. *Default: enable autoconfiguration.*
callin	Denies/permits incoming calls on a specified line. *Default: permit incoming calls.*
command	Assigns a command, such as telnet, to a line. This command is automatically invoked when a dial-in connection is made to the line.
characteristics	Assigns a characteristic group to a line. This is used by the dial-out routine to select a line by matching the characteristic tags of the dialout record to a line's characteristic tags. *Default: dialout*.

Table 7-1. Line Commands Summary (Continued)

Command	Function
dcdtime	Sets the amount of time a modem must hold DCD low before the NetBlazer drops the connection. *Default: 2 seconds*.
delay	Sets the amount of time the NetBlazer pauses after the modem has accepted a command and returned a result code. *Default: 0 milliseconds*.
dsrtime	Sets the minimum time that a modem must hold DSR on before the NetBlazer assumes the modem is ready. *Default: 3 seconds*.
dtrtime	Sets the time duration that the NetBlazer holds down the DTR signal for a specified line. This is followed by a series to AT commands to bring the modem back to a known state before the NetBlazer uses the modem. *Default: 1100 milliseconds*.
flowcontrol	Sets the flow control method the NetBlazer uses on the device attached to a specified line: CTS-RTS, XON-XOFF, no flow control, or auto switch. *Default: auto flow control*.
idle	Configures a specified line to automatically reset if it is idle for a set number of minutes. *Default: idle reset timer disabled.*
modem	Used to specify the modem type attached to a specific line. (This is normally performed automatically by the NetBlazer whenever the DSR signal is raised.)
speed	Sets the data transfer rate on a specified serial port to a standard rate between 300 and 57600 bps. *Default: 9600 bps if not set by autoconfigure*.
terminal	Sets the attributes of the terminal type attached to a specified line.

Setting Up a Dedicated SLIP or PPP Link

You can reconfigure an asynchronous line to be used as a dedicated SLIP or PPP link for transmitting IP datagrams as illustrated in Figure 7-1.

To set up a dedicated SLIP or PPP link, first install an RS-232 null modem cable between the two devices on each side of the link.

Caution: *If you are using line00 for a dedicated link, issue a* ***setconsole internal*** *command to the NetBlazer to prevent console messages from interferring with the link.*

Next, create an interface that is attached to the asynchronous line using the **interface attach** command as illustrated below.

```
Pearl:Top> interface
Pearl:Top>Configure>Interface> attach line01 ppp ppp01 1500 143.191.63.253
line01: stopping login and disabling autoconfigure
line01: interface=ppp01, protocol=PPP, MRU=1500, ip address=143.191.63.253
```

In the previous **attach** command, line01 is the name of the line that is used for the dedicated link, ppp (or slip) specifies the serial link protocol to use, ppp01 is the name of the interface attached to the line, 1500 sets the MRU (maximum receive unit) of the interface, and 143.191.63.253 is the IP address assigned to the interface.

Before you can use a dedicated link, you must set up a route to the remote end of the link.

```
Pearl:Top>Configure>Interface> route add 143.191.65/24 ppp01
```

If the dedicated link is using the PPP protocol, you must activate the link by entering a **ppp activeopen** <interface> command.

```
Pearl:Top>Configure>Interface> ppp activeopen ppp01
```

The other side of the link must also be active and have a valid route across the link.

For more information on controlling a PPP interface, see the PPP command description in the *NetBlazer Command Reference Manual*.

To set up a dedicated SLIP link, follow the same procedure just described for a PPP link, except you do not need to activate the link by issuing a **ppp activeopen** command.

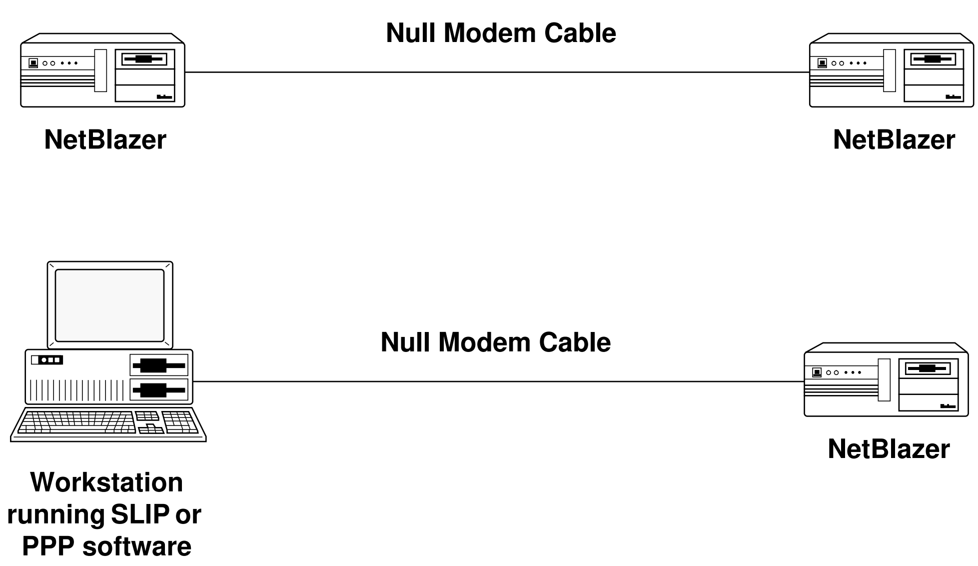


Figure 7-1. Dedicated SLIP or PPP Link

Setting Up a Raw TCP Link

The raw server allows you to establish a raw TCP connection from a host on the network to the modem or other asynchronous device attach to a NetBlazer line. Raw data from the host is transferred directly to the device attached to a line that has been set up for a raw connection. Data from a raw connection is encapsulated in TCP/IP packets and sent back to the host on the network.

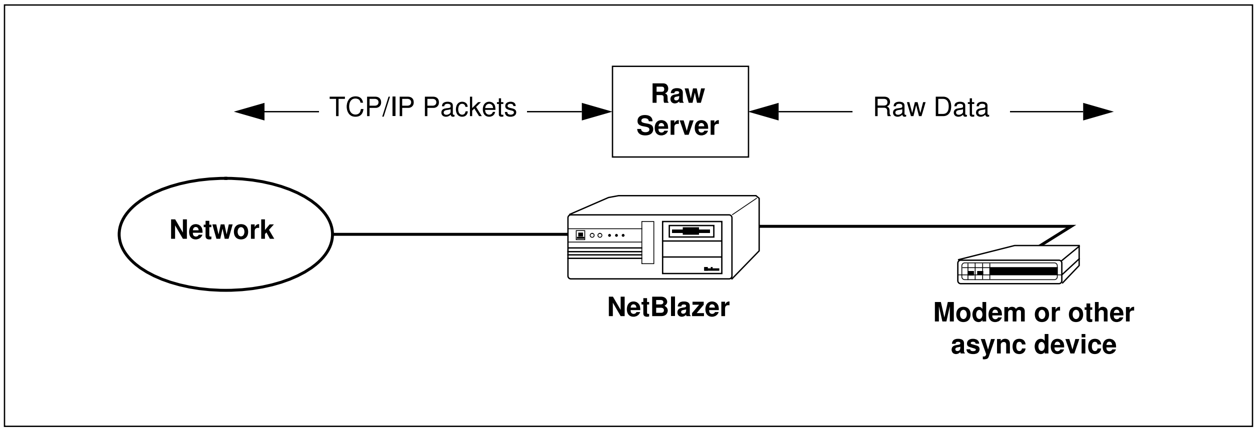


Figure 7-2. Raw TCP Connection

Configuring a Line for a Raw Connection

To configure a line for a raw connection, you must assign the line a characteristic group that includes the raw characteristic tag. Two characteristic groups provided by Telebit include the raw characteristic tag: raw and raw_dial. The raw characteristic group only allows raw connections on the line. The raw_dial characteristic group allows both raw connections and dialout connections if a modem is attached to the line.

Go to the Configure Line command level, and assign the raw or raw_dial characteristic group to the line that you want to use for a raw connection.

```
Pearl:Top> line
Pearl:Top>Configure>Line> characteristic line01 raw_dial
```

Assigning a line to a characteristic group with the raw characteristic tag automatically assigns an IP address and port number to the line unless they are already set. The defaults are the NetBlazer’s global IP address and a port number equal to 7000 plus the line number (e.g. line01 is assigned port number 7001).

To change the line's IP address or port number use the **ip address** <ip-addr> <line> <port> command as shown in the following example.

```
Pearl:Top>Configure>Line> ip address 143.191.10.28 line01 543
```

Starting the Raw Server

To use lines with the raw characteristic for raw data connections, you must start the raw server.

```
Pearl:Top>Configure>Line> server start raw
```

One process is started for each line that is assigned the raw characteristic.

Caution:If you use a program that pushes a large amount of data through the raw connection to the NetBlazer and then exits, it will close the socket before the NetBlazer can transfer all the data to the line. These programs must wait a sufficent amount of time to let the NetBlazer drain the socket.

Stopping a Raw Connection

To stop all raw services, enter a **server stop raw** command.

```
Pearl:Top> server stop raw
```

To stop the raw service on an individual line, assign the line to a characteristic group which does not contain the raw characteristic tag.

```
Pearl:Top> line
Pearl:Top>Configure>Line> characteristic line01 dialout
```

This automaticallly stops the raw server listening for that line.

Stopping the raw server in either of these manners does not stop raw sessions in progress. To interrupt a raw session in progress, use the **line reset** command for the line you wish to stop.

Setting Up a Line to Automatically Connect to a Host

You can configure one or more lines to automatically establish a telnet or rlogin session with a specific host whenever a call is recieved on the line. This allows a dial-in user to connect directly with a designated host without logging in to the NetBlazer.

The following example shows you how to assign a telnet command to a line using the **line command** command.

```
Jade:Top> line
Jade:Top>Configure>Line> command line11 telnet napa
```

In this example, any call received on line11 is automatically passed through to the host named napa. When the caller completes the telnet session with napa and logs out, the line drops the telephone connection, and waits for another call to connect to napa.

To remove the command assigned to line11, enter a **line command line11** command with no command and argument parameters.

Setting Up Call Back Security

The NetBlazer can be configured to call back a remote user at a predetermined destination. When call back security is used, the remote user must call the NetBlazer at the telephone number associated with the call back request line. The NetBlazer would prompt the caller for the dialout destination to call back, then disconnect the original call and call the user back at the user specified destination.

Setting Up a Call Back Request Line

To set up the NetBlazer to use call back security, you need to configure one or more of the asynchronous lines as a call back request line. No one can log in on a line which has been designated for processing call back requests. The following example shows you how to set up line10 as a call back request line.

```
Jade:Top> line
Jade:Top>Configure>Line> request-call line10 on
Request-callbacks on line 'line10' is on
```

This is the line that remote users should use to call into the NetBlazer to request a call back. When a call is received on line10 in the above example, the NetBlazer prompts the caller for a destination. The destination entered must exist in the dialout list. Therefore, you must create a dialout record for each site that uses call back security. See Setting Up Dialout Destination Records in Chapter 8.

Call Back with Auto-Telnet

This section describes how to configure the NetBlazer to automatically establish a telnet session with a predetermined host system when calling back a specified destination site.

First set up a call back request line as described in the previous section, then configure one or more lines to automatically telnet to a specific host.

```
Jade:Top>Configure>Line> group add napa
Jade:Top>Configure>Line> group modify napa add napa
Jade:Top>Configure>Line> characteristics line11 napa
Jade:Top>Configure>Line> command line11 telnet napa
```

The above example creates a new characteristics group named napa, then assigns the group to line11 and configures the line to automatically connect with the host system named napa.

Next, turn off the callin parameter for each line that is used to call back a specified destination site as shown below.

```
Jade:Top>Configure>Line> callin line11 off
Callin on line 'line11' is off
```

This prevents a user from calling directly into the host system without going through the call back security process.

Finally, create dialout records for each site that can be called back from the NetBlazer. The following example shows you how to set up a dialout record for a remote destination site named saturn.

```
Jade:Top>Configure>Line> dialout add saturn 555-8912 napa
```

When a user calls into the NetBlazer on the call back request line and enters "saturn" as the call back destination site, the NetBlazer looks up saturn in its dialout list, selects a line that matches saturn's characteristics group (i.e. napa), and calls the user back at the telephone number specified for saturn. In this example, the NetBlazer would use line11 to call back saturn. Since this line is configured to automatically telnet to napa , the user is connected directly with napa when the connection is established.

Dialout Destinations 8

Using the NetBlazer as a modem server allows users on the IP network to log into the NetBlazer through a telnet session and establish a serial character mode connection with a remote computer.

Each remote destination that can be called through the NetBlazer must be defined in the dialout list. The dialout records include the following information, which is used to establish a connection with the remote system:

- **Dialout Name** – This is the named assign to the dialout record. It is used by other other routines in the NetBlazer to select the dialout record that contains the information needed to call the remote site.
- **Phone Number** – This is the telephone number used to call the remote system.
- **Characteristics Group** – This is the name of the characteristic group that is assigned to the dialout record. Information contained in the characteristics group record is used to select and preconfigure a modem for calling the remote system.
- **Character Mode Timeout** – This field determines how long a character mode connection is allowed to be idle before disconnecting the call. Packet mode connections use a different timeout, which is set in the dynamic interface record.

The remainder of this chapter is divided into two parts. The first part, Setting Up Dialout Destination Records, shows you how to set up one or more dialout records for a remote site. The second part, Using Characteristics Groups on page 8-4, explains how to use the characteristics group function to select a line and preconfigure the modem for a dialout connection.

Setting Up Dialout Destination Records

Go to the Configure Dialout command level and enter "?" to display a list of commands used to maintain the list of dialout destinations used by the NetBlazer.

```
Top> dialout
Top>Configure>Dialout> ?
Available commands:
add             characteristics delete          ditto
list            phone           timeout         top>
up>             ?
```

Adding a Dialout Destination

To add an entry to to the dialout destination list, enter an **add** command at the Configure Dialout command level.

```
Jade:Top>Configure>Dialout> add
```

The NetBlazer prompts you for the required information:

```
Name of dialout: databank
```

The destination name entered is used by other routines in the NetBlazer to select this record. If the destination is another NetBlazer with more than one modem connected, enter the same destination name for each telephone line connected to the remote NetBlazer.

```
Phone number: 555-9876
```

Enter the telephone number of the remote connection. Refer to the Dial command description in your modem's reference manual for a description of acceptable dial strings.

```
Line characteristics (dialout):
```

This entry is used to select a serial line with matching characteristics when placing a call to the dialout destination. If no line is available with the specified characteristics, the call is not made. The name entered must be an existing characteristics group. Press the return key to accept the default characteristics group, dialout.

```
Char mode timeout (in minutes) (60):
```

This entry sets the number of minutes that the NetBlazer maintains an idle character mode connection. Enter the number of minutes or press return to accept the default value of 60 minutes.

```
Dialout: name=databank, phone=555-9876, characteristics=dialout
Okay (yes|no|quit) [y]?
```

If the dialout record that you just entered is correct, press the return key or enter "y" to accept the record. If you want to discard this record and enter a new record, enter "n" at the prompt. If you want to discard this record and return to the NetBlazer's command prompt enter "q" to quit.

Displaying the Dialout Destinations

Enter **list** at the Configure Dialout command level to display a list of dialout destinations:

```
Jade:Top>Configure>Dialout> list
Name            Phone               Characteristics
tbupdate        1-408-745-3700      dialout
Ruby            9-555-6789          dialout
Ruby            9-555-4321          dialout
databank        555-9876            dialout
```

Notice that the NetBlazer now has a new dialout destination named databank. This destination can be reached through a **session dial** command, or through a **session connect** command if the destination is included in the user's destination group as described in Chapter 9, User IDs and Destination Groups.

The two entries for Ruby were created for a dynamic interface using the **ipdial** command in an earlier example. However, these entries can also be used to establish a character mode connection with the destination named Ruby.

Changing Dialout Destination Parameters

You can change the phone number, characteristics group and/or character mode timeout value assigned to a particular dialout destination.

The **dialout phone** <dialout> [#<item-number>] <phone-number> command is used to change the phone number assigned to the specified dialout destination.

The **dialout characteristics** <dialout> [#<item-number>] <group> command is used to change the characteristics group assigned to the specified dialout destination. See Using Characteristics Groups on page 8-4 for additional information.

The **dialout timeout** <dialout> [#<item-number>] <timeout (minutes)> command is used to change the timeout value assigned to the specified dialout destination.

The optional [#<item-number>] parameter in each of the above commands is used to specify which dialout destination to change when more than one entry share the same dialout destination name.

Removing a Dialout Destination

The **dialout delete** <dialout> [#<item-number>] command removes a dialout record.

The <dialout> parameter specifies the name of the dialout record, and the #<item-number> parameter determines which record to remove when there is more than one entry with the specified dialout name.

The following example shows you how to delete the first entry for the Ruby dialout destination.

```
Jade:Top>Configure>Dialout> list
Name             Phone              Characteristics
tbupdate         1-408-745-3700     dialout
Ruby             9-555-6789         dialout
Ruby             9-555-4321         dialout
databank         555-9876           dialout
Jade:Top>Configure>Dialout> delete Ruby #1
deleted dialout 'Ruby'
Jade:Top>Configure>Dialout> list
Name             Phone              Characteristics
tbupdate         1-408-745-3700     dialout
Ruby             9-555-4321         dialout
databank         555-9876           dialout
```

If you do not enter the #<item-number> parameter for a dialout destination with more than one entry, the NetBlazer will prompt you for the item number.

Using Characteristics Groups

The NetBlazer uses the characteristic group records assigned to the lines and the dialout records along with information contained in the MODEMCAP.TXT file to select and preconfigure a modem before calling a remote site.

Preconfiguring a Modem for a Dialout Destination

Each characteristics group record contains one or more user-defined tags. The NetBlazer selects a line and preconfigures the modem by matching the user-defined tags in the characteristics group that is assigned to a dialout record with the characteristic tags contained in the line’s characteristic group or in the MODEMCAP.TXT file.

Displaying a list of the characteristics groups shows seven groups that are provided by Telebit.

```
Jade:Top> group
Jade:Top>Configure>Group> list
dialout          v32dialout       request          pepdialout       raw
raw_dial         v25dialout
```

The dialout group is the default group, which is assigned to each line with a modem attached when the Netblazer is first powered up. Listing the dialout group shows that this group has one characteristic tag, i.e. dialout.

```
Jade:Top>Configure>Group> list dialout
dialout:
 0: dialout
```

Assigning the dialout characteristic group to a dialout record allows the NetBlazer to use any line that has a dialout characteristic group assigned to it when placing a call to the destination specified by the dialout record.This also allows the modem to use its default configuration, which is defined by the MODEMCAP.TXT file.

Using the default modem configuration is preferred in most cases because it allows the modem to negotiate the operating mode with the modem at the remote site.

However, suppose that you want to force the modem to always use the V.32 transmission mode when calling a particular dialout destination. Listing the v32dialout characteristic group shows that this group has two characteristic tags assigned to it.

```
Jade:Top>Configure>Group> list v32dialout
v32dialout:
 0: dialout          1: v32
```

The v32 characteristic tag is a special tag defined in the MODEMCAP.TXT file. When the NetBlazer detects this tag while using a dialout record to place a call, it selects a modem that supports V.32 mode and preconfigures the modem for V.32 mode.

The following example shows you how change the characteristic group assigned to an existing dialout record, named Ruby.

```
Jade:Top>Configure>Group> dialout
Jade:Top>Configure>Dialout> characteristics Ruby v32dialout
  1: Ruby            9-555-6789         dialout
  2: Ruby            9-555-4321         dialout
Enter the number of the dialout to use [2]:
Jade:Top>Configure>Dialout> characteristics Ruby #1 v32dialout
```

Listing the dialout records show that the v32dialout characteristic group is assigned to the two dialout records for Ruby.

```
Jade:Top>Configure>Dialout> list
Name            Phone              Characteristics
tbupdate        1-408-745-3700     dialout
Ruby            9-555-6789         v32dialout
Ruby            9-555-4321         v32dialout
```

Figure 8-1 illustrates how the NetBlazer uses the v32dialout characteristics group to select and preconfigure the modem for V.32 transmission mode by matching the characteristic tags contained in the dialout characteristics group to the characteristic tags contained in the line characteristics group and MODEMCAP.TXT file.

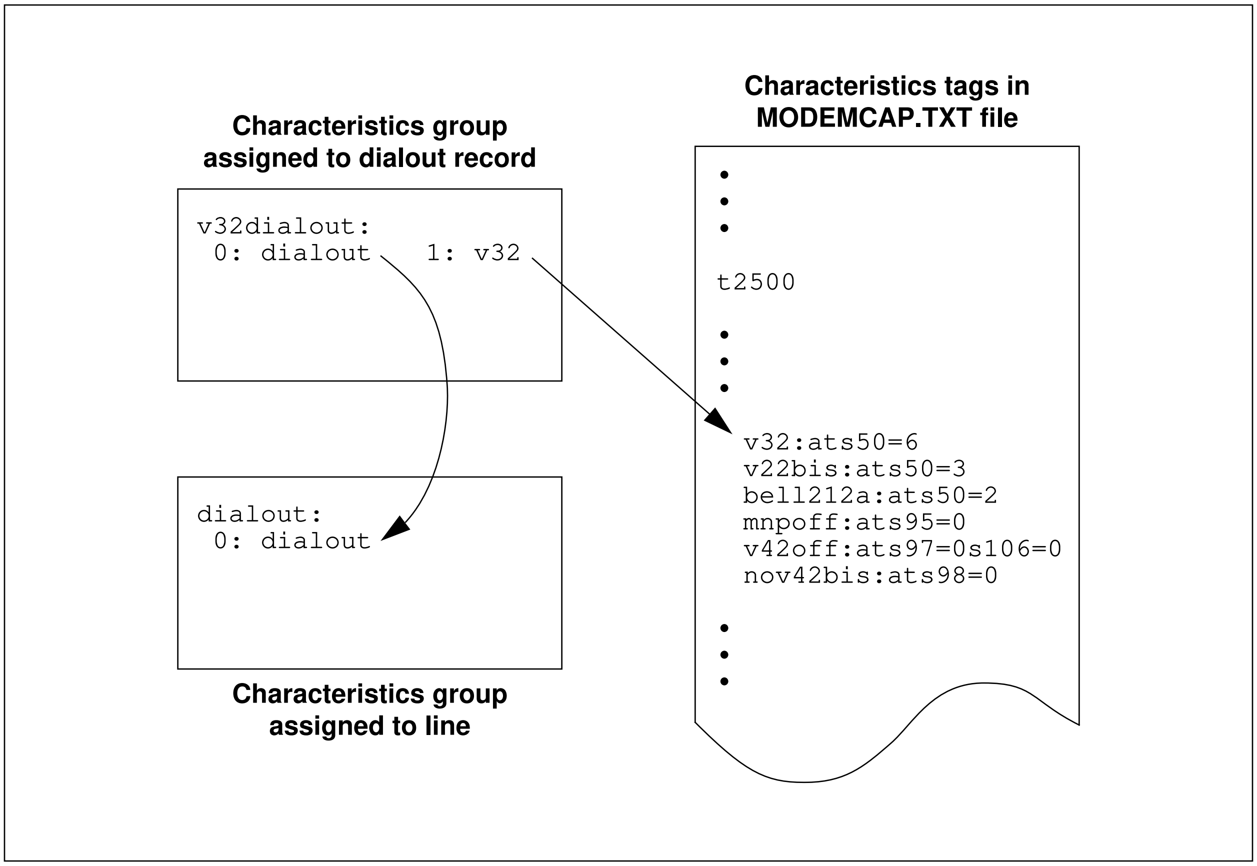


Figure 8-1. Using Characteristic Tags to Preconfigure the Modem

Before placing a call to the destination named Ruby, the NetBlazer selects a line that has a dialout characteristic tag in its assigned characteristics group and has a modem attached that supports V.32 transmission mode, and then sends the v32 command string, defined in the MODEMCAP.TXT file, to the modem.

Restricting the Use of a Line for Specific Dialout Destinations

You can create custom line characteristics groups to restrict the use of one or more serial lines for a select group of dialout destinations. For example, to reserve line10 exclusively for use by the dialout destination named Ruby perform the following steps.

1. Go to the Configure Group command level and create a new group, named reserved, with a characteristic tag named ruby.

```
Jade:Top>Configure>Dialout> group
Jade:Top>Configure>Group> add reserved
Jade:Top>Configure>Group> modify reserved add ruby
```

2. Assign the reserved characteristics group to line10.

```
Jade:Top>Configure>Group> line characteristics line10 reserved
```

This reserves line10 for use by only those dialout records that have a single user-defined characteristic tag named ruby. Any dialout destination that includes a user-defined characteristic tag other than ruby is not allowed to use line10.

3. If you want to use the default modem configuration when placing a call skip this step. Otherwise, create another characteristics group that includes the appropriate MODEMCAP.TXT characteristic tags along with the tag named ruby. For example, to preconfigure the modem for V.32 mode, create another group with the characteristic tags ruby and v32.

```
Jade:Top>Configure>Group> add v32reserved
Jade:Top>Configure>Group> modify v32reserved add ruby
Jade:Top>Configure>Group> modify v32reserved add v32
```

4. Assign the reserved or v32reserved characteristics group to the dialout records for Ruby. The following example shows you how to assign the v32reserved characteristic group to Ruby.

```
Jade:Top>Configure>Group> dialout
Jade:Top>Configure>Dialout> characteristics Ruby #1 v32reserved
Jade:Top>Configure>Dialout> characteristics Ruby #2 v32reserved
```

This allows the NetBlazer to use line10 to call the dialout destination named Ruby in V.32 mode. If you want the dialout record to use the modem's default configuration, use reserved in place of v32reserved in the above example.

User IDs and Destination Groups 9

Every user that logs in to the NetBlazer, either through a telnet session or by calling in from a remote terminal, must have an assigned user ID record. User records may be entered either manually or by downloading a copy of the password file from a UNIX system.

Before downloading or manually creating user ID records, consider setting up destination groups for the users. Doing so has the following advantages:

- Simplifies the user's interaction with the NetBlazer by allowing the user to select a telnet or dial-out destination from a list of destinations assigned to that user.
- Allows you to restrict the user's access to a limited group of telnet hosts and/or dial-out destinations.

Figure 9-1 shows the steps to follow depending on if you are downloading or manually adding user ID records, and if you are assigning the users to destination groups. When deciding which approach is best for you, consider the following points:

- Loading a copy of a UNIX system's password file is a quick and easy way to add a large number of UNIX system users to the NetBlazer's users list. This also allows the UNIX users to log in to the NetBlazer using their same user ID and password.
- If you have one or more groups of UNIX user's that you want to restrict access to a limited set of destinations, consider editing the users' password records and creating destination groups with a matching GID (Group ID) field before loading the password into the NetBlazer.
- If there is only a small group of users that you want to allow direct access to the NetBlazer, you can manually enter the user ID records. However, you may still want to consider setting up destination groups for the users before adding them to the user list.

An alternative to setting up destination groups for users who have only one destination is to create a special alias record that is used to automatically connect the user to his or her destination.

If you decide to assign users to destination groups, refer to the next section, User Destination Groups, for instructions on creating destination groups. For information on how the NetBlazer PASSWD file is used, refer to NetBlazer PASSWD File on page 9-6. For instructions on manually adding user records, see NetBlazer User IDs on page 9-11. For instructions on how to automatically connect a user to a predetermined destination see Automatically Connecting a User to a Specific Destination on page 9-13.

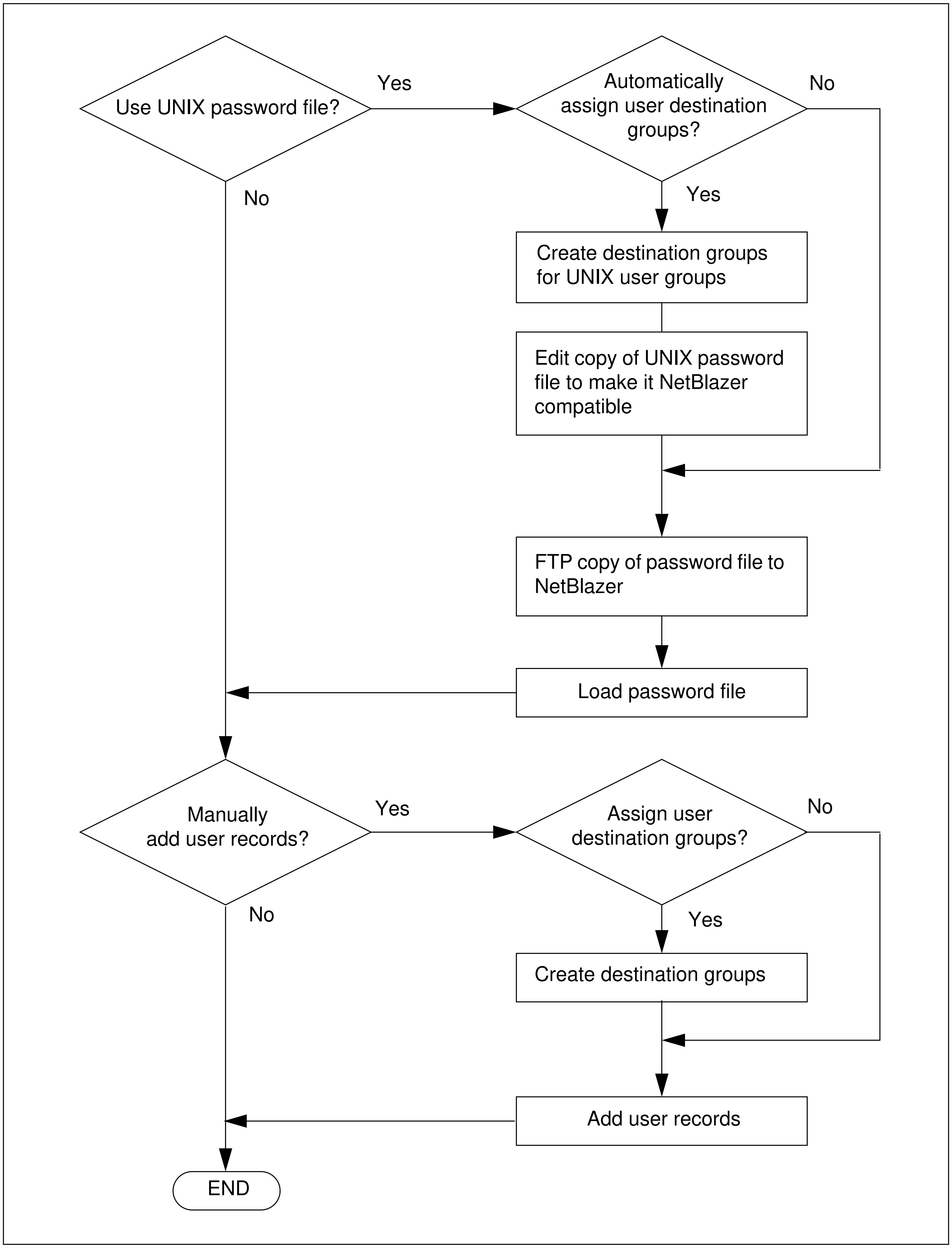
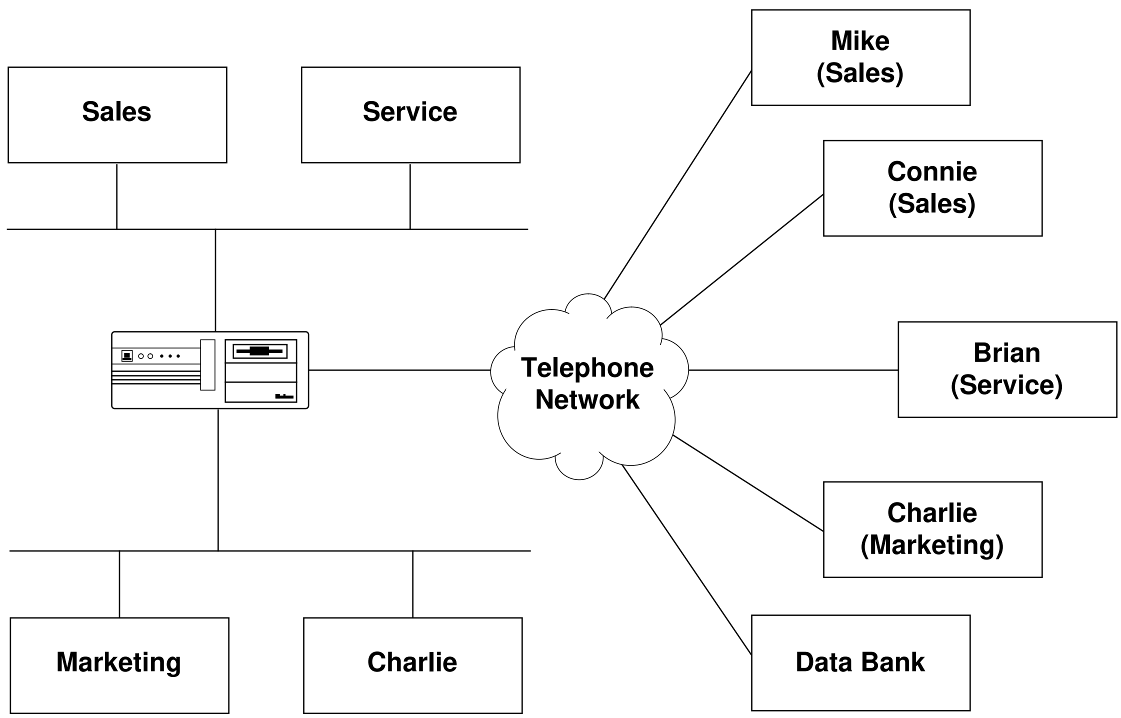


Figure 9-1. Creating User Records Flow Chart

User Destination Groups

Destination groups allow users to easily establish a connection with a selected destination (IP host or remote computer) by simply specifying the desired destination through the **session connect** command.

To illustrate how destinations groups are used, consider the organization shown in Figure 9-2.



Destination Groups

Sales sales (telnet) service (telnet)	Service service (telnet)	**Marketing** marketing (telnet) databank (dialout)

Figure 9-2. Destination Group Example

Mike and Connie are sales people, requiring access to the corporate sales computer via a dial-in connection from a remote site. They sometimes also need to get information from the customer service computer. Creating a sales destination group with two destinations, sales and service, and assigning this destination group to Mike's and Connie's user account, allows them to dial into the NetBlazer and establish a telnet session with the Sales or Customer Service computer by issuing a **connect sales** or **connect service** command to the NetBlazer.

Brian is a field service representative and only needs access to the Customer Service computer at the corporate office. Creating a service destination group with only one destination, service, should meet Brian's requirements.

Charlie works in Marketing, and has a workstation at both the corporate office and at his home. He needs access to the marketing computer from either location. He also occasionally needs to dial out to the data bank computer for additional information. Creating a marketing destination group with two destinations, a telnet connection with the marketing computer, and a dial-out connection with the data bank computer, should satisfy Charlie's requirements.

To set up the marketing destination group, used by Charlie, first create a dial-out record for the data bank computer as described in Chapter 8.

Next, go to the Configure Destination command level and enter **add** at the command prompt.

```
Top> destination
Top>Configure>Destination> add
```

You are prompted for the required information:

```
Please enter the name of the destination group: marketing
Destination group marketing doesn't exist; creating it.
```

This is the destination group name assigned to Charlie and any other users in the marketing group.

```
Enter group id or <CR> to accept default [1]:
```

Enter the GID that you want to use for this destination group or press the return key to accept the suggested default value.

```
Destination number 1.
Enter the destination's short name (press return to end): databank
```

This is the destination name used to establish a dial-out connection to the data bank computer through a **session connect** command.

```
Enter the destination's long name: Data Bank Service
```

This is a descriptive name for the computer accessed through the databank destination.

```
Enter the destination’s type: dial
```

The destination type may be either telnet or dial. Since databank is a dialout destination , enter dial.

```
Enter the dialout destination: databank
```

The dial-out destination entered at this prompt must be created through a **dialout add** command before the destination can be used.

```
Destination 1: shortname: databank  dial string: databank  long name: Data
Bank Service
Do you really want to enter this destination (y/n) [y]?
```

This prompt allows you to verify that the destination information is correct, before accepting the entry. Press the return key or enter “y” to accept the entry.

```
Destination number 2.
Enter the destination's short name (press return to end): marketing
```

This is the destination name which is used to connect to the marketing computer through a **session connect** command.

```
Enter the destination’s long name: Marketing Computer
```

Information entered in this field is used to further describe the machine that is accessed through this destination.

```
Enter the destination’s type [dial]: telnet
```

The destination type may be either telnet or dial. Since the marketing computer is connected to the network, this is a telnet destination.

```
Enter the telnet hostname: marketing
```

The telnet hostname may be either a valid domain name or an IP address. To use a domain name, the NetBlazer must be configured to use the domain naming system.

```
Destination 2: shortname: marketing  telnet string: marketing  long name: Market
ing Computer
Do you really want to enter this destination (y/n) [y]?
```

This prompt allows you to verify that the destination information is correct. Press the return key or enter “y” to accept the entry.

```
Destination number 3.
Enter the destination's short name (press return to end):
```

When you have entered information for all destinations in this destination group, press the return key to quit this dialog and return to the NetBlazer command prompt.

Enter a **list** command at the Top>Configure>Destination> prompt to display a list of the current destination groups.

```
Jade:Top>Configure>Destination> list
Group/#  Short Name      T/D          Host          Long Name
1    marketing
       1  databank         D  databank          Data Bank Service
       2  marketing        T  marketing         Marketing Computer
```

The above listing shows that the marketing destination group is comprised of two destinations, a telnet destination to IP address 143.191.6.2 and a dial-out destination through the databank dial-out destination.

This destination group can now be assigned to Charlie and any other user in the marketing group through the **user add** or **user destgroup** commands.

NetBlazer PASSWD File

The NetBlazer uses a PASSWD file to keep track of authorized user logins and permissions. The record format of the NetBlazer's PASSWD file, shown in Figure 9-3, is similar to the password file format used by UNIX. This allows you to FTP one or more UNIX password files to the NetBlazer, and use the information contain in these password files to create or update the NetBlazer's list of authorized users.

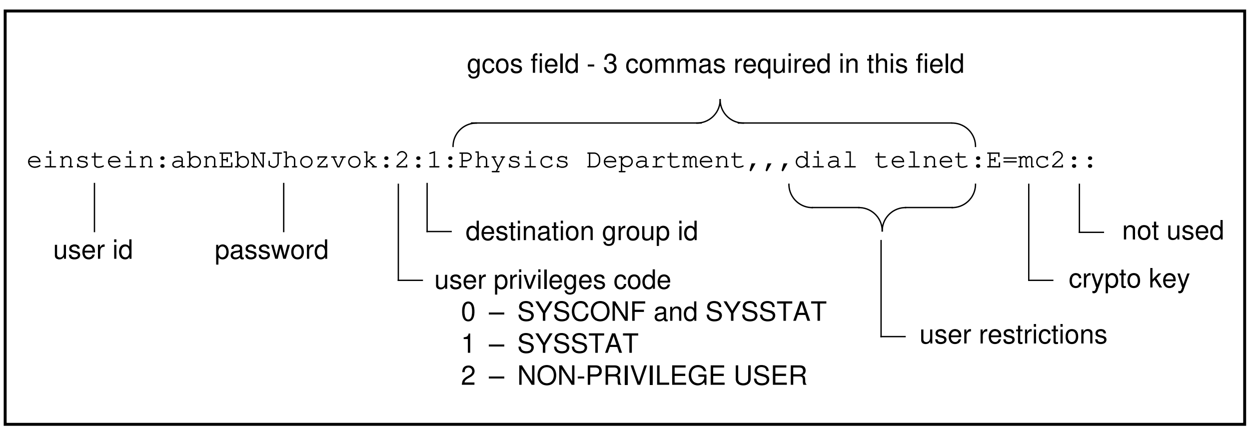


Figure 9-3. NetBlazer PASSWD File Record Format

Like the password file format used by UNIX, each record in the NetBlazer PASSWD file consists of seven fields separated by colons. However, the definitions of the fields used in the NetBlazer and UNIX password files differ as shown in Table 9-1.

Table 9-1. Password File Field Definitions

Field	UNIX Password File	NetBlazer Password File
1	userid – user's login name	userid – user's login name
2	password – user's encrypted password	password – user's encrypted password
3	uid – user's ID number	user's numeric privileges code (0,1, or 2)
4	gid – user's group ID number	user's destination group ID number
5	gcos field – information about the user	gcos field and user restrictions – see note
6	home dir – user's home directory	crypto key – used for packet mode connections
7	login shell – user's initial shell program	not used – information in this field is discarded
Note: If the gcos field in a UNIX password file contains three or more commas, the NetBlazer treats the record as a NetBlazer record. To prevent misinterpreting a home directory entry as a crypto key, the NetBlazer discards crypto keys that begin with a slash (/) in the first character position.		

The **userid** field is the user's login name. This field must contain from one to sixteen characters with no spaces.

The **password** field is the user's encrypted password. If this field is empty, a password is not required for that user to log in.

The **user privileges code** field determines whether or not the user has system configuration or status privileges. Only system administrators should be granted system configuration or status privileges (privilege codes 0 or 1). If this field contains a value of 2 or greater, the user is assigned non-privilege user status (privilege code 2).

The **destination group id** field is the gid number of the destination group assigned to the user. If this field's value does not point to a destination group with a matching gid number, the field is left empty, indicating that the user does not have an assigned destination group.

The **gcos** field in a NetBlazer PASSWD record must always contain three commas. This divides the field into four parts. The first three parts contain information about the user taken from the UNIX password file. The last part, **user restrictions**, indicates if the user's dial-out and/or telnet privileges are restricted to the user's assigned destination group.

The **crypto key** normally is used during a crypto handshaking sequence to authenticate a packet mode connection. This field should be blank for login users unless they have some way to generate the correct responses to a crypto challenge. For a packet mode connection between two NetBlazers, the crypto key assigned to the dynamic interface on the calling NetBlazer must match the crypto key assigned to the corresponding user record on the answering NetBlazer.

Editing a Copy of the UNIX Password File

If you choose to edit a copy of the UNIX password file before transferring it to the NetBlazer, you should observe the following guidelines:

- Make sure that you are using a copy of the master password file, which includes the encrypted passwords.
- Records with an asterisk (*) in the first character position of the **password** field are skipped by the NetBlazer when loading the password file.
- Records that do not contain at least three commas in the **gcos** field are treated as UNIX password records. In which case, only the **userid**, **password**, and **gcos** fields are used. The remaining fields are set to the default values listed in Table 9-2.
- Make sure that the **gcos** field contains at least three commas in any record that you want to be treated as a NetBlazer password record.

Table 9-2. NetBlazer PASSWD File Default Values

Field	NetBlazer Field Name	Default Value
1	userid	UNIX user’s login name
2	password	UNIX user’s encrypted password
3	user’s privileges code	2 – non-privileged user
4	user’s destination group ID number	null – no destination group assigned
5	gcos field and user restrictions	UNIX user’s gcos information, no restrictions
6	crypto key	null – no crypto key
7	not used	null

The remaining guidelines in this section only apply to NetBlazer records, which include at least three commas in the **gcos** field. All other records are treated as UNIX password as described earlier.

- Any **user restrictions** (i.e. dial and/or telnet) must follow the last comma in the **gcos** field. If you do not want to restrict a user’s dial-out or telnet privileges to the user’s assigned destination group, the third comma should be the last character in the **gcos** field.

- Users with a **uid** equal to 2 or greater are assigned non-privileged user status by the NetBlazer. Therefore, you do not need to change the **uid** field for most UNIX users, unless you want to grant a user NetBlazer system configuration and/or status privileges.

- If a user's **destination group ID** field does not point to a destination group with a matching gid number when the password file is loaded, it is replaced by a null value indicating that the user does not have an assigned destination group.

- To make sure that a group of users are assigned to the proper destination group, you can either edit the users' **destination group ID** fields to point to an existing destination group, or create a destination group on the NetBlazer with a gid number that matches the users' **destination group ID** field. See User Destination Groups on page 9-3.

- To prevent misinterpreting the UNIX **home dir** field as a **crypto key**, the NetBlazer discards entries in the **crypto key** field that begin with a slash (/). Therefore, you should not need to change the **crypto key** fields because normal users typically do not use crypto handshaking when logging in to the NetBlazer.

- Since the NetBlazer does not use the last field of a password file, entries in the UNIX **login shell** fields are discard, and do not need to be changed.

Loading a Password File into the NetBlazer

Before loading a password file into the NetBlazer, make sure that you understand how the NetBlazer will treat each record in the password file by reading Editing a Copy of the UNIX Password File on page 9-8.

If you want a group of users to be automatically assigned to a particular destination group, make sure that a destination group with a matching gid number exists. For instructions on how to create a destination group, refer to page 9-3.

Caution: *When a password file is loaded into the NetBlazer, the user records in the password file overwrite any current user records that have userids that are identical to the password file's userids.*

Perform the following steps to load user information from a password file into the NetBlazer.

1. Using FTP, transfer a copy of the password file from your UNIX system to the NetBlazer.

```
NB:Top> ftp napa
Type ^[ (decimal 27) <CR> to return to NetBlazer
Trying 143.191.10.100:21...
FTP session 1 connected to napa
220 napa FTP server (SunOS 4.1) ready.
Enter user name: root
331 Password required for root.
Password:
230 User root logged in.
ftp> get passwd passwd.01
200 PORT command successful.
150 ASCII data connection for passwd (143.191.10.28,1027) (17095 bytes).
Get complete: 5062 bytes in 7 sec (600/sec)
226 ASCII Transfer complete.
ftp> quit
221 Goodbye.
FTP session 1 closed: EOF
```

2. Go to the Configure User command level, and enter a **load** <filename> command as shown below.

```
NB:Top> user
NB:Top>Configure>User> load passwd.01
```

This loads and merges the user information from the password file (passwd.01 in the example) with the current user records in memory.

If you have more than one password file to load, repeat steps 1 and 2 for each additional password file.

3. List the root user information to check root's system configuration privileges.

```
NB:Top>Configure>User> list |grep root
root        ab6TRGT20sY26     (null)          login
```

The root login should have "SC" in its privileges field. If it doesn't as shown in the above example, assign **sysconf** and **sysstat** privileges to root.

```
NB:Top>Configure>User> privilege root sysconf sysstat
NB:Top>Configure>User> list |grep root
root        ab6TRGT20sY26     (null)          login     SC
```

4. To get a list of all user's with **sysconf** and **sysstat** privileges, issue the following command.

```
NB:Top>Configure>User> list |grep " SC"
root        ab6TRGT20sY26     (null)          login     SC
```

This command produces a list of users with a “ SC” in their record. The above example shows that only the root user has **sysconf** and **sysstat** privileges.

5. Verify that you can still log into the NetBlazer as root by starting a telnet session to the NetBlazer’s loopback address (127.0.0.1).

```
NB:Top>Configure>User> telnet 127.0.0.1
Type ^[ (decimal 27) <CR> to return to NetBlazer
Trying 127.0.0.1:23...
Telnet session 0 connected to 127.0.0.1

Telebit's NetBlazer 1.5

NB login: root
Password:
NB:Top> logout
Telnet session 1 closed: EOF
```

If you cannot log in as root, escape out of the telnet session, and change the root password by issuing a **password root** <password> command.

6. Save the user information to disk by issuing a **save** command at the Configure User command level.

```
NB:Top>Configure>User> save
```

The **user save** command saves the current user information from memory into the PASSWD file on the diskette.

NetBlazer User IDs

You should add a user ID for each person that uses the NetBlazer to dial out to a remote computer, or logging into the NetBlazer from a remote terminal.

To manually add a user to the system, invoke the **add** command at the Configure User command level:

```
Top>Configure>Destination> user
Top>Configure>User> add

Enter the uname (1-16 chars): charlie
Enter a user ID consisting of from 1 to 16 characters with no spaces.
```

The uname is the name that the user enters to log in to the NetBlazer.

```
Want to enter a password [y]?
```

Press the return key if you want to assign a password to this user. You will be prompted for the password. If you do not want to assign a password to this user enter “n” at the prompt.

```
Enter the password (no echo):
Re-enter the password:
```

Enter the user’s password. The password is not displayed on the screen as you type it, so you are asked to enter the password twice to ensure that you entered the password correctly.

```
Issue a crypto handshake [n]?
```

Unless this user has some way of responding to a crypto challenge, press the return key or enter “n” to prevent the NetBlazer from issuing a crypto challenge to the user during the login process.

```
Allow system configuration privileges (n/y) [n]?
Allow system status privileges (n/y) [n]?
```

Only the system administrator and designated staff should be granted system configuration or status privileges. If this is one of those users enter “y” to either of the above prompts. If you grant system configuration privileges to this user, the NetBlazer will automatically grant status privileges to the user.

```
Restrict dialout privileges (n/y) [n]?
Restrict telnet privileges (n/y) [n]?
```

These two prompts are used to restrict a user’s dial-out or telnet privileges to the user’s assigned destination group. Answering “y” to either of these two prompts restricts the user’s access to those dial-out or telnet destinations contained in the user’s destination group.

If no destination group is assigned to the user, then the restrictions will not apply. However, if the user is assigned to a destination group that does not include at least one dial-out or telnet destination, restricting the user’s dial-out or telnet privileges will block the users ability to establish a dial-out or telnet connection.

```
Assign to a destination group (y/n) [n]? y
Enter the destination group name: marketing
```

If you have a destination group for this user answer “y” to the first prompt, and enter the name of the destination group. Otherwise, press the return key or enter “n” at the first prompt. You can assign a destination group to the user later by issuing a **user destgroup** <userid> <destination> command where <userid> is the user’s login ID and <destination> is the name of the destination group to be assigned to the user.

```
User: charlie           privs:     dest-group: marketing
Okay (yes|no|quit) [y]?
```

Press the return key or enter “y” to accept the above user ID record. Otherwise, enter “n” or “q” to discard this record.

Enter **list** at the command prompt to display a listing of user ID records:

```
Top>Configure>User> list
USERID      PASSWORD          CRYPTO          USE       PRIVS   DEST-GROUP
charlie     abhp2JZi0DfYI     (null)          login             marketing
branch      ab9b/E5gZTar6     incrypt         dial SLIP
root        abJnggxhB/yWI     (null)          login     SC
```

When you have finished entering users, make sure that you have a NetBlazer system diskette in the drive, and save the user ID records to diskette by entering **save** at the Configure User command level.

```
Top>Configure>User> save
```

Anytime you make changes to the user ID records, you should issue a **user save** command to the NetBlazer.

Automatically Connecting a User to a Specific Destination

If you have users that only use the NetBlazer as an intermediary to connect to a single destination, you can create special aliases for those user that automatically starts a session with a predetermined destination when the user logs into the NetBlazer. For example, suppose that you have a user named Carol who uses the NetBlazer to call in from home and telnet to a host system named napa. You can a user account and aliases for Carol that automatically connects her to napa whenever she logs into the NetBlazer as illustrated below.

```
Jade:Top> user add carol
Jade:Top> alias add login_carol telnet napa
```

The **alias add** command in the above example creates a special login alias for the user ID, carol, that automatically starts a telnet session with the host named napa.

If you have several users, like Carol, who need to automatically connect to napa from the NetBlazer consider create a user account for napa that can be used by any user logging into the NetBlazer as napa.

```
Jade:Top> user add napa
Jade:Top> alias add login_napa telnet napa
```

Another way to automatically connect a dial-in user to a host is to assign a telnet command to one of the asynchronous lines as described under Setting Up a Line to Automatically Connect to a Host in Chapter 7, Asynchronous Lines Usage. When a call is received on that line, the NetBlazer automatically establishes a telnet between the user and the host. The disadvantage of use this method is that the line only can be used for dial-in connections with the host.

IP Packet Filtering 10

The **ip filter** commands maintain a list of IP packet filters that are used to restrict IP packet routing to and from designated networks or host addresses through a particular interface.

Input and/or output filtering for a specified interface can be performed on a source address, a destination address, or a source-destination address pair. You can further restrict the filter to only apply to a specified protocol and port number or range.

Initially the packet filter list is empty which allows all IP packets to pass through the NetBlazer unimpeded. However, adding a filter to the list automatically turns on packet filtering for the specified interface, and all packets routed through that interface are checked for permission to continue.

When packet filtering is turned on for a particular interface, a default filter for that interface is automatically created to be the complement of the first filter. If the first filter is a permit filter, the default filter is set to be a deny filter. On the other hand, if the first filter is a deny filter, the default filter is set to be a permit filter.

Note: *If an interface is assigned filters in only one direction (i.e. all inbound or all outbound filters), the default filter for that interface only applies to packets traveling in the same direction. Packets going in the opposite direction are automatically permitted.*

To determine whether or not an IP packet is allowed to pass through the NetBlazer, it first checks the input interface filter list for an inbound filter that applies to the packet. If the packet is permitted to continue based on a matching inbound filter, the NetBlazer then checks the routing table to verify that a route exists for the packet. If a route exists, the NetBlazer checks the output interface filter list for an outbound filter that applies to the packet. If the packet is also permitted to continue based on a matching outbound filter, it is allowed to pass through the NetBlazer toward its destination.

If the NetBlazer determines that it does not have permission or the ability to route the packet, it discards the packet and sends an ICMP Destination Unreachable message back to the source address of the packet.

To illustrate how IP packet filtering may be used, consider the network shown in Figure 10-1.

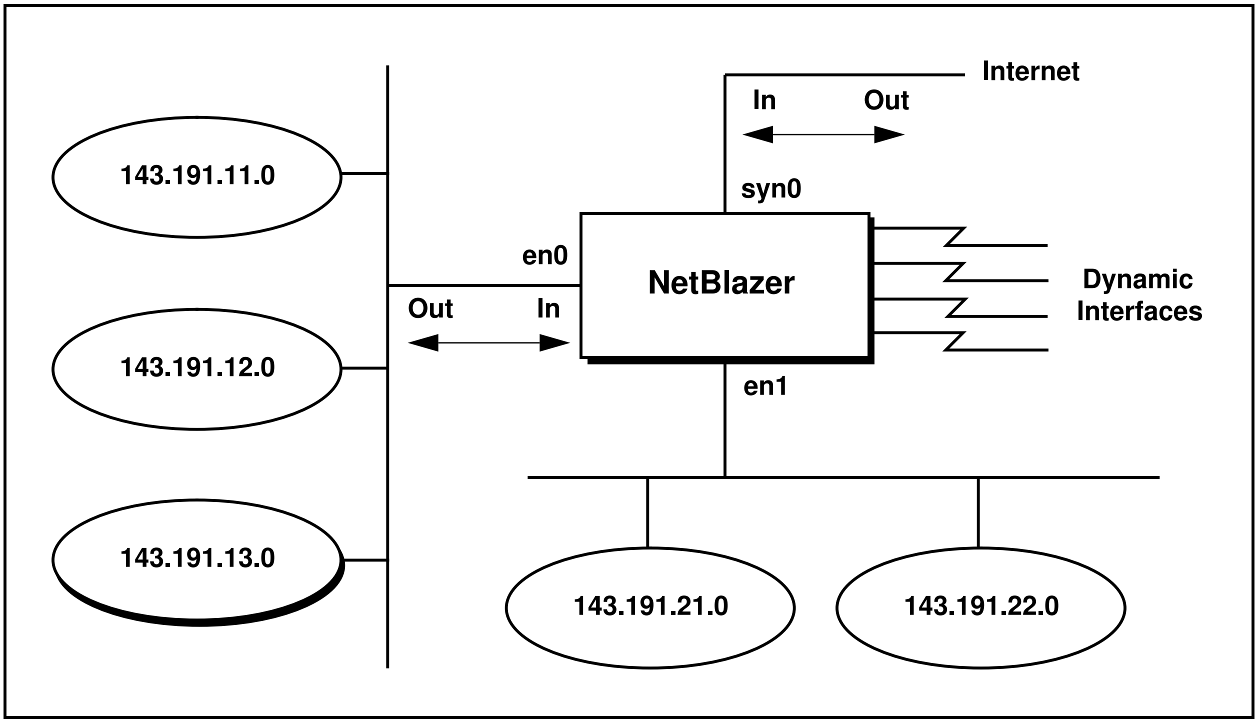


Figure 10-1. Filtering Example Network

Filtering Network IP Addresses

Assuming that you want to allow hosts on the Internet to use the NetBlazer only to reach hosts on the 143.191.0.0 network, go to the Filter command level and add a permit filter for network address 143.191.0.0 via the syn0 interface as shown below.

```
NB:Top> filter
NB:Top>Configure>IP>Filter> permit 143.191/16 syn0
NB:Top>Configure>IP>Filter> list
Source              Destination         Interface Protocol I/O  Permit/Deny
143.191.0.0/16                          syn0               In   Permit
                    143.191.0.0/16      syn0               In   Permit
143.191.0.0/16                          syn0               Out  Permit
                    143.191.0.0/16      syn0               Out  Permit
Default                                 syn0                    Deny
```

At this point, only IP packets with a source or destination address of 143.191.X.X are allowed to pass through the syn0 interface.

Notice that a default filter has been automatically created that denies packets not sent to or from the 143.191.0.0 network from passing through the syn0 interface.This prevents hosts on the Internet from using the NetBlazer to reach a host that is not part of the 143.191.0.0 network.

A more precise way to accomplish the goal from the previous example is illustrated below.

```
NB:Top>Configure>IP>Filter> flush syn0
NB:Top>Configure>IP>Filter> permit 143.191/16 syn0 source out
NB:Top>Configure>IP>Filter> permit 143.191/16 syn0 dest in
NB:Top>Configure>IP>Filter> list
Source             Destination        Interface Protocol I/O  Permit/Deny
                   143.191.0.0/16     syn0               In   Permit
143.191.0.0/16                        syn0               Out  Permit
Default                               syn0                    Deny
```

The first command in this example removes all filters assigned to the syn0 interface from the list of filters. The next two commands permit outbound packets with a source address of 143.191.X.X or inbound packets with a destination address of 143.191.X.X to pass through the syn0 interface.

Using Filter Lookup to Check a Filter

You can use the **filter lookup** command to confirm that the filters and routing table are set up correctly. For example, to verify that packets can be routed to or from the 143.191.13.0 subnetwork, enter the following commands.

```
NB:Top>Configure>IP>Filter> lookup 128.10.2.1 143.191.13.1 syn0
permission to route the packet granted by syn0 input filters
route to 143.191.13.1 using interface en0
NB:Top>Configure>IP>Filter> lookup 143.191.13.1 128.10.2.1 en0
route to 128.10.2.1 using interface syn0
permission to route the packet granted by syn0 output filters
```

Notice that the <interface> parameter used in the **filter lookup** command must specify the name of the input interface. Otherwise you will get misleading results as shown below.

```
NB:Top>Configure>IP>Filter> lookup 143.191.13.1 128.10.2.1 syn0
permission to route the packet denied by syn0 input filters
```

This example also confirms that an inbound packet destined for a network other than 143.191.0.0 will not be permitted through the syn0 interface.

Filtering Destination Port Numbers

To illustrate how destination port number ranges are used, assume that you want to block outside access to the 143.191.13.0 subnetwork for any purpose other than transferring mail, but you want to allow hosts on this subnetwork to initiate Telnet or FTP sessions.

To accomplish this goal, you should first create deny filters that prevent any packet sent to or from subnetwork 143.191.13.0 from passing through the en0 interface. Then create permit filters which specifically allow packets with the following well-known TCP destination port numbers to pass through the en0 interface from the 143.191.13.0 subnetwork:

Number	Keyword	Description
20	FTP-DATA	File Transfer (Default Data)
21	FTP	File Transfer (Control)
23	TELNET	Terminal Connection
25	SMTP	Simple Mail Transfer Protocol

Port numbers greater than 1023 are generally assigned to server clients at the source when a session is initiated. This assigned port number is used by the server at the other end to send information back to the appropriate client. Therefore, it is important to include a port number range of >1023 in the permit filters to ensure that server responses can be sent back to the client.

The two **filter deny** commands in the next example prevent a packet sent to or from the 143.191.13.0 subnetwork from passing through the en0 interface unless the packet meets the TCP port matching criteria defined by the two **filter permit** commands.

```
NB:Top>Configure>IP>Filter> deny 143.191.13/24 en0 dest out
NB:Top>Configure>IP>Filter> deny 143.191.13/24 en0 source in
NB:Top>Configure>IP>Filter> permit 143.191.13/24 en0 tcp 25 >1023 dest out
NB:Top>Configure>IP>Filter> permit 143.191.13/24 en0 tcp 20-25 !=22 !=24
>1023 source in
NB:Top>Configure>IP>Filter> list en0
Source             Destination        Interface Protocol I/O  Permit/Deny
143.191.13.0/24                       en0       TCP      In   Permit
                                     port rules:  20<= <=25  !=22 !=24 >1023
143.191.13.0/24                       en0                In   Deny
                   143.191.13.0/24    en0       TCP      Out  Permit
                                        port rules: =25 >1023
                   143.191.13.0/24    en0                Out  Deny
Default                               en0                     Permit
```

Packets destined for subnetwork 143.191.13.0 with a port number equal to 25 or greater than 1023 are allowed to pass through the en0 interface by the first **filter permit** command. Packets sent from the 143.191.13.0 subnetwork with a port number that is equal to 20, 21, 23 or 25 or is greater than 1023 are also permitted to continue through the en0 interface based on the second **filter permit** command.

Caution: *Many servers have assigned port numbers greater than 1023. If you set up permit filters in both directions for port numbers greater than 1023 as in the above example, and do not want outside hosts to reach a server with an assigned port number greater than 1023, you must exclude the port number in the destination permit filter.*

The following example shows you how to exclude destination port 6000 in the 143.191.13/24 en0 destination out permit filter.

```
NB:Top>Configure>IP>Filter> permit 143.191.13/24 en0 tcp !=6000 dest out
NB:Top>Configure>IP>Filter> list en0
Source             Destination        Interface Protocol I/O  Permit/Deny
143.191.13.0/24                       en0       TCP      In   Permit
                                     port rules:  20<= <=25  !=22 !=24 >1023
143.191.13.0/24                       en0                In   Deny
                   143.191.13.0/24    en0       TCP      Out  Permit
                                        port rules: =25 >1023 !=6000
                   143.191.13.0/24    en0                Out  Deny
Default                               en0                     Permit
```

The above filter list for the en0 interface does not allow UDP or ICMP packets to be sent to or from the 143.191.13.0 subnetwork. Permit filters for each of these protocols must be defined separately. For example, to allow ICMP packets to be sent to or from the 143.191.13.0 subnetwork, enter the following **filter permit** commands.

```
NB:Top>Configure>IP>Filter> permit 143.191.13/24 en0 icmp dest out
NB:Top>Configure>IP>Filter> permit 143.191.13/24 en0 icmp source in
NB:Top>Configure>IP>Filter> list en0
Source             Destination        Interface Protocol I/O  Permit/Deny
143.191.13.0/24                       en0       ICMP     In   Permit
143.191.13.0/24                       en0       TCP      In   Permit
                                     port rules:  20<= <=25  !=22 !=24 >1023
143.191.13.0/24                       en0                In   Deny
                   143.191.13.0/24    en0       ICMP     Out  Permit
                   143.191.13.0/24    en0       TCP      Out  Permit
                                        port rules: =25 >1023
                   143.191.13.0/24    en0                Out  Deny
Default                               en0                     Permit
```

Note: *ICMP Destination Unreachable messages are sent even if ICMP outbound packets are denied.*

The next series of examples show you how to verify that the filter list is set up correctly.

First, issue a **filter lookup** command to verify that the NetBlazer will still route packets through the en0 interface from a subnetwork other than 143.191.13.0.

```
NB:Top>Configure>IP>Filter> lookup 143.191.12.1 128.10.2.1 en0
permission to route the packet granted by en0 input filters
route to 128.10.2.1 using interface syn0
permission to route the packet granted by syn0 output filters
```

Verify that hosts on the 143.191.13.0 subnetcan send packets to Telnet's well-known port number 23 through the en0 interface, and that Telnet (and other servers) have a response path to the 143.191.13.0 subnetwork.

```
NB:Top>Configure>IP>Filter> lookup 143.191.13.1 128.10.2.1 en0 tcp 23
permission to route the packet granted by en0 input filters
route to 128.10.2.1 using interface syn0
permission to route the packet granted by syn0 output filters
NB:Top>Configure>IP>Filter> lookup 128.10.2.1 143.191.13.1 syn0 tcp 1024
permission to route the packet granted by syn0 input filters
route to 143.191.13.1 using interface en0
permission to route the packet granted by en0 output filters
```

Verify that outside hosts cannot telnet into the 143.191.13.0 subnetwork.

```
NB:Top>Configure>IP>Filter> lookup 128.10.2.1 143.191.13.1 syn0 tcp 23
permission to route the packet granted by syn0 input filters
route to 143.191.13.1 using interface en0
permission to route the packet denied by en0 output filters
```

Finally, verify that ICMP messages can be sent to or from the 143.191.13.0 subnetwork.

```
NB:Top>Configure>IP>Filter> lookup 143.191.13.1 128.10.2.1 en0 icmp
permission to route the packet granted by en0 input filters
route to 128.10.2.1 using interface syn0
permission to route the packet granted by syn0 output filters
NB:Top>Configure>IP>Filter> lookup 128.10.2.1 143.191.13.1 syn0 icmp
permission to route the packet granted by syn0 input filters
route to 143.191.13.1 using interface en0
permission to route the packet granted by en0 output filters
```

Filtering Source and Destination Address Pairs

To illustrate how source and destination address pairs are used, assume that you want to prevent hosts on network 128.128.0.0 from communicating with hosts on the 143.191.0.0 network for any purpose other than exchanging mail. This can be accomplished by entering the following commands.

```
NB:Top>Configure>IP>Filter> deny 128.128/16 143.191/16 syn0 tcp <1024 !=25 in
NB:Top>Configure>IP>Filter> deny 143.191/16 128.128/16 syn0 tcp <1024 !=25 out
NB:Top>Configure>IP>Filter> list syn0
Source            Destination        Interface Protocol I/O  Permit/Deny
128.128.0.0/16    143.191.0.0/16     syn0      TCP      In   Deny
                                       port rules: !=25 <1024
                  143.191.0.0/16     syn0               In   Permit
143.191.0.0/16    128.128.0.0/16     syn0      TCP      Out  Deny
                                       port rules: !=25 <1024
143.191.0.0/16                       syn0               Out  Permit
Default                              syn0                    Deny
```

The two deny commands in the above example prevents any packet with a destination port less than 1024 and not equal to the SMTP well-known port 25 from traveling between the 143.191.0.0 and 128.128.0.0 networks through the syn0 interface.

Packets with a TCP destination port equal to 25 or greater than 1023 are permitted through the syn0 interface by the filters assigned to the syn0 interface in a previous example.

Verify that only packets with the SMTP well-known destination port 25 are allowed to pass between the 143.191.0.0 and 128.128.0.0 networks through the syn0 interface by issuing the following **filter lookup** commands.

```
NB:Top>Configure>IP>Filter> lookup 128.128.1.1 143.191.12.1 syn0 tcp 25
permission to route the packet granted by syn0 input filters
route to 143.191.12.1 using interface en0
permission to route the packet granted by en0 output filters
NB:Top>Configure>IP>Filter> lookup 128.128.1.1 143.191.12.1 syn0 tcp 23
permission to route the packet denied by syn0 input filters
NB:Top>Configure>IP>Filter> lookup 143.191.12.1 128.128.1.1 en0 tcp 25
permission to route the packet granted by en0 input filters
route to 128.128.1.1 using interface syn0
permission to route the packet granted by syn0 output filters
NB:Top>Configure>IP>Filter> lookup 143.191.12.1 128.128.1.1 en0 tcp 23
permission to route the packet granted by en0 input filters
route to 128.128.1.1 using interface syn0
permission to route the packet denied by syn0 output filters
```

Deleting a Filter

To illustrate how the **filter delete** command is used, assume that you no longer wish to restrict communications between the 143.191.0.0 and 128.128.0.0 networks. The following example shows you how to remove the filters associated with this restriction.

```
NB:Top>Configure>IP>Filter> list syn0
Source             Destination        Interface Protocol I/O  Permit/Deny
128.128.0.0/16     143.191.0.0/16     syn0      TCP      In   Permit
                                        port rules: =25
128.128.0.0/16     143.191.0.0/16     syn0               In   Deny
                   143.191.0.0/16     syn0               In   Permit
143.191.0.0/16     128.128.0.0/16     syn0      TCP      Out  Permit
                                        port rules: =25
143.191.0.0/16     128.128.0.0/16     syn0               Out  Deny
143.191.0.0/16                        syn0               Out  Permit
Default                               syn0                    Deny
NB:Top>Configure>IP>Filter> delete 128.128/16 143.191/16 syn0
NB:Top>Configure>IP>Filter> delete 128.128/16 143.191/16 syn0 tcp
NB:Top>Configure>IP>Filter> delete 143.191/16 128.128/16 syn0
NB:Top>Configure>IP>Filter> delete 143.191/16 128.128/16 syn0 tcp
NB:Top>Configure>IP>Filter> list syn0
Source             Destination        Interface Protocol I/O  Permit/Deny
                   143.191.0.0/16     syn0               In   Permit
143.191.0.0/16                        syn0               Out  Permit
Default                               syn0                    Deny
```

Network Management 11

Configuring the NetBlazer to Support SNMP

The NetBlazer supports the Simple Network Management Protocol (SNMP) to allow a remote management station to access information contained within the NetBlazer and to perform limited remote control over the NetBlazer.

The NetBlazer currently supports two Management Information Base (MIB) standards, MIB-1 and MIB-2, which define how to monitor and control the operation of a typical router device.

```
NB:Top>Configure>snmp
NB:Top>Configure>SNMP> sysContact John Smith, 555-1234
NB:Top>Configure>SNMP> sysDescr Gateway to Telebit Network 143.191.19
```

The **sysContact** command allows you to enter information which is retrieved by a remote management station to identify the person or organization responsible for maintaining the NetBlazer.

The **sysDescr** command is used to provide a description of the NetBlazer to a remote management station.

Next, enter the community names for the SNMP get and set functions. These community names act like passwords in that the NetBlazer refuses requests or commands from a management station that does not provide the proper community names.

```
NB:Top>Configure>SNMP> get-community public
NB:Top>Configure>SNMP> set-community private
```

The **get-community** command sets the name that must be provided by the remote management station when it attempts to retrieve status or configuration information.

The **set-community** command sets the name that must be provided by the remote management station when it attempts to issue configuration commands to the NetBlazer.

Normally the NetBlazer sends responses to SNMP queries from a remote management station. There is an exception to this, however, called a trap. A trap is an unsolicited message transmitted by the agent in response to a significant event. A trap may be considered an alarm message.

Before the NetBlazer can send any traps the administrator must set the trap community name with the **trap-community** command.

```
NB:Top>Configure>SNMP> trap-community SNMP_trap
```

The **trap-community** command sets the name that is transmitted to the remote management station when the NetBlazer sends a trap.

You must also set up one or more destinations for trap messages, and enable the transmission of specific trap messages. Go to the Configure SNMP Trap command level and enter **list** to display the current status of the SNMP traps.

```
NB:Top>Configure>SNMP> trap
NB:Top>Configure>IP>SNMP>Trap> list
trap addresses:
No managers set
Startup traps are off
Authentication traps are off
```

Two types of traps can be sent by the NetBlazer – an authentication trap and a startup trap. An authentication trap is sent by the NetBlazer when a query arrives from a management station and the community name in the request is incorrect. The startup trap is sent whenever the NetBlazer's SNMP server is started. This lets the management station(s) know that the NetBlazer is operational.

The next example shows you how to add the IP addresses for a couple of management stations to the trap address list, and enable the transmission of startup and authentication traps.

```
NB:Top>Configure>SNMP>Trap> add robin.telebit.com
NB:Top>Configure>SNMP>Trap> add 143.191.11.5
NB:Top>Configure>SNMP>Trap> startup on
NB:Top>Configure>SNMP>Trap> authentication on
NB:Top>Configure>SNMP>Trap> list
trap addresses:
143.191.10.6    143.191.11.5
Startup traps are on
Authentication traps are on
```

Notice that you can add an IP address to the trap address list by enter a domain name or an IP address. The NetBlazer supports up to five trap addresses. The startup and authentication traps can be set to on or off.

You can remove an addresses from the list of trap addresses by entering a **delete** command at the Configure SNMP Trap command level.

```
NB:Top>Configure>IP>SNMP>Trap> delete robin.telebit.com
NB:Top>Configure>IP>SNMP>Trap> list
trap addresses:
143.191.11.5
Startup traps are on
Authentication traps are on
```

Like the **snmp trap add** command, the **snmp trap delete** command accept either an IP address or a domain name.

Before the NetBlazer will actually start using the SNMP, you must turn on the SNMP server.

```
NB:Top>Configure>SNMP>Trap> server
NB:Top>Configure>Servers> start snmp
NB:Top>Configure>Servers> list
Server     Status
discard    off
echo       off
finger     off
ftp        off
rip        on
snmp       on
telnet     on
udpdiscard off
udpecho    off
```

This allows the NetBlazer to respond to SNMP requests or commands. Refer to your management station SNMP documentation for additional information concerning how to manage a router using SNMP.

Using Syslog to Monitor NetBlazer Activity

You can configure the NetBlazer to send syslog messages to a UNIX host, the NetBlazer console, your login terminal, and/or an internal buffer in the NetBlazer's memory. Messages stored in memory can be listed to a disk file or to your terminal screen.

This section explains how to configure the NetBlazer to send syslog messages to a UNIX host. For information on sending syslog messages to the other destinations, refer to the **syslog** command description in the *NetBlazer Command Reference Manual*.

Sending Syslog Messages to a UNIX Host

The **syslog host** command controls sending NetBlazer activity information to a UNIX system's *syslogd* daemon, which then forwards this information to the appropriate system log files and/or users.

```
NB:Top>Configure>Syslog> host on byron local0
sending syslog messages to byron(143.191.10.243), facility = local0
```

The above example shows you how to configure the NetBlazer to send syslog messages to a UNIX host named byron using the local0 facility.

As illustrated in Figure 11-1, the */etc/syslog.conf* file on the UNIX system controls where the NetBlazer messages are sent.

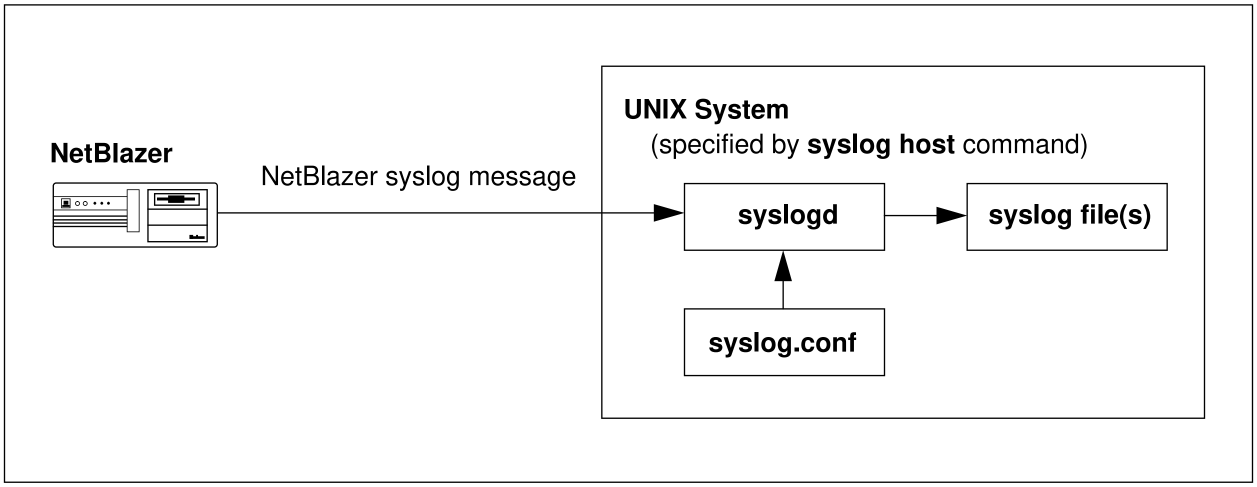


Figure 11-1. Sending Syslog Messages to a UNIX Host

For information on setting up your UNIX host system, see the following UNIX manual pages: *syslog*, *syslogd*, *syslog.conf*.

Verifying that Syslog Messages Are Reaching the Host

You can verify that syslog messages from the NetBlazer are reaching the right file on the host system by issuing a **msglog** <message> command. The <message> parameter may be a single word or several words enclosed in quotes as shown in the following example.

```
NB:Top>Configure>Syslog> msglog "message from NetBlazer"
```

If the message sent by the **msglog** command does not reach the host log file, try the following:

1. Enter a **syslog list** command to verify that the the NetBlazer's syslog messages are being sent to the correct UNIX host.

2. Enter a **trace** <interface> **101** command where <interface> specifies the name of the interface that syslog messages will be sent through. Then, send your syslog message again to verify that the message is going out.

   If you do not see trace information about the message, verify that you have the interface and route set correctly to reach the host.

3. Verify that your */etc/syslog.conf* configuration file on the host is set up correctly to route messages from the NetBlazer to the appropriate log file. You should have a line in your */etc/syslog.conf* file similar to the following:

   ```
   local0.debug          /var/log/netblazer
   ```

4. Make sure that the log file, /var/log/netblazer in the above example, is present on your host system.

5. Make sure that your host system is only running one *syslogd* process.

6. Try running *syslogd* with the *-d* (debug) option.

NetBlazer File Definitions A

The NetBlazer uses the following files which are contained on the system diskette:

- **TELEBIT.SYS** – This is the operating system file that is used to load and execute the NetBlazer's executable code.
- **FRED** – This file contains the NetBlazer's executable code that runs under TELEBIT.SYS.
- **TOKEN001.MCD** – This file contains the executable microcode for the Token Ring board that is down loaded to the Token Ring Board on startup.
- **STARTUP.CNF** – This file contains configuration commands, which are loaded when the NetBlazer is powered up or rebooted. Issuing a **configure save** command updates the STARTUP.CNF file for the current configuration.
- **DEFAULT.CNF** – This file contains the default configuration commands set by Telebit. You can restore the NetBlazer to its default configuration by copying this file to the STARTUP.CNF file and rebooting the system.
- **PASSWD** – The NetBlazer system software uses this file (when the NetBlazer is powered up or rebooted) to keep track of authorized user logins and permissions. This facilitates using UNIX password files to create or update the NetBlazer's list of authorized users.

  A detailed description of the NetBlazer's PASSWD file format and use is provided in Chapter 9, User IDs and Destination Groups.
- **MODEMCAP.TXT** – This file contains the modem capabilities definitions for modems which are supported by the NetBlazer. This information is used in conjunction with the dialout and line characteristics to automatically configure the modems attached to the NetBlazer.
- **CHAT.TXT** – This file controls the log-in and authentication exchange, which takes place between the NetBlazer and a remote device (usually another NetBlazer) when establishing a packet mode connection.

Editing a NetBlazer File Using MS-DOS

Although the NetBlazer does not run under MS-DOS, its diskette and file formats are compatible with MS-DOS. Therefore, you can edit any of the configuration (CNF) or text (TXT) files on the NetBlazer diskette using a text editor program on any MS-DOS computer system that supports high density 1.44 Mbyte 3.5 inch diskettes.

You should make a duplicate copy of the NetBlazer system diskette using the DISKCOPY command on your DOS system. Use the duplicate diskette as a working copy, and put the original NetBlazer system diskette in a safe place.

STARTUP.CNF File Definition

The NetBlazer STARTUP.CNF file consists of a list of NetBlazer commands which are executed in sequence each time the NetBlazer is powered up or rebooted. Refer to the *NetBlazer Command Reference Manual*, for a description of the commands used by the NetBlazer.

When editing the STARTUP.CNF file, do not use commands such as **ipdial** which require operator input, and include all required parameters in each command entered. Also avoid using commands which may have adverse or unpredictable results such as **reboot**, **save** or any of the **disk** commands.

Pay special attention to the order that the configuration commands are entered. Since the commands are executed in sequence starting at the top and working down through the file, configuration commands which define parameters used by other commands should be entered earlier in the list. Several cases where the command order is important are listed below.

- The IP address must be set before any command is entered that would cause the NetBlazer to start transmitting packets.
- Commands defining the NetBlazer interfaces should be entered before commands used to specify routes.
- Commands used to create and define a characteristics group should be entered before any command that requires a group name as one of its parameters.

Also keep in mind that a command appearing later in the list may override an earlier command.

MODEMCAP.TXT File Definition

The MODEMCAP.TXT file contains definitions of the various Telebit modems supported by the NetBlazer. You can edit the MODEMCAP.TXT file to add support for modem features which are not already supported.

When the NetBlazer detects an off-to-on transition on the DSR line of a RS-232 port, it attempts to determine the type of modem attached to that port by sending an AT~I command. Telebit modems respond to this command by transmitting a unique numeric product identification code. If the NetBlazer receives a response to the AT~I command, it attempts to match the response to one of the response codes associated with a modem type contained in the modemcap file. If the NetBlazer does not receive a response or if it does not find a matching code in the MODEMCAP.TXT file, the NetBlazer assumes that there is no modem attached to the line and starts a standard terminal login on that line.

MODEMCAP.TXT entries start with a name in column 1 and then tag:command-string pairs indented one tab. The tag is separated from the modem command string by a colon.

Command Line Formats

The MODEMCAP.TXT file is composed of a series of command lines. The following four basic types of command lines are used:

- A **label** is any alphanumeric string starting at the left-most character position in the line. It denotes the beginning of a new modem definition. The modem definition continues until another label is encountered in the command line sequence.
- A **tag** is always preceded by white space, i.e. a number of spaces or tab characters, and consists of a tag name followed by a colon (:) and the value assigned to that tag. In many cases, the value assigned to a tag is a modem command string. Refer to your modem's reference manual for descriptions of the modem commands and registers.
- The **$include:<label>** metacommand is used to include a previous definition in the current modem definition. Like the tag command line, a $include command line must be preceded by white space.
- A **comment** line is any line that has a pound character (#) in the first character position. It is used to add descriptive information to the MODEMCAP.TXT file.

The character sequences defined in Table 1-1 may be used in a tag command line to perform special functions.

Table 1-1. Special Functions Character Sequence

Character Sequence	Description
\b	send a backspace character
\f	send a formfeed character
\n	send a newline character (linefeed)
\r	send a carriage return character
\t	send a tab character
\\	send a backslash character
\?	send a question mark character
\’	send a single quote character
\”	send a double quote character
\0	send the null character (value zero)
\ooo	send the character whose value is the three octal digits following the backslash
\xhh	send the character whose value is the two hexadecimal digits following the backslash and the letter “x”

Required Tag Commands

The command line tags defined in table are required in each modem type definition along with the modem identification string that is returned in response to an AT~I modem identification command..

Table 1-2. Required Command

Tag	Definition
hard-reset[0-3]	strings used to initialize modem
reset-pause	milliseconds to pause after resetting modem
speed-cmd	command to set modem speed while using modem
line-speed	NetBlazer line speed while using modem
autospeed-cmd	command to set modem speed while configuring modem
autospeed	NetBlazer line speed while configuring modem
list-cmd	command to display the current modem configuration
line-characteristics	characteristic used to match dialout's characteristics
dial	string that precedes phone number
no-answer	string to turn off answering (e.g. ats0=0)

Hard-Reset Tag

The **hard-reset** tags specify the modem command strings that are sent to the modem when the NetBlazer determines that the modem is not properly configured. A modem definition may include from one to four hard-reset command lines, labeled **hard-reset0** through **hard-reset3**.

Here are the **hard-reset** tags for the Telebit T2500 modem:

```
hard-reset0:at~&fs51=254e0v0s0=1s45=255s50=0s52=2s54=3s55=3s58=2
hard-reset1:ats64=1s66=1s68=2s92=1s95=2s97=1s98=3s106=1s130=0s131=1
```

When resetting a modem, the NetBlazer issues each modem command string in sequence.

Reset-pause Tag

The **reset-pause** tag specifies the amount of time in milliseconds that the NetBlazer should wait before getting the active configuration after using the **hard-reset** tags to reconfigure the modem.

```
reset-pause:200
```

Speed-cmd and Line-speed Tag

The **speed-cmd** and **line-speed** tags are used to synchronize the interface speed of the modem to the interface speed of the NetBlazer using the modem.

```
speed-cmd:ats51=5
line-speed:19200
```

Autospeed-cmd and Autospeed Tags

The **autospeed-cmd** and **autospeed** tags are used to synchronize the interface speed of the modem to the interface speed of the NetBlazer configuring the modem.

```
autospeed-cmd:ats51=254
autospeed:19200
```

List-cmd Tag

The **list-cmd** tag specifies the modem command that is used to get the current configuration of the modem. The NetBlazer uses this information to determine if the modem's active configuration has been changed by performing a hard reset. If the modem's active configuration following a hard reset differs from its configuration prior to the hard reset, the NetBlazer writes the active configuration in the modem's nonvolatile memory.

```
list-cmd:at~n?
```

Line-Characteristics Tag

The **line-characteristics** tag specifies the default characteristics group that is assigned to any line attached to the defined type of modem. The following is the standard entry that is preassigned for all Telebit modems:

```
line-characteristics:dialout
```

Note that this default characteristics group can be overruled on a line-by-line basis by issuing a **line characteristics** command for a specified line and saving the current configuration by issuing a **configure save** command.

Dial Tag

As illustrated below, the **dial** tag defines the modem command which is sent by the NetBlazer when dialing out to a remote location.

```
dial:atdt %s
```

The NetBlazer dialout code replaces the %s with the actual phone number when dialing takes place.

No-answer Tag

The **no-answer** tag defines the modem command that is used to prevent the modem from answering incoming calls.

Modem ID Response Tag

The last mandatory tag is the identification code that is returned by the modem in response to the AT~I command. Usually there are two codes, one that will be returned by the modem when it has been properly preconfigured (does not require a hard-reset string) and one that indicates that it is not preconfigured and must be reset.

For example, the T2500 modem uses the following modem ID response tags:

```
971:0
9710:1
972:0
9720:1
```

Since the NetBlazer uses the numeric response mode of the modem and the modem normally generates text response codes, this is used by the NetBlazer to indicate whether or not the modem is configured. These tags are special in that the name of the tag is the code that is returned in response to the AT~I command and the value is either 0 (the NetBlazer should send the hard-reset string to the modem) or 1 (the modem is preconfigured).

Specific Function Tags

Several tags are included in the MODEMCAP.TXT file which are used to enable or disable a particular modem function.

The following are common characteristic tags that are used in this file. These characteristic tags are also include in the standard set of characteristic groups provided by Telebit.

Table 1-3. Special Function Characteristic Tags

Characteristic Tag	Definition
pep	string to put the modem into PEP mode
v32	string to put modem into v32 mode
v42off	string to turn off v42 mode
nov42bis	string to turn off v42bis mode
mnpoff	turn off mnp compression
bell212a	string to put modem into bell212a mode
The following two tags are used together for modems that do not hang up the line when DTR is lowered.	
attention	escape attention sequence usually +++
hang-up	hang up command

For example, since the T2500 modem definition preconfigures T2500 modems to use V.42 error correction and V.42bis data compression, the following tag is provided to allow you to disable these functions for a particular connection:

```
v42off:ats95=0s97=0s106=0
```

To use the above tag to turn off V.42/V.42bis error correction and data compression, you must first create a new characteristics group that includes the v42off tag as one of its characteristics.

```
Top> group
Top>Configure>Group> add no-v42
Top>Configure>Group> modify no-v42 add v42off
Top>Configure>Group> list no-v42
no-v42:
 0: v42off
```

Then, assign this new characteristics group to the specific dialout destination(s) which will not be using V.42/V.42bis error correction and data compression.

```
Top>Configure>Group> dialout
Top>Configure>Dialout> char branch no-v42
  1: branch          555-1234          dialout
  2: branch          555-6789          dialout
Enter the number of the dialout to use [2]: 1
Top>Configure>Dialout> char branch #2 no-v42
Top>Configure>Dialout> list
Name            Phone            Characteristics
branch          555-1234         no-v42
branch          555-6789         no-v42
```

Now, the NetBlazer will select and preconfigure a modem to not use V.42/V.42bis error correction and data compression prior to calling the branch dialout destination in the above example.

If you do not want a particular line to use V.42/V.42bis error correction and data compression when answering an incoming call, you can assign the no-v42 characteristics group created in the above example to that line.

```
Top> line
Top>Configure>Line> char line15 no-v42
```

Note that if a particular tag is included in a line characteristics group, the modem command string associated with that tag is sent to the modem whenever the line is reset.

Creating New Function Tags

If you need to control a particular function of a modem that does not have a defined function tag in the MODEMCAP.TXT file , you can add your own function tag and associated modem commands to one or more of the modem definitions contained in the MODEMCAP.TXT file. However, if you are not quite sure how to proceed, contact your technical support representative for assistance.

CHAT.TXT File Definition

The CHAT.TXT file automatically controls the conversation which takes place between the NetBlazer and a remote NetBlazer (or some other device) when attempting to establish a packet mode connection to exchange IP datagrams. This file uses a special programming language called chat script which allows the NetBlazer to respond correctly to prompts from the remote system.

The CHAT.TXT file may include multiple chat scripts to allow the NetBlazer to call and establish a packet mode connection with different types of systems. Each chat script starts with a label which consists of colon (:) immediately followed by the chat script name. For example, :ics is the label assigned to the chat script used to control the log-in exchange between two NetBlazers.

Chat Script Commands

Generally, each command line in a chat script starts with a command letter and a state number followed by the command parameters which are separated by commas. Any of the following chat script commands may be used:

send **s<num>,<tmo>,<tmo-st>,<next-st>,<string to send>**

This command allows the NetBlazer to send a character string to the remote system.

The <num> parameter is the state number of this command.

The <tmo> parameter is the timeout value in seconds.

The <tmo-st> parameter is the next state number if there was a timeout.

The <next-st> parameter is the next state number if the send was successful with no timeout.

The <string to send> parameter is the string to be sent to the remote system.

expect **e<num>,<tmo>,<tmo-st>,<next-st>,<string to expect>**
Expect-case **E<num>,<tmo>,<tmo-st>,<next-st>,<string to expect>**

These commands cause the NetBlazer to stop and wait for a string of characters from the remote system. The **expect** command is not case sensitive, whereas the **Expect-case** is case sensitive and the characters received must exactly match the <string to expect> parameter.

The <num> parameter is the state number of this command.

The <tmo> parameter is the timeout value in seconds.

The <tmo-st> parameter is the next state if there was a timeout.

The <next-st> parameter is the next state if the expected string was received prior to timing out.

The <string to expect> parameter is the string to wait for from the remote system.

or **o<next-st>,<string to expect>**
Or-case **O<next-st>,<string to expect>**

These commands must immediately follow an **expect**, **Expect-case**, **or**, or **Or-case** command. This is how the NetBlazer can respond to one of several different expected strings and process them appropriately. The **or** command is not case sensitive, whereas the **Or-case** is case sensitive and the characters received must exactly match <string to expect>.

The <next-st> parameter is the next state if the expected string was received prior to timing out

The <string to expect> parameter is the string to wait for from the remote system.

return **r<num>,<ret-val>**

This command is the end state for a chat script. It returns a value which indicates the success or failure of the connection.

The <num> parameter is the state number of this command.

A <ret-val> of zero (0) indicates that the chat was successful and that packet mode is enabled remotely. A non-zero <ret-val> indicates failure.

uname **u<num>,<tmo>,<tmo-st>,<next-st>**
password **p<num>,<tmo>,<tmo-st>,<next-st>**

The **uname** and **password** commands are how the chat script transmits the user name and password. Unlike the **send** command which always transmits the same string, these commands get the string to send from the interface information.

The <num> parameter is the state number of this command.

The <tmo> parameter is the timeout value in seconds.

The <tmo-st> parameter is the next state if there was a timeout.

The <next-st> parameter is the next state if the expected string was received prior to timing out.

get-challenge **c<num>,<tmo>,<tmo-st>,<next-st>**

The **get-challenge** command reads the crypto challenge from the remote system. Usually control is transferred to this state after a successful expect of "challenge." This command reads the crypto challenge and feeds it into the crypto challenge processing code where a response is generated. The next state is almost always a **send response** command.

The <num> parameter is the state number of this command.

The <tmo> parameter is the timeout value in seconds.

The <tmo-st> parameter is the next state if there was a timeout.

The <next-st> parameter is the next state if the challenge was received prior to timing out.

send-response **y<num>,<tmo>,<tmo-st>,<next-st>**

The send-response command is used to transmit a response to a crypto challenge. This command always follows a get-challenge command.

The <num> parameter is the state number of this command.

The <tmo> parameter is the timeout value in seconds.

The <tmo-st> parameter is the next state if there was a timeout.

The <next-st> parameter is the next state if the challenge response was sent prior to timing out.

Chat Script Examples

The simplest chat script is one that immediately returns a success code. This would be used when a NetBlazer must call a system that does not require a login chat sequence (the called system has SLIP or PPP already running on the port and will never prompt for a user ID or a password). Here is such a script:

```
:null
r1,0
```

A more complex chat script is the ics chat script used by the NetBlazer to establish a packet mode connection with another NetBlazer.

```
:ics
e1,10,30,2,login:
```

Start the ics chat script and wait 10 seconds for a “login:” prompt from the remote NetBlazer. If the prompt is received, transition to state 2. If 10 seconds expire without receiving the login prompt, go to state 30.

```
u2,5,100,3
s3,5,100,4,\r
```

Send the user ID and a carriage return (\r) to the remote NetBlazer.

```
e4,5,30,5,password:
p5,5,100,6
s6,5,100,7,\r
```

Wait 5 seconds for a “password:” prompt. If 5 seconds expire without receiving the password prompt, go to state 30. If the prompt is received, send the password followed by a carriage return to the remote NetBlazer.

```
e7,5,30,99,Packet mode enabled
o32,login:
o20,challenge:
```

Wait 5 seconds for a string from the remote NetBlazer. If 5 seconds expire without receiving a string go to state 30. If a “Packet mode enabled” string is received, the login was successful; go to state 99. If a “login:” prompt is received, go to state 32. If a “challenge:” prompt is received, a crypto challenge is pending, go to state 20.

```
c20,5,30,21
e21,5,30,22,response:
y22,5,100,23
s23,5,100,7,\r
```

Accept the crypto challenge, and respond to the “response:” prompt by sending back the proper crypto response. Then, go back to state 7, and check for a “Packet mode enabled” string from the remote NetBlazer.

```
s30,5,100,31,\r\r
e31,10,100,32,login:
u32,5,100,33
s33,5,100,34,\r
e34,5,100,35,password:
p35,5,100,36
s36,5,100,37,\r
e37,5,100,99,Packet mode enabled
o100,login:
o50,challenge:
c50,5,100,51
e51,5,100,52,response:
y52,5,100,53
s53,5,100,37,\r
```

Something went wrong in the login process. Try sending a couple of carriage returns and repeat the whole process again. If unsuccessful this time, go to state 100 and return a failure code.

```
r99,0
r100,1
```

State 99 returns a success code indicating that the NetBlazer has successful established a packet mode connection with the remote NetBlazer; whereas state 100 returns a failure code indicating that something went wrong during the login sequence.

Software Update Procedures B

If you are currently using an earlier version of the NetBlazer system software, you can use any of the following three methods to update your NetBlazer system software.

- Use the **diskcopy** command to make a working copy of the new NetBlazer system diskette with your current system configuration. Refer to *Using Diskcopy to Update Your System* on page 1.
- Use the **update** command to download the latest version of the NetBlazer software from Telebit via a dial-up FTP session as described under *Using Update to Download NetBlazer Software* on page 2. To use this method, you must have at least one modem attached to your NetBlazer.
- Use the **ftp** command to download the latest version of the software from Telebit over the Internet as described under *Using FTP to Download NetBlazer Software* on page 5.

If you have any problems updating your system software using any of the above methods, call Telebit's Technical Support at **1-800-TELEBIT** or **408-734-5200**.

Using Diskcopy to Update Your System

To create a new working copy of the NetBlazer system diskette with your current system configuration using the **diskcopy** command, perform the following steps.

1. Start the NetBlazer using your current working copy of the NetBlazer system diskette, and log in as "root" or as some other user with system configuration privileges.
2. Format a new high-density diskette, and make a backup disk copy of the new system distribution diskette. Store the original NetBlazer system diskette in a safe place.
3. Save your current system configuration to your backup copy of the system diskette by entering a **configure save** command at the command prompt.
4. If you have customized your CHAT.TXT or MODEMCAP.TXT files, you will need to apply your modifications to the new files using a text editor on an MS-DOS system, or use FTP to transfer the files to a system with a text editor, modify the files and then transfer the files back to your NetBlazer using FTP.

5. Reboot your NetBlazer by entering a **reboot** command at the command prompt, and log in again as “root” or as some other user with system configuration privileges.

6. If an UPDATE.CNF file is included on the distribution diskette enter **source update.cnf** at the command prompt to update your current configuration.

7. Save your current system configuration to the system diskette by entering a **configure save** command at the command prompt.

Using Update to Download NetBlazer Software

Caution: *Before attempting to update your NetBlazer software, make sure that you have a backup copy of your current NetBlazer system diskette.*

To download NetBlazer software via a dial-up FTP session using the **update** command perform the following steps:

1. Start your NetBlazer using your current working copy of the NetBlazer system diskette, and log in as “root” or any other user with configuration privileges.

2. Go to the Configure Dialout command level and list the tbupdate dialout record.

```
NetBlazer:Top>Configure>Dialout> list tbupdate
      Name: tbupdate
     Phone: 1-408-745-3700     Characteristics: dialout
   Timeout: 10 minutes
   maxwait: 0 seconds
   minwait: 0 seconds
```

3. Verify that the phone number for this record uses the proper prefix and is formatted correctly for your telephone system. If necessary, change and save the telephone number using the **dialout phone** and **configure save** commands. For example, to change the telephone number prefix to a 9, do the following.

```
NetBlazer:Top>Configure>Dialout> phone tbupdate 9-408-745-3700
NetBlazer:Top>Configure>Dialout> list tbupdate
      Name: tbupdate
     Phone: 9-408-745-3700     Characteristics: dialout
   Timeout: 10 minutes
   maxwait: 0 seconds
   minwait: 0 seconds
NetBlazer:Top>Configure>Dialout> save
```

4. Enter **update** at the NetBlazer command prompt. This is an alias for ftp 143.191.20.1.

```
NetBlazer:Top> update
Type ^] (decimal 29) to return to NetBlazer
Trying 143.191.20.1:21...
FTP session 0 connected to 143.191.20.1
220 tbupdate FTP version 1.0 ready at Fri Jan 18 11:00:48 1991
```

5. Log in as “anonymous” for this FTP session. No password is required.

```
Enter user name: anonymous
```

6. Get the README file for information concerning the latest release.

```
ftp> get README
```

7. To display the README file, escape to the NetBlazer command prompt, and enter a **more README** command.

```
ftp> enter escape character ctl-]
NetBlazer:Top> more README
```

8. After reading the contents of the README file, enter a **continue** command to resume the FTP session.

```
NetBlazer:Top> continue 0

ftp>
```

9. Enter the following FTP commands to switch to binary mode and transfer the latest version of the NetBlazer software.

```
ftp> binary
200 Type I OK
ftp> get fred
200 Port command okay
150 Opening data connection for RETR /fred
Get complete: 529336 bytes in 218 sec (2400/sec)
226 File sent OK
ftp> get token001.mcd
200 Port command okay
150 Opening data connection for RETR /token001.mcd
Get complete: 30066 bytes in 25 sec (1100/sec)
226 File sent OK
```

10. Switch back to ASCII mode and get the MODEMCAP.TXT, CHAT.TXT, DEFAULT.CNF and UPDATE.CNF files.

```
ftp> ascii
200 Type A OK
ftp> get modemcap.txt
200 PORT command okay
150 Opening data connection for RETR /modemcap.txt
Get complete: 4987 bytes in 7 sec (700/sec)
226 File sent OK
ftp> get chat.txt
200 Port command okay
150 Opening data connection for RETR /chat.txt
Get complete: 1537 bytes in 2 sec (700/sec)
226 File sent OK
ftp> get default.cnf
200 Port command okay
150 Opening data connection for RETR /default.cnf
Get complete: 1187 bytes in 1 sec (600/sec)
226 File sent OK
ftp> get update.cnf
200 Port command okay
150 Opening data connection for RETR /update.cnf
Get complete: 177 bytes in 0 sec (100/sec)
226 File sent OK
```

Caution: *If you have customized your CHAT.TXT or MODEMCAP.TXT files, you will need to apply your modifications to the new files using a text editor on an MS-DOS system, or use FTP to transfer the files to a system with a text editor, modify the files and then transfer the files back to your NetBlazer using FTP.*

11. When you are finished transferring files, quit FTP and reboot your NetBlazer.

```
ftp> quit
221 Goodbye!
FTP session 0 closed: EOF
NetBlazer:Top> reboot
```

12. Log in again as “root” or as some other user with system configuration privileges.

13. If an UPDATE.CNF file was provided, enter **source update.cnf** at the command prompt to update your current configuration.

14. Save your current system configuration to the system diskette by entering a **configure save** command at the command prompt.

Using FTP to Download NetBlazer Software

Caution: *Before attempting to update your NetBlazer software, make sure that you have a backup copy of your current NetBlazer system diskette.*

To download NetBlazer software over the Internet via a FTP session, perform the following steps:

1. Start your NetBlazer using your current working copy of the NetBlazer system diskette, and log in as "root" or any other user with configuration privileges.
2. Make sure that your NetBlazer is configured to reach an authorized Internet domain name server.
3. Enter the following **ftp** command at the Sessions command level.

```
NetBlazer:Top>Sessions> ftp ftp.telebit.com
Type ^] (decimal 29) to return to NetBlazer
Trying 143.191.3.1:21...
FTP session 0 connected to ftp.telebit.com
220 apache FTP server (SunOS 4.1) ready.
```

4. Log in as "anonymous" for this FTP session.

```
Enter user name: anonymous
331 Guest login ok, send ident as password.
Password:
230 Guest login ok, access restrictions apply.
ftp>
```

5. Go to the /pub/netblazer/current-release directory.

```
ftp> cd /pub/netblazer/current-release
250 CWD command successful.
ftp> pwd
257 "/pub/netblazer/current-release" is current directory.
```

6. Get the README file for information concerning the latest release.

```
ftp> get README
```

7. To display the README file, escape to the NetBlazer command prompt, and enter a **more README** command.

```
ftp> enter escape character ctl-]
NetBlazer:Top> more README
```

8. After reading the README file, enter a **continue** command to resume the FTP session.

```
NetBlazer:Top>Sessions> continue 0

ftp>
```

9. Enter the following FTP commands to switch to binary mode and transfer the latest version of the NetBlazer software.

```
ftp> binary
200 Type set to I.
ftp> get fred
200 PORT command successful.
150 Binary data connection for fred (143.191.10.28,1040) (529336 bytes).
Get complete: 529336 bytes in 368 sec (1400/sec)
226 Binary Transfer complete.
ftp> get token001.mcd
200 PORT command successful.
150 Binary data connection for token001.mcd (143.191.11.5,1042) (30066 bytes).
Get complete: 30066 bytes in 13 sec (2200/sec)
226 Binary Transfer complete.
```

10. Switch back to ASCII mode and get the MODEMCAP.TXT, CHAT.TXT, DEFAULT.CNF and UPDATE.CNF files.

```
ftp> ascii
200 Type set to A.
ftp> get modemcap.txt
200 PORT command successful.
150 ASCII data connection for modemcap.txt (143.191.10.28,1041) (14325 bytes).
Get complete: 14713 bytes in 9 sec (1500/sec)
226 ASCII Transfer complete.
ftp> get chat.txt
200 PORT command successful.
150 ASCII data connection for chat.txt (143.191.10.28,1042) (1513 bytes).
Get complete: 1584 bytes in 2 sec (600/sec)
226 ASCII Transfer complete.
ftp> get default.cnf
200 PORT command successful.
150 ASCII data connection for default.cnf (143.191.10.28,1043) (1158 bytes).
Get complete: 1187 bytes in 2 sec (500/sec)
226 ASCII Transfer complete.
ftp> get update.cnf
200 PORT command successful.
150 ASCII data connection for update.cnf (143.191.63.254,1036) (172 bytes).
Get complete: 177 bytes in 1 sec (100/sec)
226 ASCII Transfer complete.
```

Caution: *If you have customized your CHAT.TXT or MODEMCAP.TXT files, you will need to apply your modifications to the new files using a text editor on an MS-DOS system, or use FTP to transfer the files to a system with a text editor, modify the files and then transfer the files back to your NetBlazer using FTP.*

11. When you are finished transferring files, quit FTP and reboot your NetBlazer.

```
ftp> quit
221 Goodbye!
FTP session 0 closed: EOF
NetBlazer:Top>Sessions> reboot
```

12. Log in again as "root" or as some other user with system configuration privileges.

13. If an UPDATE.CNF file was provided, enter **source update.cnf** at the command prompt to update your current configuration.

14. Save your current system configuration to the system diskette by entering a **configure save** command at the command prompt.