The information in this guide may change without notice. The manufacturer assumes no responsibility for any errors which may appear in this guide.



The revision date for this manual is July 15, 1994.

Part Number: 900-037

WARNING

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC Rules. These limits are designed to provide reasonable protection against such interference when operating in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if not installed and used in accordance with this guide, may cause harmful interference to radio communications.

Operation of this equipment in a residential area is likely to cause interference in which case the user, at his or her own expense, will be required to take whatever measures may be required to correct the interference.

Warning: Changes or modifications to this device not explicitly approved by Lantronix will void the user's authority to operate this device.

Cet appareil doit se soumettre avec la section 15 des statuts et règlements de FCC. Le fonctionnement est subjecté aux conditions suivantes:

(1) Cet appareil ne doit pas causer une interférence malfaisante.
(2) Cet appareil doît accepter n'importé quelle interférence reìue qui peut causer une opération indésirable.

# Note

This Reference Manual is printed on 8 1/2 by 11 paper. It is designed to be placed in a 3-ring binder for maximum shelf-life and for easy retrieval. If you would like an official Lantronix binder for this document, please fill out the enclosed Product Warranty Card, write "Binder" in the lower right-hand corner of the card and Lantronix will ship you a binder for this manual.

# ETS/EPS Reference Manual

## For Lantronix Terminal/Printer Server Products

# Contents

## 1 Introduction

## 2 Concepts

# 3 Configuration

## 3 Configuration (cont.)



## 4 User Sessions



## 5 AppleTalk Host Setup



## 6 LanManager Host Setup

# 1 Introduction

# 1 Introduction

The Ethernet Terminal Server (ETS) is a device that can connect computing equipment such as personal computers, terminals, modems, printers, etc. – devices that were not built to be networked – to an Ethernet network. The ETS supports a serial device on each of its ports and the Ethernet simultaneously. It is configurable and can offer its attached devices as services to the network, and conversely, can provide connections to other services on the network.

The Ethernet Printer Server (EPS) provides the same functionality as the ETS with the addition of one parallel port for connections to a Centronics or Dataproducts-compatible printer. For an EPS4, the parallel port is located on port 5. For an EPS12, the parallel port is located on port 13.

The EPS1 and EPS2 are compact versions of the EPS4 - they have 1 serial port and 1 or 2 parallel ports, respectively. They are different from the other EPS and ETS units, in that they cannot be used to initiate connections to hosts - they can only accept connections **from** hosts. They are intended solely for outputting data to locally attached devices. Throughout this manual, the terms "ETS" and "Server" refer to both the ETS (8,16) and EPS (1,2,4,12) units unless otherwise noted. Any discussion of sessions and login-related features (dialback, menu mode, etc.) are not applicable to the EPS1 and EPS2.

Five industry-standard network protocols are supported:

- AppleTalk

  AppleTalk allows networked Apple Macintosh computers to see devices attached to the ETS and access them as any other networked printer. A utility run from Macintoshes can be used to configure the ETS remotely.

- LanManager

  Lan Manager allows networked printer access to devices attached to the EPS. Systems running OS/2 and Windows NT can access the devices over the DLC or NetBios protocols.

- Local Area Transport (LAT)

  LAT is a protocol developed by Digital Equipment Corporation for local network terminal connections and is supported on almost all DEC operating systems. It provides both logins to hosts and host-initiated print spooling.

- NetWare

  Netware provides networked printer access to devices attached to the ETS. It also allows users to download system files from NetWare hosts and log into the ETS via NetWare for remote configuration.

- TCP/IP

  Support includes Telnet, Login, DNS, and the RTEL and LPR printing systems. The Telnet terminal protocol is supported on most UNIX systems. It is an easy to use interface that creates terminal connections to any networked host supporting Telnet. Rlogin is another protocol that allows you to initiate a TCP/IP login session. The ETS supports the use of Domain Name Servers, allowing a network name server to translate text node names into numeric IP addresses. RTEL and LPR allow host-initiated print requests to be sent to devices attached to the ETS.

Each port on the terminal server can run multiple sessions (with a combination of LAT and TCP/IP sessions) to communicate with any host. The user can automatically connect to a host when the user turns on his or her equipment. While all sessions keep running concurrently, only one is visible on the screen at a time, and the user can easily switch between them.

# About This Manual

The rest of this manual describes the features and commands of the ETS. In addition, the Glossary explains terms used throughout this manual. Appendix A contains information on SNMP support provided by the ETS products.

- The remainder of this chapter explains some terms used throughout the manual, lists the ETS features, and explains the command line interface.
- Chapter 2 explains the major concepts involved in the ETS' operation - LAT, TCP/IP, Remote Configuration, Modem Control, Multisessions, NetWare, and AppleTalk.
- Chapter 3 shows how to configure the ETS for normal operation.

*Installation and Cabling are covered in the ETS/EPS Installation Guide. They are not covered in this manual.*

- Chapter 4 gives an introduction to the ETS for end users.
- Chapter 5 covers the setup needed on LAT hosts to utilize the ETS.
- Chapter 6 covers UNIX host setup, for RTEL and LPD printing.
- Chapter 7 covers NetWare fileserver setup needed to print to the ETS units.
- Chapter 8 covers AppleTalk setup needed to print to the ETS units.
- Chapter 9 covers LanManager setup needed to print to the ETS units.
- Chapter 10 shows the ETS command set in detail, including syntax, options, errors, and examples.

Throughout this manual, the terms “ETS” and “Server” refer to all variants of the ETS servers (ETS8, ETS16, EPS4, EPS12, EPS1, EPS2). Some commands apply only to the EPS (or not to the EPS1 and EPS2) and are so noted.

If you need to contact Lantronix, contact your local distributor or refer to the section at the end of this manual.

# Terms

In this manual, the following terms are used to describe parts of a network. See the Glossary for more detailed explanations of these terms.

- **Host** A computer, sometimes referred to as a CPU, attached to the network. The term “Host” is generally used for interactive computers, i.e. ones that users can log into.
- **Node** Any intelligent device directly connected to the Ethernet network such as a host, an Ethernet printer, or a terminal server. All are devices that have their own Ethernet addresses. Devices connected to the ETS’ serial ports are *not* nodes.
- **Service** A resource you can access, either locally or across the network. For example, a host is a service that terminals can connect to. The ETS also can offer attached printers and modems as services.
- **Session** A logical connection to a service. A typical session is a terminal connected to a host through the terminal server.
- **Local Mode** The ETS’ user interface. It is used to issue configuration and session management commands and to establish sessions with services.

# ETS Features

## DEC Compatibility

The ETS supports LAT and TSM/NCP, making it fully compatible with most DEC operating environments.

## TCP/IP and UNIX Compatibility

Almost all UNIX systems support Telnet. It is an established industry standard. UNIX systems generally implement Rlogin, as well; due to security considerations, it may not be enabled at your site. Support for Domain Name Servers is also provided. Network gateways, SLIP access, and a local domain name are all supported. In addition, IP security can be enabled to control access to the server.

## NetWare Support

The ETS can be configured to appear as a print node to other NetWare fileservers. Queues can be configured to transparently forward print jobs to the ETS. The ETS can also be used as a dedicated print server device. In addition, the ETS can be configured and logged into from a NetWare fileserver, and can also download its executable file from a fileserver.

## AppleTalk Support

The ETS can provide Ethernet access to Apple Macintoshes for serially attached laser printers. (The EPS1 and EPS2 can also provide Macintosh access to parallel printers.) ETS print services appear in the Macintosh Chooser window like other printers on the network. The ETS can also be configured remotely from a Macintosh via the MACCON utility.

## LanManager Support

The ETS can be configured to appear as a print node to other LanManager nodes. Supported systems include OS/2 and Windows NT. Windows for Workgroups is also supported, but there are significant restrictions.

## Connectivity

The ETS connects terminals directly to the network. This conserves physical ports on the host, frees the terminal from being tied to only one host, and simplifies terminal cabling.

## Multiple Session Support

Users can have multiple outgoing LAT, Rlogin and Telnet sessions and can easily switch among them. Up to eight sessions are supported per connected terminal. Each session can be connected to any host and can use any supported protocol.

## Load Balancing

If a LAT service is provided by more than one node, the server chooses the least busy one to connect to. This helps balance CPU usage and provides better user response times.

## Small Size

The small, attractive case is designed to fit into an office environment. The ETS is silent, as there is no fan.

## Easy Configuration

A simple but powerful command interface is provided for both users and system managers. The ETS' operating code is downloaded automatically on power-up, making software upgrades as easy as copying a file. The Flash-ROM ETS units (including all EPS1/EPS2's) have operating software stored permanently on-board, so they do not need to download code unless new versions become available. Both versions can also be configured to request a downloaded configuration file at boot time as well.

## Easy Use

The server's local mode supports command line editing, command line recall, and command completion. An extensive Help facility is also provided.

## Host Initiated Transfers

The ETS may be configured to provide its attached devices as services to other nodes. This allows hosts to share printers and modems. TCP/IP, NetWare, AppleTalk, LanManager, and LAT hosts can queue jobs to ETS services simultaneously. The optional RTEL host software provides both printer backend access and a named pipe interface to the ETS from TCP/IP hosts.

## Security

Using group codes, the ETS can be configured to act as a filter to limit the user's knowledge of and access to services. Sessions can be automatically logged out when a port is disconnected or when a device is turned off. Ports can be made secure, preventing them from issuing privileged commands and giving them a limited view of the network. Privileges, ports, services, maintenance commands and the remote console can be password protected. A user may be allowed to lock his or her port and can specify the password that will unlock it.

The server also provides an IP security table, which allows the server manager to restrict incoming and outgoing TCP/IP connections to certain ports and hosts. This allows managers to restrict ETS access to a particular local network segment or host, for example. It also allows restrictions for SLIP access to the network.

Dialback ports are supported, so that managers can improve security with dial-in modem users. In addition, a menu mode is provided, so that managers can tailor which individual commands users are allowed to use.

## Remote Configuration

The ETS can be configured remotely using DEC's NCP and TSM facilities. A Telnet console port, which functions similarly to the NCP remote console, is also provided. In addition, the network login feature allows managers to log into the ETS from TCP/IP, LAT, NetWare, and AppleTalk hosts to configure the ETS.

Configuration utilities that run on NetWare and Macintosh clients are provided.

## SNMP Support

The server also supports the Simple Network Management Protocol (SNMP). SNMP can be used by network managers to monitor network load and error conditions. No special MIBs are supplied by Lantronix.

## Diagnostics

Diagnostics run on power up and can be run interactively to help troubleshoot network and serial line problems.

# Command Line Interface

The Command Reference section of this guide describes the commands available in the ETS' local command line mode. These commands control session management, port and server configuration, and other tasks. Ports can be set to *secure* to allow only a subset of the commands. In addition, a menu mode is provided so that users will not have to learn the ETS' command set.

Command line entry is both simple and powerful. You can enter up to 132 characters on a command line. Commands are executed when the Return key is pressed or when the command line exceeds 132 characters. Commands can be abbreviated to the smallest unique keyword; in many cases, to a single character.

## Command Line Editing

Table 1-1 lists the special keys used for command line editing. Note that some terminal devices might not provide the key codes that the ETS expects for the cursor arrow keys.

*Command line editing is disabled when the terminal type is set to hardcopy (for example, a printer).*

| Key | Purpose |
|---|---|
| <Return> | Causes the command line to be executed |
| <Delete> | Deletes the character before the cursor |
| <Ctrl-A> | Toggles insert mode (insert or overstrike) |
| <Ctrl-D> or <Ctrl-Z> | Logs you out of the server |
| <Ctrl-E> | Moves cursor to end of line |
| <Ctrl-H> or <Backspace> | Moves cursor to beginning of the line |
| <Ctrl-R> | Redisplays the current command |
| <Ctrl-U> | Deletes the entire current line |
| <Right Arrow> | Moves cursor right |
| <Left Arrow> | Moves cursor left |
| <Up Arrow> or <Ctrl-P> | Recalls the previous command |
| <Down Arrow> or <Ctrl-N> | Recalls the next command |
| !!<Return> | Executes the previous command |
| !*text*<Return> | Recalls the last command starting with *text* |

**Table 1-1: Command Line Editing Keys**

## Command Recall

Pressing the Up and Down Arrow keys (or Ctrl-P/Ctrl-N) allows you to display and edit the past nine command lines. New command lines are added to the list only if they are different from the previous command. Many UNIX systems have this "history" feature as well, although it is usually activated using the exclamation point ( **!** ). The ETS supports the ! and !! characters as well. Typing !! at the command line performs the previous command again. Typing ! and one or more characters attempts to recall the last command that began with those characters.

Recalled command lines may be edited normally using the command line editing keys described above. Pressing the <Down Arrow> key moves forward through the list. For example, pressing <Up Arrow> twice followed by the <Down Arrow> key displays the last command executed (provided you have executed at least two commands). If there are no more commands in either direction, a blank line is displayed.

## Command Completion

When command completion is enabled, the server fills in the remainder of a command or option when a space or tab character is typed. If the partial command is ambiguous, the terminal will beep to prompt you for more characters. For example, if "Bro<sp>" is typed, the server will fill in "adcast" and will wait for the additional command parameters. See the **Set Port** command description for more details.

## Context-Sensitive Help

*See Help on page 10-12 for more information.*

Context-sensitive online help is available at any point. You may type "HELP" by itself for overall help, "HELP <command>" for help on a specific command, or you may type a partial command line and then "?" (question mark) to get help for what is appropriate at that particular point.

# 2 Concepts

# 2 Concepts

This chapter of the Reference Manual introduces the major concepts involved in the ETS' operation.

There are 2 major functions provided by the ETS - outgoing sessions (interactive connections to other nodes), and incoming connections to services (printer and modem support, etc.). All session-related topics are covered first; NetWare, AppleTalk and LanManager discussions are later since they do not provide outgoing session support.

Services, discussed here, are the basic method of connecting to the ETS from any host or another server (**including** non-LAT hosts). In general, a service is required on the ETS before any job or connection queueing will take place. See the "Server Management" chapter for details on creating and using services on the ETS. The EPS1 and EPS2 come with some services preconfigured; the ETS8/16 and EPS4/12 do not.

# Network Protocols

A basic description of network protocols is useful before discussing the ones supported by the ETS. A network protocol describes the data contained in Ethernet packets, and how to interpret that data. The network protocols provided by the ETS are completely separate, other than the use of the Ethernet data layer. Figure 2-1 shows the protocol stacks supported by the ETS.

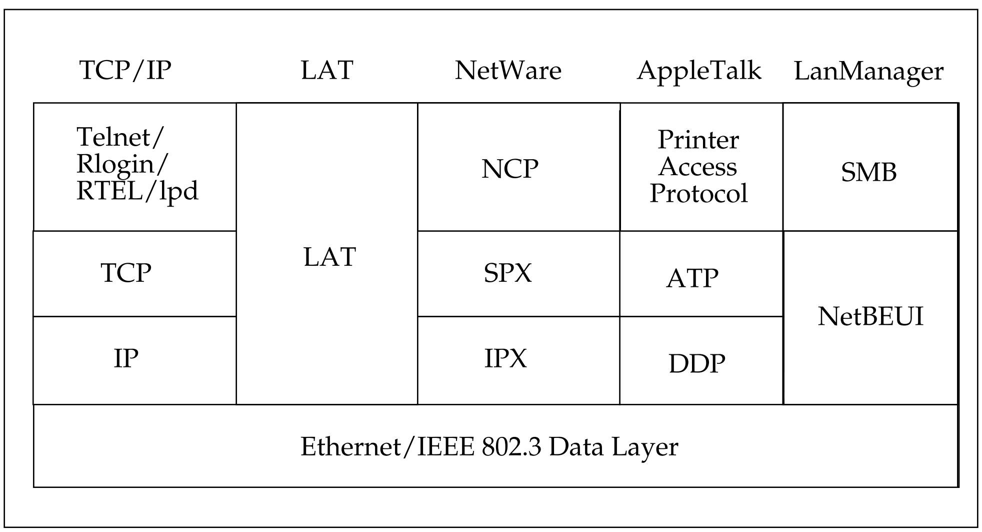


**Figure 2-1: ETS Network Protocols**

There are 2 different "Ethernet" frame formats, and sub-formats of one of those:

- What is typically called "Ethernet", is technically referred to as Ethernet v2. This format has a protocol type in the frame header, specifying which protocol stack will process the packet. This is the default frame type for most TCP/IP, LAT, and MOP/NCP protocol stacks. NetWare stations can be configured to use this format as well.
- The other format, IEEE 802.3, has a length field where the Ethernet v2 protocol field is. It then generally includes protocol information after the length field. This protocol information is in the form of a Service Access Point, or SAP. This SAP can then be used to determine which protocol stack will receive the packet.
- 802.3 frames can also have a SNAP SAP, which then includes a protocol field like Ethernet v2. Other than the extra 5 bytes of header, these frames are the same as Ethernet v2. AppleTalk uses the SNAP format by default; NetWare stations can be configured to use SNAP.
- NetWare 2.x and 3.x stations, by default, do not use any of these frame types. Their "native mode" format does not adhere to either the Ethernet v2 or 802.3 header format, and its use is being gradually phased out.

# LAT and Services

Digital Equipment Corporation's LAT (**L**ocal **A**rea **T**ransport) networking software is designed to make local area networks easy to access and manage. LAT software is built around the concept of **services**. A service may be provided by a dedicated device, such as a printer, or by a network host. The word **node** in this User's Guide refers to a device on the network that allows one or more services to be accessed – such as a computer you can log in to. The ETS is also a node since it can offer services and logins.

Services have names, and there can be more than one service on a network with the same name. For example, you could attach eight modems to an ETS and set up the service **modem** to use ports 1-8. A user who typed **Connect Modem** would not necessarily know which exact modem port he or she would be connected to. Note that if a service is provided by the local ETS and also a remote node, it is not guaranteed that a connect done locally will go to the local service (although this can be explicitly requested). In general, all services offered by the ETS will be associated with one or more serial (or parallel) ports; exceptions to this will be noted later. Figure 2-1 shows an example of a network with services.

**Figure 2-2: Example of Network Services**

Nodes advertise their services to the network by broadcasting occasional messages about them. These messages, referred to as **multicasts**, contain the node's name and its list of services. By monitoring these multicast messages, all hosts on the network know what nodes and services are available and can provide this information to their interactive users. The Show Services and Show Nodes commands display this information.

The LAT multicast messages contain a **rating** for each service offered. This rating is an estimate of how busy that service is. Ratings range from 0 to 255; 0 means the service is unavailable, while 255 means the service is available and has no current users. Ratings for a given service may change over time. The rating for a computer accepting logins will generally change often as its workload changes. Ratings for a modem might be only 0 or 255, when it is or is not being used, respectively. (In the example above, the server with eight modems attached will advertise that the service "modem" is available (a 255 rating) until all eight modems are in use (a 0 rating).

Service ratings may concern even casual users, since they are used to determine which service a user will be connected to when a choice could be made. For example, if a user types **Connect Hub** and five nodes offer service **hub**, the user will be connected to the least busy node automatically. In the case mentioned above, where both the local ETS and a remote node offered the same service, the ratings determine which node will service the connect attempt.

*You can connect to a specific node using the Node and Destination parameters. See the* ***Connect*** *command on page 10-9 for more details.*

The other major network management feature LAT implements is the concept of **service groups.** Each port on the ETS (or any terminal server) and service on the network can be thought of as belonging to one or more groups. When a user or device requests a service connection, the LAT host will check to see if the groups that the requester belongs to match up with those of the service he's requesting. If any group number is common to both the requester and the service, the connection attempt will continue. If there are no common group numbers, access is not allowed. Note that there may be additional access restrictions, such as password protection, on the service.

As an example, suppose Bob is logged in to port 4 on his ETS and the server manager has given port 4 access to groups 1, 7, 13, and 105. Bob, or anyone else using port 4, can only access services that have one of those group numbers. Suppose Bob wishes to access a modem on a different server. If the server the modem is attached to allows access to groups 8, 12, 16, 42, and 105; Bob will be allowed to use the modem because he and the modem service have group 105 in common. See the "Set/Define Port Authorized Group" and "Set/Define Server Group" commands for more information.

Group numbers also are useful to nodes because each node only needs to pay attention to multicasts that involve its users' groups. As a result, groups can hide services that would otherwise be visible. This allows the server manager to hide services from a set of ports, and saves on network processing by the ETS.

*Remote control can be disabled; see* ***Set Port Remote Configure*** *on page 10-42.*

Setting up and managing services and groups is discussed in more detail in the Server Management chapter. ETS' used in the LAT environment support **remote configuration** of server ports by VMS hosts. This feature of the LAT software allows VMS commands that change a user's "terminal line" to take effect and be implemented by the server. For instance, if user Bob connects to a VMS host and types in the command **Set Terminal/Speed=2400**, the speed will be changed on the host and the host will ask the ETS to change the "real" terminal line (in this case, Bob's serial port on the server). This feature will support changes to parity and flow control as well.

LAT is significantly different from the other protocols below in two important ways:

- LAT is not routable. There is no way to divide LAT networks into smaller subnetworks, and use routers to reduce traffic between nodes.
- LAT is a timer-driven protocol - packets are expected at certain intervals, and the protocol cannot adapt to slow network links dynamically.

For these two reasons, LAT traffic over wide areas is typically carried inside (or **encapsulated**) in TCP/IP or IPX/SPX packets. The latter 2 protocols are fully routable, and can handle wide-area, slow network links. If LAT is used unencapsulated across slow links, session timeouts and errors are very likely.

# TCP/IP (Telnet, Rlogin, SLIP, RTEL)

## IP Addresses

Every TCP/IP node on a network has an address unique to that network. This is in addition to the Ethernet Address, which is unique across all hardware anywhere in the world. This IP address provides the information needed to forward packets across multiple networks, if necessary. The address is of the form **xxx.xxx.xxx.xxx**, where each **xxx** is a number from 0 to 254. An example address is 192.0.0.1. A unique address has to be specified on the ETS before any of the TCP/IP functionality is available. Setting it is covered in *TCP/IP Server Parameters* on page 3-3.

## Telnet

Telnet is an industry-standard network protocol. The theory behind Telnet is that a user anywhere on the network can access a foreign host and start a terminal session. The user will not have to worry about what kind of computer or operating system he is dealing with, and the remote host will know nothing about the type of terminal the user is using. This is essential for the wide-area networks that Telnet was designed for. Recall that LAT is for local-area networks only - networks that support Telnet can (and do) stretch for thousands of miles and contain thousands of hosts. Telnet accomplishes this compatibility with the concept of a Virtual Terminal. Any Telnet connection looks the same from both sides of the network. Figure 2-3 illustrates the concept.

**Figure 2-3: Telnet Connections**

All the terminals on the network can access all the hosts, but do not need to (nor can they) know exactly what type of hardware the host is. Theoretically, terminal 4 in the example above could establish a Telnet session with host Phred, and would see the same type of session as would terminal 5 which is directly connected. Phred, the CPU, will treat terminal 4 as if it were also directly connected.

A Telnet connection begins with both sides of the connection assuming the other terminal is "dumb," i.e. knows little about characters, fonts, cursor movement, etc. The two sides then negotiate options to determine the capabilities of their peer. For example, a terminal might support full cursor-control characters. Once the connection is established, the terminal and host may agree to support the extra characters. This all happens transparently to the user, who sees only a login screen and none of the negotiation process.

## Rlogin

In addition to Telnet, ETS users can use the **Rlogin** application to connect to remote TCP/IP hosts. Rlogin is similar to Telnet, but it can be configured to allow trusted users to log into a host without password verification. **For this reason, Rlogin can be a security problem**, and might not be enabled at your local site. *TCP/IP Server Parameters* on page 3-3 explains how to set up Rlogin access and discusses the methods of user verification. Like the UNIX version of the command, the ETS Rlogin command allows the user to specify a username other than his current one to use for logging in. See the Rlogin description in the command section for more details. Both Telnet and Rlogin support the port's **Termtype** field, which allows hosts to discover the type of terminal device automatically.

Both Telnet and Rlogin can be forced to connect to specific ports. By default, each connects to a pre-set port number to access a remote host's Telnet or Rlogin server. The ETS allows the specification of a port number with the connect request to connect to a different port number. For example, if a host is set up to listen to Telnet requests on port number 1034, that port number can be specified in the Telnet command from the ETS (instead of the Telnet default, 23). See the Telnet and Rlogin commands in the command reference chapter for more details.

*See **Show Hosts** on page 10-85 or your local host documentation for more details.*

Unlike LAT network service (with its multicasts), TCP/IP implementations generally have no knowledge of remote hosts until a connection is attempted. Therefore, there is no TCP/IP parallel to the Show Services commands. A host is not searched for until the actual connection request is made. The UNIX operating system does support the **rwho** facility to allow hosts to see each other without an active connection, but it is not always implemented, enabled or supported. Hosts running rwho send out occasional broadcasts containing the host name, address, and user information. The ETS will listen for these **rwho** broadcasts and keep track of which UNIX hosts are available.

## Subnetworks

Subnetworking is a major difference between the network protocols. The LAT hosts on a network can all see each other, while hosts using TCP/IP can frequently see only a fraction of the hosts on the whole network. Usually, a TCP/IP internet is broken down into "networks," where a host is able to see only the hosts on its own network (or subunit). TCP/IP networks then rely on **routers** (or **gateways**) to transfer network traffic to hosts on other networks. Routers are typically connected to two or more networks, and will pass, or route, TCP/IP messages across network boundaries. The ETS can be told explicitly which hosts are the gateways for the local network. If no gateways are currently specified, the ETS will listen to broadcasts on the network from other gateways. It will listen to routing-protocol packets (such as RIP, OSPF, IGP, etc.) to decide which hosts are acting as gateways.

*See **Set Server Gateway** on page 10-55 and **Set Server Secondary Gateway** on page 10-68 for more details.*

The ETS decides at connection time whether the desired TCP/IP host is on the local network segment with the help of the **subnet mask** on the server. This mask tells how much of the IP address is the network address, and is applied to both the ETS' and the remote host's IP addresses. If the resulting addresses are identical, the connection is deemed local and the host is contacted directly. If not, the connection attempt and all subsequent messages to this host will be directed to the ETS' gateway host for forwarding. The subnet mask, if not set explicitly, will be set based on the type of IP address the ETS is using.

*See **Set Server Subnet** on page 10-72 for more details.*

## Name Resolving

TCP/IP hosts generally have an alphanumeric host name, such as **Phred**, as well as the numeric IP address, such as 192.0.1.99. For this reason, the ETS supports a local host table or **domain name servers** - hosts that can translate text host names into the numeric addresses that are needed to actually connect. The text names make remembering long strings of address numbers unnecessary. For example, suppose user Bob wishes to log in to a host "alex.weasel.ctcorp.com." The other hosts on the network have no knowledge of which piece of hardware this name refers to, so the ETS must translate it into a numeric address (such as 199.25.172.6). After the name is translated (or **resolved**, as it is commonly called) the connection can be attempted. If there is no local host table or name server to provide text name translation, or if the name server(s) can not translate the name, the ETS cannot connect to the host. In this case the host's numeric IP address would have to be used for the connection. Also, if the ETS has not been told which local host is acting as the name server (using the Set Server Nameserver and Set Server Secondary Nameserver commands), it will assume none is available and the connect request will likely fail.

The ETS also allows the setting of a default **domain name**, to be appended to any host name for which name resolution is done. In the example above, if the server Bob was using had a default domain name of **weasel.ctcorp.com**, then Bob could type **telnet alex** and the ETS would attempt to resolve **alex.weasel.ctcorp.com**. This name checking occurs for all ETS commands that require text name resolution, such as Telnet, Rlogin and Ping. If the connect destination name ends with a period (".") the domain name is not added to the hostname for resolution.

*See **Set Server Domain** on page 10-54 for more information.*

## Local Host Table

The ETS enhances name resolution by providing a simple **host table**. The host table is a list of node names and the IP addresses they correspond to. Using the **Set Telnet Hosts** command, the server manager can add hosts to this table. This allows the user to use a text name described in the table in place of an IP address. This eliminates the need to remember IP addresses, and is especially useful for sites that do not have a domain name server or where a few hosts are accessed frequently.

You can display the currently known TCP/IP nodes on the network with the **Show Hosts** command. It will display all the known IP nodes and whether they are from the local host table, the name server, rwho broadcasts, or were resolved by an actual connect attempt.

If name serving is enabled, the name server will be asked for the node's IP address only if the local host table does not contain the information. This means that if the information in the local host table is incorrect, it will be used, even if it disagrees with what a nameserver might report. The **Clear/ Purge Telnet Hosts** commands are used to remove hosts from the host table. The local host table does not have to be consistent with the name server, but we recommend that they be kept consistent to avoid confusion.

For example, we might have node **alex.weasel.ctcorp.com** at address 192.0.1.119. You might define **alex.weasel** as an alias to address 192.0.1.119. Users typing **Telnet alex.weasel** will be connected as if they had typed the numeric IP address. In addition, **alex** could be mapped to the same IP address, and users could simply type **Telnet alex** and be connected. Both names could be placed in the table (and either Telnet command above would work) but care should be taken that both (or all) are updated if any one name is changed.

Host table aliases should be easy to type, remember, and recognize. They are limited to 64 characters, with a maximum length of 16 characters per segment (i.e. **alex.weasel.cticorp** is legal, while **alexweaselcticorp is not**.)

The ETS will also listen to nodes' **rwho** broadcasts and attempt to add host table information based on the name and IP address in the rwho message. If these rwho addresses are Cleared from the table, they will reappear the next time the host sends out an rwho message. They will also reappear if a user attempts to connect to one of them.

## Serial Line Internet Protocol (SLIP)

The SLIP protocol allows hosts to communicate via TCP/IP over a serial line instead of Ethernet. Due to the point-to-point nature of serial lines, only 2 hosts can communicate over a single line. SLIP is most frequently used for dialup modem lines, where a SLIP-equipped TCP/IP host can dial into the ETS, begin a SLIP session, and thus gain access to the network attached to the ETS. The remote host can start sessions to the ETS or any other host on the network, subject to security and other login restrictions. If multiple TCP/IP hosts are connected to the ETS via SLIP connections, each SLIP host can also access any of the other SLIP hosts, with the ETS routing packets between the serial lines. For example [Figure 2-4]:

*See **Set/Define SLIP** on page -81 for more details.*

**Figure 2-4: SLIP Example**

In the example diagrammed here, assume all hosts have a subnet mask of 255.255.255.0. All 4 hosts will appear to be on the same network segment, and all 4 will be able to start TCP/IP connections with each other. The ETS in this case "routes" packets between the serial lines and the Ethernet without help from the hosts involved. Note that since all the hosts are on the same network, the ETS is not routing in the true sense of the word, and any "real" IP routers on the network will not be affected by the SLIP sessions. The ETS is only directing network traffic to the proper serial port. For each packet the ETS receives it checks to see if it is for the ETS, then for the SLIP hosts, and finally sends it out onto the Ethernet if need be. If a packet off the Ethernet is sent to the ETS and there is no SLIP connection for it, the packet is discarded. Also, if a packet is received off a SLIP port that is not from the host that is supposed to be there, the packet is discarded. This prevents other hosts from connecting to an active SLIP line and being able to access the ETS' Ethernet. The ETS will also provide ARP replies for its SLIP hosts, so that hosts on the Ethernet can resolve their IP addresses.

There are 2 restrictions on the ETS' SLIP support - first, there can be only one host at the end of the serial line. The ETS will only forward packets to and from the host registered on the SLIP line. Thus, the ETS will not route packets to and from a SLIP network (multiple hosts). Second, all the hosts connected via serial lines must be on the same network and subnet as the server. See the subnet entry in the Glossary if you are unsure of this. The SLIP section of the "Server Management" chapter covers SLIP setup and restrictions.

## Simple Network Management Protocol (SNMP)

*See **Set/Define SNMP** on page 10-84 for more details.*

The server also supports the SNMP network protocol. SNMP allows hosts on the network to query other hosts for counters and network statistics and change some parameters on those hosts. The form of these requests is documented by RFC 1098 (see the Glossary for a description of RFC's). The list of items that can be queried and/or set and the type of data (integer, string, etc.) are documented in various Management Information Bases (MIBs). These MIBs cover a variety of things, such as parallel port status and counters, or IP address resolution tables.

The ETS will respond to queries for unknown MIBs with a "not in MIB" error to the requesting host. See RFC's 1065, 1066, and 1098 for more information on SNMP queries and the structure of the MIBs. SNMP also provides for the setting the values of certain items in the MIBs as well as retrieving the values. The ETS has a local SNMP security table to restrict or prevent unauthorized SNMP configuration.

The ETS will also generate limited forms of 3 of the SNMP traps. Traps are sent to a host when an abnormal event occurs on the ETS. Currently, the ETS will generate a Coldstart trap when it first boots, and will send a Linkup trap when the startupfile (if any) has been read from a host and normal operation commences. If a startupfile has been configured but the download fails, the ETS will send an Authentication trap. In all 3 cases, the trap will be directed to the IP address of the loadhost for the ETS. If a loadhost has not been specified (Flash-ROM based units, for example), the traps will not be sent. The ETS will not generate traps other than the cases listed here.

## Reverse Telnet Enhancements (RTEL)

When the ETS (or any terminal server) provides a service to a LAT host, the connection is often referred to as Reverse LAT, in that “normal” LAT connections are logins from the terminal server to a host. Hosts request this connection when they wish to access a service, such as a printer or modem, connected to the server.

There was no real standard (or even analogy) for this type of data connection in the TCP/IP world until now. To add this functionality to the TCP/IP environment, the ETS comes with special software for your UNIX host. Reverse Telnet, much like DEC’s Reverse LAT, provides the software capability for a UNIX host to initiate connections to the ETS. These connections can be through a back-end program for a printer, or through a named pipe. For example, you can print files from your UNIX host to a printer attached to the ETS through your host’s **lp** or **lpr** print system. The ETS’ services do not care what type of hosts are using them. Multiple jobs from any of the supported protocols can be queued up simultaneously on any service.

*See Chapter 8, UNIX Host Setup, for setup information and more details.*

## LPD Support

The ETS and many UNIX systems implement the Berkeley remote printing protocol (LPD). This protocol makes it very easy to add print hosts to a system. Typically the only steps that are needed to add the ETS as a print host are to add the ETS' IP address to a host table and add the ETS' service name(s) as a print queue. No special host software or other configuration are needed, and the ETS' services can be accessed via the normal lpr commands on the host. Note that print jobs can be forwarded multiple “hops” in the network - if you only want 1 host to know about the ETS print queues, all other hosts can be configured to forward their print jobs to that one host, who then forwards them to the ETS for printing.

## UUCP Support

UUCP (Unix-Unix Copy Protocol) is a utility that allows Unix hosts to transfer files between machines. It is most commonly used for electronic-mail transfers, and can be run over Ethernet or serial links. If run over Ethernet, it typically is run via TCP/IP. When run via TCP/IP, a handler for UUCP typically listens on TCP port 540 of the host; when a connection to this port is made, a UUCP connection is assumed and the UUCP negotiation begins. The ETS provides support for incoming (serial/modem line to network host) and outgoing (network host to serial/modem line) UUCP connections.

Incoming UUCP can be accomplished with the server handling the serial line and creating a network connection to the UUCP machine. Since UUCP typically does not provide Telnet interpretation, the **Connect TCP** command would be used from the ETS to provide a raw TCP data stream to the UUCP host. Furthermore, the connection must be made to the UUCP port (540). Dedicated connections to the UUCP port can also be made with TCP environment strings in conjunction with the Port Dedicated parameter and the :xxx port number feature of TCP/IP connections.

Outgoing UUCP connections can be made in a similar manner. The UUCP host must be configured such that outgoing UUCP network connections are to the ETS. The host connects to the UUCP port (540) on the ETS. If the ETS has been configured with the Set Server UUCP command, it will attempt to connect to a specific service, named “UUCP”. This service needs to map to one or more ETS serial ports, possibly with modems attached. The chat script (L.sys or Systems files) on the UUCP machine then provides the commands to dial the modem and log into the remote UUCP machine. The chat script needs no knowledge of the ETS command set or service connection sequence.

## TCP/IP Utilities and Commands

Several commands have been added to help TCP/IP usability:

- netstat — Displays the status of the routing tables and current TCP and UDP sessions.
- send — Enables users on Telnet session to send explicit Telnet control characters, such as Abort Output (AO), Are You There (AYT), etc.
- finger — Displays users on local and remote hosts. The finger command by itself will show the users on the ETS. If the finger command is given with a parameter such as bob@hydra, information regarding user **bob** on TCP/IP host "hydra" will be shown. The username can be omitted, in which case all the users on the remote host will be displayed. If the host cannot be reached or accessed, the finger command will fail. To see a list of processes on the ETS, issue the command "finger finger."

## LAT and TCP/IP Connectivity

The service capabilities of the ETS also allow translation between LAT and Telnet protocols. For example, a user on a LAT host might see and connect to an ETS offering service **ernie** as a LAT connection. This "service" might actually be a Telnet connection to host **ernie**. In the other direction, a TCP/IP user can use the UNIX **Telnet** or **Rlogin** commands to form a connection to a LAT host through the ETS. Both cases can take place transparently to the user. Note what this implies - the TCP/IP host does not need LAT software, and the LAT host does not need TCP/IP software to communicate with each other. The two different directions (Telnet to LAT and LAT to Telnet) are set up differently, however. One should keep in mind that each such connection consumes ETS memory and processing resources - it is treated as another "port" and user on the ETS.

*Chapter 8, **UNIX Host Setup**, explains how to set up Telnet to LAT and LAT to Telnet translations.*

## Logging into the ETS

The mechanics of the protocol translator also add the ability to actually "log in" to the server from either a TCP/IP host (using Telnet or Rlogin) or a LAT host (using the Set Host/LAT command). For example, if the server's IP address is 192.0.1.92, using Telnet to connect from an IP host to this address will connect and return the local prompt as if the user had logged in on a hardwired port. Also, the server can advertise itself as a "service" to LAT hosts, i.e. LAT hosts can connect to the server and can also obtain the local prompt. Incoming logins can be selectively disabled and/or password protected via the Set Server Incoming command described later. The IP security table can also be used to restrict logins to the ETS. Ports for network logins can be "preconfigured" with a standard set of characteristics for each login. See the section entitled "Preconfiguring Virtual Ports" later in this manual for further information on template ports.

Once an incoming (LAT or Telnet) connection has been started, the user can use the server as if he or she were connected on a local port - the user can start outgoing connections, connect to local services, configure the server, etc. Note that this login capability gives TCP/IP users a convenient way to remotely "manage" the server - managers can simply log in and become privileged, change server parameters, and log out again.

# NetWare

Novell's NetWare software provides a way to link computers together and provide file, printer and modem sharing. It is typically used to network DOS-based PC's, but is starting to appear under Unix and other operating systems. NetWare is built around file servers, which handle user logins, provide network resources, and control security. At least one file server (PC, Unix host, etc) is required in any NetWare environment. NetWare users typically have to log into a file server to enjoy the networked benefits - until they do, they cannot access the shared resources. This is different from AppleTalk and LAT, where services are generally visible to all users by default.

The ETS supports a significant subset of the NetWare functionality, most notably print spooling. Fileservers can be configured to send queued print jobs to printers attached to the ETS. No special software is required on the fileserver, and configuration uses the standard PCONSOLE utility or the included configuration utility. Any users or applications who can use the NetWare print queues can spool jobs to the ETS. The queues are managed by the file servers, but the ETS must occasionally query them for pending jobs. For this reason, the ETS also knows how to log into the file server to access the print spooler. The ETS will try to connect to all file servers on the local network (subject to access lists, below) to check for pending jobs. The file servers will not initiate print jobs to the ETS - the ETS will poll all the fileservers for queues that are configured to use it.

*See EPSCON Netware Utility on page 3-47 and Define Server Netware Host on page 10-62.*

NetWare support also allow logins from fileservers to the ETS (for configuration) and file downloading (to download the system software at boot time).

# Networking

Each NetWare node uses its hardware address as its node ID, so no address needs to be configured. The ETS gets all the networking information it needs from periodic broadcasts from any NetWare routers on the net. It will learn its own network number and routes to non-local file servers. If the ETS needs to contact a non-local host, it will automatically select a router to use. No further configuration is needed.

The NetWare protocol can utilize various Ethernet frame formats. By default, NetWare nodes use what is termed “native mode” 802.3, which is slightly different from the usual Ethernet v2 frames. Fileservers can also be configured to use full 802.3 headers, either in SNAP or IPX SAP mode. In general, these distinctions do not matter to ETS users - the ETS will listen for all frame types, and use the correct one to talk to the appropriate fileserver.

The different frame types are treated as different networks, and thus each has a different network number. If there is only one frame format in use on the LAN, the ETS will use the network number for that frame type. If there are multiple frame types, the ETS needs to use a different network number for each frame type it wants to advertise itself on. This is a restriction of the NetWare protocol itself. The ETS does this by creating a new, unique “network number”, and advertising itself as a “router” to this internal network. Any nodes or fileservers that need to communicate with the ETS use this new network number, and treat the ETS as a router to that network.

If this behavior is not desired, the ETS can be forced to use only one frame type (and thus not need to create an internal network number).

## Access Lists

Since NetWare networks can be quite large, with hundreds of file servers, a way to restrict the ETS print polling is needed. If left unrestricted, the time to poll each server would make printing incredibly slow and inefficient. By using the access lists (the Set/Define Protocol Netware Access command) you can control which file servers the ETS will and won't contact for print jobs. This can also be used to restrict the NetWare users who can see and use the ETS printing resources. Note that the file servers have no control over the ETS' access lists, so they will be unaware if they can't spool jobs to the ETS due to access mis-configuration.

# AppleTalk

AppleTalk is a protocol used primarily by Apple Macintoshes to access network resources such as file servers and printers. AppleTalk is also available for Unix, NetWare, and other operating systems. The AppleTalk protocol can be run over Ethernet (EtherTalk), Token Ring (TokenTalk) and LocalTalk, a medium speed network type built into every Macintosh. These various network media can be connected with AppleTalk routers to allow all nodes to communicate with one another. The AppleTalk protocol supports features like file sharing and security, as well as printing support.

The ETS supports only the Printer Access Protocol (PAP) and therefore cannot create outgoing AppleTalk sessions - only incoming print requests are accepted. Any services on the ETS with AppleTalk enabled will show up as LaserWriters in Macintosh Chooser windows (subject to zones - see below). Users who select the ETS service and then print will have their jobs forwarded to the ETS for printing. They will be queued if the print port on the ETS is in use or otherwise unavailable.

AppleTalk printing is a bit different from other protocols. There are standard Macintosh drivers provided for specific Apple printer types, such as LaserWriters and ImageWriters. There are no plain ASCII line printers, by default - jobs are converted into PostScript (for laser printers) or bitmaps (for ImageWriters, etc.) when printed. For this reason, not all printers are usable by Macintoshes. The ETS supports only laser printer devices under AppleTalk. In addition, the Macintosh client will need to query the printer about status, so only laser printers that will reply to these interactive PostScript requests can be used. This also prevents printing AppleTalk jobs to the EPS4/12 parallel ports, as they do not allow bi-directional communication. The EPS1 and EPS2 parallel ports do support this “Bitronics” mode, provided the attached laser printer also implements it.

AppleTalk provides for dynamic node addressing, allowing a node to choose its address at boot time. It will send network packets to the other nodes to avoid choosing an already-used node ID. Any node can also discover its network number by listening for AppleTalk router broadcasts; if none are heard, a default network number is chosen. Network numbers and zones (see next) are configured in the routers, leaving the nodes mostly configuration-free. The only configuration the ETS supports is specifying a zone name other than the default. In addition, the ETS will save its discovered zone/network/node IDs in permanent memory, reducing traffic at reboot time. The ETS' AppleTalk address may change across boots, however, in response to any network changes. Due to these no-configuration features of AppleTalk, most AppleTalk devices are truly “plug-and-play” - they can be powered up and are used right out of the box.

## Zones

Zones are arbitrary groupings of AppleTalk nodes. Zones are used to organize resources into groups that are easier for users to understand. For example, a college may organize zones around departments, i.e. **Math Dept.**, **Physics Dept.**, etc. They allow users to sift through large numbers of nodes by choosing those groups they are familiar with regardless of the organization of the network. In general, zones need not have a correlation to physical or network location, thus any node can declare itself a member of any single zone.

Zones originate from and are configured on AppleTalk routers. One zone on each network will be chosen by the router as the default zone. If no zone name has been previously defined on the ETS, or if the defined zone is no longer valid, the ETS will join this default zone. The ETS can be placed in a different zone with the Set/Define Protocol AppleTalk Zone command. Each time the ETS is booted, or when its zone is changed, it will verify the zone name with a router. If there are no AppleTalk routers on the network, there will be only a default zone that all nodes are members of and no other configuration is needed.

## Name Binding Protocol (NBP)

NBP is used by AppleTalk to advertise resources, such as printers and fileservers, on the network. Any resource that other users can access will have a set of NBP information that other nodes will need to know. The items in the Chooser windows reflect the various NBP resources that are visible on the network. NBP and the Chooser organize resources by three levels: name, type, and zone. Names are arbitrary strings assigned by users, such as **Kathy** or **MyPrinter**. Types are generic types of resources, such as **Macintosh IIci**, and **LaserWriter**. Zones, mentioned above, are a collection of nodes on the network. A typical Macintosh's NBP information might be (Kathy, Macintosh IIci, Accounting) for the name, type and zone, respectively. A service offered by the ETS called **MyPrinter** that had AppleTalk enabled and is located in the **Engineering** zone would have an NBP description of (**MyPrinter, LaserWriter, Engineering**). If LaserWriter resources in the Engineering zone were selected in the Chooser, one of the resources shown would be the ETS' MyPrinter service. The Chooser sends out queries whenever a new zone or resource type is selected, so if an ETS service that is expected to show up does not, there is a configuration problem somewhere.

The NBP type LaserWriter designates a PostScript printer. Nodes printing to printers of type LaserWriter assume that the printer supports PostScript. Thus, care must be taken to only attach PostScript printers to ETS services with AppleTalk enabled and to disable AppleTalk on services that do not support PostScript printers. It is not possible to print to non-PostScript printers (ImageWriters, etc.) from a Macintosh via the ETS.

# LanManager

LanManager is a protocol used by several PC-based operating systems, notably OS/2 and Windows NT (and Windows for Workgroups). LanManager servers have been written for HP and Sun workstations. LanManager is based on the NetBios protocol. The ETS implements only enough of the NetBios protocol stack to provide print services to nodes - no interactive logins are allowed.

The ETS also implements the DLC printer protocol, typically used by HP laserprinters. This DLC operation is only supported under Windows NT, and does not provide queueing on the ETS. It is very straightforward to configure and use, however.

## Networking

NetBios is not a routable protocols, so all the nodes that the ETS can communicate with either have to be local, or a gateway has to provide an alternate method for transporting the NetBios data. Note that NetBios can be run over the TCP/IP protocol, but the ETS does not support this mode of operation.

Node lookups are done by text name - any resolving that happens takes a text resource name and resolves it into a hardware address. For this reason, node and resource names much be unique on the network, and the ETS will print an error message if any configuration that violates this is attempted.

# Sessions

When a user connects to a service on the network (LAT, Telnet, or Rlogin), he or she creates a **session.** A non-networked terminal with a dedicated line from the terminal to the CPU is typically limited to a single session–you must log out of the current session before you can connect to new host or service. With the ETS, however, this restriction is eliminated. A user can have several connections to various services at once, although only one is displayed on the screen at a time. Each separate connection is a session. Four important topics go along with our discussion of multiple session support.

First, there must be a way to leave an active session and return to the ETS' local prompt. This is accomplished with the **Break** key. Pressing it will return the server to local mode, leaving any sessions connected. In local mode, you can issue server commands to open another connection, configure your port, etc. If your terminal device has no Break key, you can configure a local switch key with the Set Port Local Switch command. You can also change who actually processes the Break key (the server, the remote host, or neither) by using the Set Port Break command. By default, the server handles the break key and does not pass it to the remote service.

Second, there are commands and keys that allow movement between sessions and from local mode to a session. The Backwards and Forwards commands, entered in local mode, move backwards and forwards through sessions. With the Set Port Forward and Set Port Backward commands, you can define keys to move forwards and backwards through sessions without returning to local mode. The key will be interpreted by the server and the current session will be changed appropriately. You can think of all the user's open sessions as being a list from the earliest to the most recently created. **Forward** refers to a more recent connection, and **Backward** refers to a session started earlier. The list is also circular, so going forward from the most recent session takes you to the earliest, and backward from the first session resumes the last. For example:(1) User Bob connects to host **Thor**. He works for a while, then, (2) breaks and connects to host **Duff**. He works some more and then (3) connects to host **Conan** to read mail. At this point, his session list (shown with the Show Sessions command) looks like this:

1. Thor
2. Duff
3. Conan

with Conan the **current session**. The current session is the one you are currently connected to, or the last one you were in before you entered local mode. Now, if Bob presses his Backward key while working on Conan, he will resume the session on Duff (without reentering local mode). Pressing it again puts him in the Thor session. Pressing the Forward key from the Thor session will move him forward to the Duff session, etc. Due to the circular list, pressing the Forward key from Conan will place Bob in the Thor session.

Note that unlike the Break key (which can also be set up with the Set Port Local command), no forward or backward keys are enabled by default. You must use the Forward or Backward commands explicitly or define the keys yourself.

A way to set the characteristics of your active sessions is also needed. This is the purpose of the Set Session commands. Each session can be configured to do various amounts of processing on the input character stream. For example, while transferring a file, you would not want the server to interpret and strip out flow control characters or add server messages to the data. You can also redefine the Delete and Return keys for Telnet connections, depending on the host you are connected to. The latter is especially useful for TCP/IP hosts that expect both Carriage Return and Line Feed, for example, or only Line Feed. Note that you can only set the current session - you must move through the session list if you wish to change a non-current session.

*See **Disconnect** on page 10-10 for more information.*

Finally, users need to be able to control and disconnect sessions from local mode. For example, if a session on a remote host freezes up or gets stuck executing code, you can Break out of the session and disconnect it. You can also resume work on a session after you've returned to local mode with the Resume command. Both commands can affect any of the active sessions, not just the current one. Logging out of the server closes all active connections, as well.

*See **Set Server Session Limit** on page 10-70 and **Set Port Session Limit** on page 10-43 for more information.*

The number of active sessions a user can have on the server is limited by three factors - available server memory resources, and two software limits. There is a server-wide limit on sessions as well as a port-specific limit. The absolute maximum number of sessions per port is eight.

# Environment for TCP/IP and LAT

The ETS can be configured to set up sessions according to the host connected to. These environment settings can be supplied via the Connect command, or can be saved either as part of the Preferred or Dedicated hostname or as part of a virtual translation service. The available environment settings are discussed below. Note that most of the environment options mimic the Set Session command options.

The environment string is a sequence of key letters, each optionally prefixed by '+' or '-'. The keys are added after the ':' for Telnet or Rlogin connections, and are specified via the Environment option for LAT connections. The key-letters are:

- D +D = Backspace mode, -D = Delete mode.
- E +E = Local Echo mode, -E = Remote Echo mode.
- I I = Interactive mode.
- P +P = Passall mode, -P = Passthru mode.
- C +C = (CR becomes CRLF), -C = (CR becomes LF)

for Telnet connections the following are valid as well:

- T TCP mode (i.e. raw, uninterpreted data stream)
- R Rlogin protocol (sets port # to 513 if not already set).
- Q Queued (i.e. RTEL) connection to the target.
- nnn Optional port number.

Note that the "R" key allows Rlogin use where it would otherwise not be usable (for example, Preferred and Dedicated hosts). If the Rlogin option is used this way, you will probably want to set/define a username for the port, since the username is passed to the host as part of the Rlogin connection sequence.

The "Q" key allows users on the ETS to make outgoing RTEL connections. One possible use for this LAT to RTEL print translators - see the "VMS Host Setup" chapter for more details.

For example, to connect to a LAT modem service (named modem) with local echo and passthru modes [Figure 2-5]:

```
Local> CONNECT modem ENV +E-P
```

**Figure 2-5: Connect Command**

To connect to TCP host ALBERT on the FTP port (port 21) with local Echo and Carriage Returns changed to Line Feeds [Figure 2-6]:

```
Local> TELNET albert:21e-c
```

**Figure 2-6: Telnet Command**

To add environment to a port's preferred service [Figure 2-7]:

```
Local> DEFINE PORT 5 PREFERRED bilvax ENV +P+D
```

**Figure 2-7: Specifying Environment**

To set a dedicated Rlogin host, with a connection to TCP port 480 [Figure 2-8]:

```
Local> DEFINE PORT 5 DEDICATED TELNET lab_sgi:480R
```

**Figure 2-8: Define Port Dedicated Command**

In this case the Telnet part of the command indicates the TCP/IP dedicated field, as opposed to the LAT dedicated field. The R key letter will force an Rlogin session instead of Telnet.

Note that the key letters are not case-sensitive, and no white space is allowed in the environment string. Also, commands that counteract previous commands will overwrite the previous setting. For example [Figure 2-9], the following command will set remote echo, since the -E supersedes the earlier +E.

```
Local> telnet HOST:+E-P-E
```

**Figure 2-9: Telnet Command**

# Multisessions

Multisessions allows two windows and sessions to run simultaneously over a single serial line. Multisessions give the user the flexibility of controlling both sessions from the terminal rather than from the server. Using multisessions requires a multisession-capable terminal, such as the VT340, VT420, or equivalent. See your terminal device's manual for information on the presence and/or use of multisessions. Both multisessions can access all of the ETS' services, and are otherwise indistinguishable from an actual physical port. Both LAT and TCP/IP hosts (using Telnet only) are available from the multisession connection prompt. Note that the multisession protocol adds characters to the serial session to control the windows, and thus may produce a performance degradation if multiple sessions are running.

In general, starting multisessions on a port involves configuring the attached terminal to run multisession mode (a setting such as **Multiple Sessions on Comm1**) and then executing the command **Set Port Multisession Enabled** on the ETS. This will start multisessions running on the port and provide both sessions for the terminal. If the terminal device is to remain in multisessions mode permanently, a **Define Port Multisession Enabled** should also be performed. Once your terminal and the ETS' port are configured as multisession devices, you should be able to use both windows and sessions simultaneously. Both sessions are independent of each other - you can log out of one while the other session runs unchanged. The "new" multisessions will show up (using the Show Users command) as a new virtual port on the server and will function as a **separate** port. This means that any configuration done on one port (or window) will not automatically be reflected on the "sister" port. The only exceptions to this are serial-specific configurations, such as baud rate, parity, etc.

When multisessions are started (or each time the user logs out of the port) the server will prompt for a service to connect to. This service can be a LAT service or Telnet host; typing just a Carriage Return will put the user at the ETS' Local**>** prompt. Multisessions are much like Autoconnect ports - users specify this preferred "service" when they connect. Users can exit or **<**Break**>** out of that session and begin another session to another host at any time.

A few notes are in order about multisession connections. First, the port will remain configured as a multisession port even *after* the user logs out of the port. Multisessions must be explicitly disabled (Set Port Multisession Disabled) or the port logged out to be turned off. Disabling multisessions on the terminal device will also disable multisessions and log the port out. Also, the port is placed in Autoconnect mode as part of multisessions, and will remain this way as well. If multisessions are disabled and then the port is logged out, the port's settings will return to their normal, saved state. Disabling multisessions logs out the "sister" port and closes all its connections as well.

When a service connection is attempted by multisessions, LAT services are tried first. A Telnet connection will be attempted if the LAT connection requested fails. This may have a small drawback - if you mistype the LAT service name, a Telnet connection will also be attempted; if you have a nameserver enabled for the ETS, it will be queried for the LAT host's IP address. This may show up as a small delay (3-5 seconds) after a failed LAT connect. Telnet is tried by default for any IP host attempts; Rlogin can be requested by adding the **:R** environment string (see above) to the host name.

# Menu Mode

For added security or a uniform command interface, system managers can set some or all ports to be in Menu Mode. Users logging into a port in menu mode will have the following text displayed on their terminal [Figure 2-10] instead of the local prompt.

```
          Menu Title Bar
1.     command1          19. command19
2.     command2          20. command20
       ...                   ...
18.    command 18        36. command 36

Enter Menu Selection>
```

**Figure 2-10: Menu Mode Display**

The users will not be able to enter text commands; they will only allowed to enter a numeric choice from the menu. Up to 36 commands can be provided for the users, including a command to exit menu mode and return to the interactive Local**>** prompt. Menuing frees users from having to remember ETS syntax and prevents them from entering privileged configuration commands, although it can allow them privileged access if so configured.

Configuring menuing is a two step process. Each port that is to be running in menu mode needs to be configured via the Define Port Menu command. The command entries themselves are configured with the Set/Define Menu command. A menu command of **Exit** can be specified, in which case the user will be allowed to exit to the command line to enter normal ETS command lines. This might be useful, for example, for users who prefer the menu for ease of use, but need to return to the command line occasionally. Menu mode can then be re-entered via the Set Port Menu command, or by logging out and re-logging in (if Menu mode was Defined on that port).

The current or saved menu tables can be displayed via the Show/List Menu commands. Individual menu entries and the whole table can be cleared via the Clear/Purge Menu commands.

# Modem Features

The ETS supports both incoming and outgoing modem connections. The server can be configured to offer a modem service to both LAT users and Telnet network users. Using the DSR and DTR signals from the serial lines (see the Serial Pinouts in the Install Guide) the ETS ports can both log out an attached modem after a connection exits and be logged out if the modem hangs up the phone line. Ports can also be configured to autobaud at login time, allowing multispeed modems to connect easily.

The ETS allows ports to be configured with the Dialup and/or DTRwait characteristics. **Dialup** will tell any LAT service that this port is connected to a modem; the remote node may decide to disallow the connection based on that flag for security reasons. The **DTRwait** characteristic will force the ETS to deassert DTR on a port until a user actually connects to the port. This connection can be either an interactive login from a device or a service connection, (either LAT or Telnet). Note that on some modems, having DTR deasserted will prevent the modem from answering an incoming call unless someone is connected to the port already.

The ETS also provides **Dialback** support, which requires port users to enter a trusted username and then allow the ETS to close the connection and attempt to reconnect via a specified modem command string. This ensures that only trusted usernames are used and that the usernames only log in from a particular phone number. If an invalid username is entered or the reconnect attempt fails, the connection is assumed to be insecure and the port is logged out. Dialback does not interfere with outgoing modem connects, so a modem can be restricted for incoming logins but still allow full outgoing access to network users. Ports configured for Dialback have a 30-second time limit on entering a username at login time, to prevent unauthorized users from denying access to others.

# Remote Configuration

See **Set Server Maintenance Password** on page 10-60 for information on remote access control.

Users with VMS hosts can remotely control their ETS using the NCP utility. The NCP program can both signal remote hosts to do specific operations across the network or form connections to the server. This is especially useful for Flash-ROM servers, which may not be in easily accessible locations. With the NCP program, the ETS can be rebooted and be forced to reload downloaded code. Access to remote control of the ETS can be protected by the maintenance password. The ETS also supports the TSM utility, which uses NCP to start login sessions with the ETS and allows files of ETS commands to be sent to the ETS.

For TCP/IP environments, the ETS allows a user to Telnet into the "console" TCP port to configure the server. Users connecting to this console port (port 7000) will be prompted for the ETS' login password, as if they had logged in via NCP. After entering the login password, the user can issue normal configuration commands. Users will still have to enter the privileged password to issue privileged commands. Note that connections to this port are not subject to the Set Server Incoming command, and thus managers can log into this port regardless of whether regular TCP logins are enabled.

Note that NCP, TSM, and the Telnet console all depend on the login password to log in properly. The default password is **access**, and can be changed via the Set/Define Server Login Password command.

The server also supports the SNMP network protocol. SNMP (for Simple Network Management Protocol) allows hosts on the network to query other hosts for counters and network statistics. See the discussion of SNMP in the TCP/IP section. In general, one host on a network will be running an SNMP application that queries the other hosts on the network and collects statistics, signals errors conditions, etc. The ETS will only respond to queries directed to it - it will not generate queries of its own.

# Event Reporting/Logging

See **Set Logging** on page 10-23 for details.

The ETS can be configured to report various errors and events to either the console serial port or to a network host. Reporting to a network host can be via TCP/IP (using the syslogd facility), LAT, or NetWare. Events that can be logged include logins, modem events, printer state changes, system boot/resets, or even all commands issued on the ETS itself.

# 3 Configuration

# 3 Configuration

This chapter explains how to configure your ETS for operation. Please refer to the Commands and Glossary chapters for more information on unfamiliar terms and commands. Remember that you can abbreviate commands and parameters to the fewest letters needed to identify them. Commands are occasionally abbreviated in the examples to help you get acquainted with the practice.

In addition to setting up the ETS, be sure to read the Host Setup chapter to configure your VMS or UNIX host to take advantage of the ETS' features.

Refer to the Commands chapter of this guide for more information on any of these commands. The difference between the "Set" and "Define" commands is important in the following examples.

**Set** commands are put into effect immediately, but do not remain in effect after the ETS is rebooted. **Define** commands work in the opposite manner--they remain in effect after the unit is rebooted, but do not take effect immediately.

Define Port and Define Slip commands do not take effect until the port is logged out (or a new SLIP session started); the remaining Define commands (Server, Host, Service, Menu, Dialback, and IPsecurity) take effect only after the server is rebooted.

The Save command saves current configurations made with Set commands; in other words, it "Defines" commands that were previously "Set".

# Initial Configuration

After powering the server up for the first time, you will want to set up the ports and server for everyday use. Most of the parameters discussed in this section will only need to be set once and can be left until a major change is needed. There are two major areas of setup that need to be done - setting up the server, and setting up the ports. Also, if any of the server's users will be using Telnet, the server's IP parameters will need to be specified. You may also find it easier to use Telnet after using the **Set/Define Telnet Hosts** command to set up the host table. Recall when reading the examples that you can abbreviate options to their shortest unique name, and that you can include more than one option (for Set and Define) on each command line. ETS commands are also case-insensitive (except for strings enclosed in quotes).

Even if you will only be using the ETS with terminals, we suggest you take a few minutes to look over the following examples of how the ports, server, and sessions can be configured because they can make management and normal use easy and convenient. Also, recall that the Save command will make permanent any changes you make to the server or ports. Simply configure them to your liking with Set commands and then save all the setups.

*Set/Define Server Privileged Password is discussed on page 10-65.*

# Privileged Commands

To prevent unauthorized users from changing server-wide characteristics, some commands are privileged. Privileged commands can only be issued by the superuser on the server, i.e. the manager. The Set Privileged command gives the user these superuser privileges. Only one port at a time may be privileged. If someone else is currently the superuser, he or she can be forced to give up control with the Set Privileged Override command. The privileged password will still have to be entered to become the superuser. The two cases are illustrated below [Figure 3-1 and Figure 3-2]:

*The default privileged password is* ***system****.*

```
Local> SET PRIVILEGED OVERRIDE
Password> (not echoed)
```

**Figure 3-1: Set Privileged Override Command**

This will force the privileged port to be changed to the current port. Otherwise, the command would be as displayed below:

```
Local> SET PRIVILEGED
Password> (not echoed)
Local>>
```

**Figure 3-2: Set Privileged Command**

When you are through using the privileged commands, we suggest you turn off privileges (see the Set Noprivileged command) so that you do not inadvertently change the server's or other users' settings. The default privileged password is "system", but can be changed with the Set/Define Server Privileged Password command.

# LAT Server Parameters

There are 2 LAT parameters that need to be specified in order for the server to communicate on the network. These are the server's identification string and name; the LAT network timers may also need to be changed for your network.

Setting the these parameters is straightforward. Select a unique and descriptive name for the server, as well as a more descriptive string for the identification string. The name may contain up to 16 characters and the identification string may contain up to 40 characters. For example [Figure 3-3]:

```
Local> DEFINE SERVER NAME "Server_1"

Local> DEFINE SERVER IDENTIFICATION "Biolab Server:
         Modem Avail."
```

**Figure 3-3: Defining Server Name**

Note that Define is used to make these changes permanent. If you are not planning on rebooting the ETS after configuration (rebooting is suggested) you will also need to change these parameters with "Set" to make the changes effective immediately. Also, recall that you must use quotes around strings that contain lower case letters, spaces, or punctuation.

The LAT parameters (Circuit, Keepalive, and Multicast timers as well as the Retransmit Limit) are set to default values at the factory. They should be changed only if you are sure they need to be. The factory settings are correct for most networks. If you do need to change one or more of these parameters, the commands might look like the following [Figure 3-4]:

```
Local> DEFINE SERVER CIRCUIT 60
Local> DEFINE SERVER RETRANSMIT LIMIT 5
```

**Figure 3-4: Changing LAT Parameters**

# TCP/IP Server Parameters

If you want TCP/IP connections (Telnet, Rlogin, LPR, reverse Telnet support), you must also configure the ETS with IP parameters. The ETS *must* have a unique IP address. If this was set at boot time (i.e., by BOOTP or RARP), it does not need to be reset. You can also define a nameserver host if you want text TCP/IP hostnames to be resolved, and a default domain name to use in case a domain is not specified. You can set a gateway host, which allows TCP/IP connections to other connected network segments. Finally, you can set a loadhost (if not done already) so that TCP/IP boots as well as MOP boot attempts will be attempted. For example [Figure 3-5]:

```
Local> DEFINE SERVER IPADDRESS 192.0.1.228
Local> DEFINE SERVER LOADHOST 192.0.1.33
Local> DEFINE SERVER NAMESERVER 192.0.1.29
Local> DEFINE SERVER DOMAIN "weasal.ctcorp.com"
Local> DEFINE SERVER GATEWAY 192.0.1.188
```

**Figure 3-5: Setting TCP/IP Parameters**

One of the most common problems encountered is that of duplicate IP addresses on the network. Signs of this problem are Telnet/Rlogin connections that fail soon after connecting, or ARP requests that do not find a host you know is present. If these problems occur, make sure that the ETS has a unique IP address on the network.

If the server either loses its IP address when booting or will not allow a new IP address to be configured, it is probably an indication that another host is using that address.

You can also configure backup hosts in case the primary nameserver or gateway fail [Figure 3-6]:

```
Local> DEFINE SERVER SECONDARY NAMESERVER 192.0.1.193
Local> DEFINE SERVER SECONDARY GATEWAY 192.0.1.195
```

**Figure 3-6: Configuring Backup Hosts**

Note that these commands were all done as Defines, so they will not take effect until the server is rebooted. Use the Set command to make them take effect immediately.

If your environment uses a few TCP/IP hosts regularly, you may wish to add hosts to the local host table so that text names do not have to be nameserved (and numeric addresses do not *have* to be used). The **Set/Define Telnet Hosts** commands add local host table entries [Figure 3-7]:

```
Local> DEFINE TELNET HOST beeny 192.0.1.23
```

**Figure 3-7: Define Telnet Hosts Command**

*TCP/IP host names are case-independent, whether used with the host table or with a nameserving host.*

Users who Telnet or Rlogin to host "beeny" will be connected to address 192.0.1.23, without needing network nameserving support.

The TCP/IP security table can be used to restrict incoming or outgoing TCP/IP connections. Managers can restrict connections by address or network on a port by port basis. See *IP/Network Security* on page 3-12 for more information on using the security table.

# NetWare Server Parameters

Configurable NetWare parameters include the following:

| | |
|---|---|
| ♦ Fileserver from which to load the download file | Define Server Netw Load |
| ♦ Netware Access lists | Set/Define Protocol Netware Access |
| ♦ Ethernet encapsulation types to listen to | Set/Define Protocol Netware Encapsulation |

A command is provided (Set Protocol Netware Reset) that forces the ETS to rescan all fileservers for pending jobs. If a NetWare host is rebooted, for example, and then has print requests queued up, the RESET command will force the ETS to re-examine the fileserver queues. This is also useful after configuring new queues on the fileservers, to ensure that the ETS recognizes queues directed at itself.

## Routing and Encapsulation Types

By default, the ETS will listen to all NetWare frame types; if more than one is in use on the LAN the ETS will create an internal network number and advertise itself as a router to this network. In general, this will allow all nodes and fileservers to access the ETS regardless of frame type. If this behavior is not desired, the ETS can be restricted to only one frame format (and thus not need the internal network and routing support). There are two commands that control this behavior:

- Set/Define Protocol Netware Routing disables or enables the creation and use of an internal network number.
- Set/Define Protocol Netware Encapsulation is used to control which of the frame types are used when routing is disabled. The choices are ETHER_II, NATIVE, and SNAP. These provide for Ethernet v2, Native mode, and 802.3 SNAP, respectively.

By default, Routing is enabled, and all applicable encapsulation types are supported. See Chapter *2, Concepts,* for more detail on the various frame types.

## NetWare Access Lists

The NetWare access lists are used to control which fileservers the ETS will query to get print requests from. Recall that by default, the ETS will query all fileservers on the local network for pending print jobs. Since NetWare networks can have hundreds of fileservers, a way is needed to reduce the number of queries from the ETS, both to reduce network traffic and to prevent long delays in servicing active queues. The access list contains the names of the fileservers to query, and/or the keywords **All** and **Local**. All means no restrictions, while Local means only those fileservers that are on the same network as the ETS. For example [Figure 3-8], the following commands will allow the ETS to query all fileservers on the local network, as well as the non-local server RFS2.

```
Local> SET PROTOCOL NETWARE ACCESS LOCAL
Local> SET PROTOCOL NETWARE ACCESS RFS2
```

**Figure 3-8: Setting Netware Access**

By default, the ETS will only query local fileservers (but this may still be too many in a large network). The Show/List Protocol Netware Access commands show the current and saved access lists. The Clear/Purge Protocol Netware Access commands remove items from the access list.

Since the fileservers wait to be polled by the ETS for print jobs, they cannot tell if the ETS is not polling them due to access restrictions. There will be no error messages if a valid fileserver cannot print to the ETS due to the access list.

# AppleTalk Server Parameters

The only configurable AppleTalk parameter is the zone that the ETS will belong to. If there is no AppleTalk router on the network, all devices will pick the same zone (“*”). If a router is present, the ETS will use the network’s default zone unless Set/Define Protocol AppleTalk Zone is used to specify a different zone name. An error will be returned (and the default zone used) if the zone could not be confirmed with a router. For example [Figure 3-9]:

```
Local> SET PROTOCOL APPLETALK ZONE “AcctZone”
```

**Figure 3-9: Specifying AppleTalk Zone**

*Zone names are case-sensitive; if lower case letters are used in the name, the name must be in quotes (“’s).*

The current zone name is visible in the Show Protocols screen. The Show Protocols AppleTalk screen gives even more detail, including the current network range and packet counts. Show Protocols AppleTalk Zones and Nodes will show currently available zones and other AppleTalk nodes, respectively. Note that a node will only show up if it is a router or is in direct contact with the ETS (i.e. printing to it or configuring it).

# PostScript Notes

Many printers (including all LaserWriters and other AppleTalk-compatible laser printers) use a printing language called PostScript. Unlike other printers, which typically accept ASCII characters and print them verbatim, PostScript is a programming language in addition to a printer protocol. Shapes and fonts can be defined as routines and reused on successive pages, multiple fonts and copies can be printed, text and diagrams can be rotated and shifted on pages, etc. PostScript is also an interactive language, where the printing host can query the printer for its knowledge about fonts and software versions. If the host expects to receive data from the printer (as is the case for **all** AppleTalk printing) the printer must be connected to a bi-directional port on the ETS. For this reason, AppleTalk printers cannot currently use the parallel port on the EPS4 and EPS12 units, as it is an output-only device. The EPS1 and EPS2 have bi-directional (Bitronics) parallel ports and thus can be used for AppleTalk printing. Note that the attached printer must also support the Bitronics mode. PostScript printing from Unix, LAT, and NetWare hosts, where bi-directional data flow is not a requirement, can generally use any parallel port.

PostScript is a verbose language; it is recommended that the printer and ETS use a baud rate of no less than 9600 to communicate over serial lines. A baud rate of 38400 (or 57600/115200, on the EPS1) is strongly recommended for high-output duty; see the printer's documentation for information on changing its serial speed. This does not apply to parallel-attached printers, as they will accept data as fast as possible. Also, many PostScript jobs contain 8-bit characters. The ETS is configured for 8-bit characters, but if the printer is not (the default on many laser printers is 7-bit) some characters will be printed incorrectly. Again, see your printer's documentation for details on how to change the setting.

Due to the interactive nature of PostScript, it is possible for the printer and host to get out of sync in their communication. Since the printer “interprets” the entire PostScript job and then prints the pages, it is possible for the ETS to complete the serial transfer and accept a new job while the printer is still digesting the last job. For this reason, an end-of-job character (ASCII 0x4, or CTRL-D) is used to end all PostScript jobs. Typically the host will send one at the end of the job and the printer will reply with one back when it is done processing. The ETS can be configured to force a Ctrl-D character enabling PostScript on the appropriate service. This is **strongly** recommended for all PostScript printer applications. At the start of each print job, the ETS will send the printer a Ctrl-D and wait for one in return. This guarantees that the printer is ready to accept new data when the job actually starts. Job loss and printer hangs are the usual result if PostScript is not enabled on the service and the printer cannot keep up with the job rate. High-speed printers may happen to work if the PostScript attribute is not enabled, simply because they finish the current job in the time it takes for the ETS to accept the next one.

# General Server Parameters

The following network-related options can be set at the manager's discretion:

- The **Announcements** flags tells the server whether or not to broadcast service messages to the network. Normally enabled, Announcement can be disabled if you do not want network users to see or connect to the local services. If no services are defined, no announcements are broadcast.
- The **Node** and **Host Limits** parameters are used to set how many remote LAT service nodes and TCP/IP hosts (respectively) the server will store information for. This can be set in large network environments to keep ETS network overhead down. The Host limit applies only to hosts that were not explicitly set in the host table (hosts seen using Rwho, for example).
- The **Incoming** parameter is used to prevent or allow LAT and Telnet/Rlogin access into the server. Logins can be allowed but forced to enter a password, as well. See below for a more detailed description of the command.

The remaining server parameters deal with managing the local users on your system. You may wish to set passwords for the privileged and maintenance accounts. For example [Figure 3-10]:

```
Local> DEFINE SERVER PRIVILEGED PASSWORD "SERVER"
Local> DEFINE SERVER MAINTENANCE PASSWORD ac067df
```

**Figure 3-10: Defining Server Passwords**

You also may want to require users to enter a password to log in to the server [Figure 3-11]:

*For more examples of defining passwords, refer to **Setting Passwords** on page 3-10.*

```
Local> DEFINE SERVER LOGIN PASSWORD "LOGGIN"
Local> DEFINE PORT ALL PASSWORD ENABLE
```

**Figure 3-11: Defining Login Password**

The default login password supplied with the ETS is **access.** (The default privileged password is **system**.) You can require only certain ports to enter a login password by changing the "**All**" above to specific port numbers.

You can set up server-wide port characteristics, such as enabling locking and broadcasting and setting users' session limits. For example [Figure 3-12]:

```
Local> DEFINE SERVER LOCK ENABLE BROADCAST ENABLE
Local> DEFINE SERVER SESSION LIMIT 3
```

**Figure 3-12: Defining Server Characteristics**

The two preceding commands allow port users to lock their ports, send broadcast messages to other ports and set the maximum number of sessions per port to three. Recall that as many Set/Define parameters can be placed on a line as can fit on the command line - up to 132 characters.

Finally, the prompt each user receives (the Local_x> prompt) is configurable in a variety of ways. For a basic prompt [Figure 3-13]:

```
Local> SET SERVER PROMPT "Server>"
Server>
```

**Figure 3-13: Configuring User Prompt**

The following options can be included in the prompt string:

| | |
|---|---|
| %p | substitutes the current port's name |
| %n | substitutes the current port's number |
| %s | substitutes the current server name |
| %D | substitutes the product name (EPS-4, ETS-8, etc) |
| %C | substitutes the string "Lantronix" |
| %S | substitutes the current session name |
| %P | substitutes a > if user is currently privileged |
| %% | substitutes a % |

Figure 3-14 displays a few examples of changed prompts.

```
Local> SET SERVER PROMPT "Port %n: "
Port 3: SET SERVER PROMPT "%D:%s! "
ETS-8:LabServ! SET SERVER PROMPT "%p%S_%n%P%% "
Port_5[NoSession]_5>%
```

**Figure 3-14: Prompt Examples**

Placing a space after the end of the prompt is recommended, as it makes reading and editing the command line much easier.

# Examples of Setting Server Parameters

With the **Set Server** and **Define Server** commands you can specify several options on a single command line. Refer to the Commands chapter for complete information regarding the commands used here.

## Setting Passwords

Passwords are case-independent, and must be alpha-numeric characters only. A password may be entered on a command line, but if it is required and not given, the server will prompt you for it. In this case, as you type the password, it is not echoed back to you. The server will ask you for the password twice to ensure you did not mistype it. If the two passwords do not match, the command aborts. Note that passwords are *not* case sensitive, even when specified within quotation marks [Figure 3-15].

```
Local> DEFINE SERVER LOGIN PASSWORD "Hobbes"
```

**Figure 3-15: Defining Login Password**

In this example, the password will still be set to **HOBBES** (all uppercase). If the password was not given on the command line, the previous example might have been [Figure 3-16]:

```
Local> DEFINE SERVER LOGIN PASSWORD
Password> hobbes (not echoed)
Verification> hobbes (not echoed)
Local>
```

**Figure 3-16: Defining Password Example**

Remember, if the server prompts you for the password, it will not be echoed to the screen as you type it and verify it. Also, you can abort the password-entering process by typing a Ctrl-Z at either password prompt.

*Changes made with Set commands take effect immediately but are not permanent. You must use the Define command to change boot parameters.*

## Resetting the Boot Parameters

If the software loadfile or load host address change, you will need to change the parameters for the next reboot. In Figure 3-17, the new load host will be at address 199.34.67.184 (assuming a TCP/IP style load), and the ETS will request the file "newload.sys."

*Quotation marks were used to preserve the case of the filename.*

```
Local> DEFINE SERVER LOADHOST 199.34.67.184 SOFTWARE "newl
```

**Figure 3-17: Changing Boot Parameters**

# Controlling Incoming Sessions to the ETS

The Set/Define Server Incoming command allows or denies incoming LAT or Telnet connections. It can also require incoming users to enter the server login password to log in off the net. The Set/Define Rlogin command restricts the use of the Rlogin command out of the server. Refer to Chapter 10, *Command Summary*, for the exact syntax and use of these commands. *Logging into the ETS* on page 4-1 explains in detail the mechanics of setting up incoming sessions. Note that users logging in from the network are much harder to trace and monitor than those on attached physical ports. Please make sure your ETS and network environment are reasonably secure if you allow network logins. Also, refer to *Preconfiguring Virtual Ports* (below) for details on configuring ports for network logins. Note that you can configure the TCP/IP security table to enable or disable network connections from certain hosts--see *IP/Network Security* on page 3-12 for more information.

# Preconfiguring Virtual Ports

Incoming LAT or Telnet/Rlogin logins to the ETS do not have a physical “port” associated with them, since they can appear and disappear at random. Therefore, each such connection receives a “virtual port” on connection, which then disappears after logout. These virtual ports are created from a “template port”. The template is the set of characteristics that every network login receives. Each user can then use the Set Port commands to customize his or her own port, but cannot save these individual ports.

The ETS provides a way to preconfigure virtual ports. If a Define command on port 0 is done it will set up default parameters to be provided for all the network logins. If, for example, a **Define Port 0 Preferred Albert** is done, all network logins will have their preferred services set to service **Albert**. Note that users can change or clear these characteristics on their own ports with their own Set Port command after login. A recommended use for the template port is to provide local switches to network logins, as they typically do not have any Break key to use after they connect. As a note, the Set Server Incoming (No) Password command controls whether network logins require a password. The Set Port Password command has no effect on network logins.

Show Ports and List Ports will allow users to see the “template” (port 0) setup as well. Also, the NCP and Telnet remote consoles are considered virtual logins, so it will also receive the template port’s setup. It **will**, however, ignore many of the configuration options, such as Dedicated and Access. Otherwise the console could be rendered useless.

# IP/Network Security

*Define* is valid wherever Set is shown in Figure 3-18 and Figure 3-19.

The server provides a security mechanism for restricting incoming and outgoing TCP/IP (Telnet and Rlogin, as well as SLIP and RTEL) sessions. The IP security table allows the manager to enable or disable access to and from the server based on port number as well as IP address. Note that the IP security table does not apply to print connections via RTEL and lpd. Like other configurations, table entries can be set or defined, and can also be saved once the table is complete. To add an entry to the table, specify an IP address (or IP address range), a list of affected ports, and whether to allow incoming, outgoing, both, or neither direction. Figure 3-18 displays an example:

```
Local> SET IPSECURITY 192.0.1.255 OUT DIS PORT 4-7
```

**Figure 3-18: Setting Server Access**

In order, this command:

- ♦ Affects addresses from 192.0.1.1 through 192.0.1.254, using the 255 “wildcard” network address segment.
- ♦ Prevents ports 4,5,6 and 7 from beginning sessions *to* hosts with these addresses.

A few notes should be made about configuring table entries:

- ♦ The IP “address” must be four segments of 0-255 each, i.e., address 131.67 is not valid. A 255 in any segment applies to all numbers in that range - 192.0.1.255 includes 192.0.1.1, 192.0.1.2, etc.
- ♦ A trailing zero in any address segment is shorthand for “all addresses in this range, both incoming and outgoing disabled, for all ports”. For example, the following two commands are equal [Figure 3-19]:

```
Local> SET IPSECURITY 192.0.1.0
Local> SET IPSECURITY 192.0.1.255 OUT DIS INC DIS
```

**Figure 3-19: Set IPsecurity Command**

Parameters can be added after the address term to change the defaults.

- ♦ Port number 0 corresponds to the virtual ports (i.e. users who log into the server from the network). If no ports are specified on the command line, all local ports and the virtual ports are included by default.

- Incoming, Outgoing and Slip Disabled are the defaults for any Set IPsecurity command. Outgoing refers to local users connecting to other TCP/IP hosts; incoming refers to users on other hosts attempting to log into the ETS.

- Individual entries can be cleared by doing a Clear (or Purge) IPsecurity with no parameters other than the address [Figure 3-20]:

```
Local> CLEAR IPSECURITY 192.0.1.102
```

**Figure 3-20: Clear IPsecurity Command**

Note that this form is not valid for any addresses with zeros in any segment.

- The entire security table can be cleared with the following command [Figure 3-21]:

```
Local> CLEAR IPSECURITY ALL
```

**Figure 3-21: Clearing Security Table**

## Using the Security Table

Applying the entries in the table may look confusing at first, but it is rather straightforward. There are two basic rules for checking a TCP/IP connection for legality. First, a more specific rule takes precedence over a less specific one. For example, if connections to 192.0.1.255. are disabled but connections to 192.0.1.78 are enabled, a connection to 192.0.1.78 will succeed. Second, in the absence of any rule that applies to a connection, access is allowed. If this is not desired, include an entry of the following form [Figure 3-22]:

```
Local> SET IPSECURITY 255.255.255.255 INC DIS
         OUTGOING DISABLED
```

**Figure 3-22: Setting IPsecurity**

This is the least specific rule in the table, and will ensure that connections fail unless otherwise allowed by another entry (recall that all ports are included in the rule by default).

If no entries are defined in the table, all connection attempts will succeed. Also, if the user making the connection is the privileged user (see the Set Privileged command) the connection will be allowed regardless of the entries in the table.

It is suggested that managers try enabling and disabling certain addresses on their local network and see what they can and cannot do if they are unclear as to the rules on the use of the security table.

# SNMP Security

The server provides a security mechanism for restricting SNMP access to the unit. Since SNMP can be used to change server settings, a way to prevent unauthorized set commands is needed. The security is linked to the SNMP community name used - by default, only the name **Public** is allowed and it is given only Read privileges. To prevent easy access to the allowed community names, the Show and List SNMP commands are usable by privileged users only.

To change, add, or delete community names in the table, Set/Define SNMP and Clear/Purge SNMP are used. Set SNMP requires that a community name and an access type (Readonly, Both (read and write) or None) be specified. Clear requires either a community name (removes only that entry) or the All parameter (clears the entire table). SNMP queries or set requests that are not permitted are sent an error reply specifying the problem.

# Configuring Event Logging

The ETS can be configured to report various events to either a serial port on the ETS itself or to a network host. The latter can be done with the TCP/IP, LAT, or NetWare protocols. TCP/IP messages will be sent to the syslogd daemon on the host, while LAT and NetWare messages will be echoed to the console of the host/fileserver.

Configuring event logging requires two steps - setting the host to report to, and configuring which of the classes of event will be logged. The format of the host name is used to determine the protocol used to communicate:

- hostname:: Indicates a LAT host
- hostname: Indicates a TCP/IP host (specified as either an IP address or text host name)
- hostname Indicates a NetWare fileserver
- CONSOLE Indicates events should be logged to the ETS console serial port (port 1 by default).

The event classes that can be logged are:

- Commands All commands issued on the ETS will be reported. Note that this is VERY verbose output.
- Network All users logging in/out from the network will be reported.
- Printer Changes in printer status lines will be reported.
- Security All logins and Set Privileged events are logged, as well as any password failures.
- System Booting and Initialize events are logged.

*These can be individually enabled and disabled, so multiple classes of events can be logged.*

For example, to have login and printing events logged to a TCP/IP host [Figure 3-23]:

```
Local> SET LOGGING HOST 192.1.2.0: PRINT ENABLED SECURITY
         ENABLED
```

**Figure 3-23: Set Logging Command**

To clear logging, use the None option [Figure 3-24]:

```
Local> SET LOGGING NONE
```

**Figure 3-24: None Option**

To display current logging status, use Show/Monitor/List Logging.

# Downloading a Configuration File

The ETS can be configured to download a configuration file before it finishes booting. A configuration file consists of commands to be executed by the server as if they had been entered at the Local prompt by a privileged user. This allows one or more ETS servers to be automatically configured without having to manually configure them via NCP, TSM, or a login. If a configuration file is specified, the ETS will download its operational code, and then attempt to download the configuration file. (Note that the Source command is helpful in testing startup files.) If the file is readable, it will treat the file as a list of commands and execute the commands on the ETS in **privileged** mode. There is one important note about executing commands from this file. Due to the fact that the ETS is already running, any Define Server commands in the file will not take effect until the next reboot. Define Port commands will not take effect until the port(s) are next logged out. All other Define commands (Service, Telnet Hosts, SLIP, Menu, Dialback, and IPsecurity) will be permanent, but will also take effect for the current boot.

The configuration file can be downloaded from a TCP/IP host (via TFTP), from a VMS host via LAT, or from a NetWare fileserver. In either case, some host configuration will be necessary. For LAT downloading, you will have to install the ETS$CONFIGD service handler on your VMS hosts. The service handler code is included on the software media included with your ETS. For TFTP loading, you will have to enable TFTP loading on your host and place the configuration file in a download directory. For NetWare, the configuration file must be in the fileserver's login directory. See the Host Configuration sections, below, for details on setting up the download server.

If the configuration file cannot be downloaded at boot time, the server’s behavior depends on the setting of the Retry limit (see examples). If the retry limit is 0, the ETS will wait forever until it can download the file. The server is **not** usable while retrying the download. Otherwise it will try *retry* times and then continue booting. The command to specify the file looks like the following [Figure 3-25, page 3-16]:

```
Local> DEFINE SERVER STARTUPFILE “filename” RETRY 10
```

**Figure 3-25: Defining Startupfile**

There are a few things to note about the configuration:

- Since the Startupfile is a boot-time parameter, if needs to be either defined or saved - it cannot just be set.
- The filename needs to be enclosed in quotes (“s) if it contains any lowercase or non-alphabetic letters.
- The download file is re-read at every boot (unless it is re-configured), so do not put commands such as Initialize or Crash in the file. Unless the startup filename has been changed, an Initialize command will cause the ETS to boot perpetually and will require that NVR be flushed to recover.
- If the startupfile is not readable or there is a problem with the file and the server cannot boot fully, the server is still accessible via the NCP/TSM and Telnet console ports.

The configuration filename is of the form “host:[:]filename”, where host can be a VMS, TCP/IP, or NetWare nodename. If the host name is a TCP/IP host, only 1 colon (:) is used in the name; for LAT hosts, 2 colons are used (::). For NetWare hosts, a backslash is used (\).

For example, to download the file config.sys from VMS host BOZO:

```
Local> DEFINE SERVER STARTUP “BOZO::config.sys”
```

**Figure 3-26: Downloading From VMS Host**

For the same filename from TCP/IP host BASS:

```
Local> DEFINE SERVER STARTUP “BASS:config.sys”
```

**Figure 3-27: Downloading From TCP/IP Host**

For the same filename from NetWare fileserver TROUT:

```
Local> DEFINE SERVER STARTUP “TROUT\SYS:\LOGIN\config.s
```

**Figure 3-28: Downloading From NetWare Fileserver**

A path can also be included in the filename:

```
Local> DEFINE SERVER START "BASS:/boot/sys/config.sys"
Local> DEFINE SERVER STARTUP "BOZO::DUAO:[ETSLOAD]config.sy
```

**Figure 3-29: Including Path in Filename**

Note that a nameserver must be defined if a text TCP/IP hostname is used. The ETS will attempt to resolve the name at boot time, and if it cannot resolve the name, the load will fail. A numeric IP address can also be used for the name:

```
Local> DEFINE SERVER STARTUP "192.0.1.5:/boot/config.sys
```

**Figure 3-30: Using a Numeric IP Address**

The configuration file itself contains ETS commands, 1 per line. The ETS will execute each command in sequence. The commands can be privileged, since the startupfile is run in privileged mode on the server. Comments can be included in the file if they begin with the '#' character. Also, the console port is assumed to be executing the configuration file, so if you have a command like **Set Port Speed 9600**, it will affect the console port's settings. Use Show Server Status to see which port is currently the console - the default is port 1. Figure 3-31 displays an example of a configuration file.

```
DEFINE PORT SPEED 9600
DEFINE PORT PARITY NONE
# this is a comment. Now set up the ports
DEFINE PORT ALL AUTOPROMPT ENABLED
DEFINE PORT 4 ACCESS DYNAMIC
SET SERVICE port4 IDENTIFICATION "Printer on lab ETS port 4" POR
SAVE SERVICE port4
```

**Figure 3-31: Configuration File**

Capitalization is optional, as on the command line. It is used here for command clarity. Managers can use the Source command (see page 10-94) to attempt to download a configuration file before making it a part of the server's boot routine. This is strongly recommended; it helps eliminate errors that might prevent the ETS from booting.

Scripts written for TSM should work unchanged, with the possible exception of privileged mode. Set Priv cannot be used in either a Source or Startupfile file, so the command must be removed from any TSM scripts that are to be used with Startupfile.

# Daily Use and Management

## Setting Up Ports

Initial configuration of ports should be straightforward. The majority of the ports will probably be used for local terminal connections; if so, commands in the following format should be sufficient [Figure 3-32].

```
Local> DEFINE PORT ALL AUTOBAUD ENABLED
```

**Figure 3-32: Configuration Command**

This command line turns on the autobaud feature. Autobaud attempts to configure the port's speed and parity by analyzing the first **<Return>** keys typed at login. Autobaud will not work for devices that do not "log in" (printers and modems, for example) and will not function for baud rates less than 1200 or for certain parity and character size settings (8-bit, no parity and 8-bit, even parity are acceptable). See the Set Port Autobaud command for more detail. If the attached devices will only be at one speed (like 9600) permanently, Autobaud can be left off.

*The EPS1 and EPS2 do not support Autobaud.*

Terminal devices that will be running at 9600 baud will not need the autobaud feature, and can generally be used with the configuration set at the factory. Ports that will be used for non-terminals (printers or modems, for example) will probably have to be configured separately (for baud rate, etc), as will terminals that won't work with autobaud. Other than these cases, terminal-dedicated ports should need no special configuration.

There is one other very important port characteristic that may have to be configured. A port's access mode determines who can access a port - a user on that serial port logging in (access Local), a user connecting to the port from the network (access Remote) or both (access Dynamic). Ports that will offer services that other users can connect to (printers, modems, etc) must allow network connections (i.e. Dynamic or Remote). If they do not, users will not be able to connect to the services. Ports that will never be used for logging into the ETS (printer ports, for example) should have access rEmote, so that stray characters from the printer will not cause the port to be unavailable.

Access must be defined, not just set, since the configuration is reset to the saved values every time a user logs in or connects to a service on that port. If a connection to a service is failing, check the port's access mode first. The default setting is access Local for all ports. See the discussion of service setup below for more information.

Various degrees of "verbosity" can be enabled for ports receiving ETS messages. Messages from other users on the server can be enabled or disabled with the Broadcast flag. If the serial line connecting a terminal to the server is an unsure connection, setting the Loss Notification will cause an audible beep if a character is lost. Finally, Verify will allow or prohibit the printing of informational messages as session connections are started, exited, or switched.

For special port configurations, Autoconnect, Preferred Service and Broadcast are some of the most often specified configuration options. Autoconnect enabled causes the ETS to try to connect to that port's Preferred service whenever a user logs in on that port. This is useful for users who frequently use one host in particular. If it is desired to force a port to connect to one and only one host, a Dedicated service or host can be set up. See the Preferred and Dedicated discussion and warning below.

# Examples of Setting Port Characteristics

### Adding A Feature To A Port

In Figure 3-33, we wish to let port 5 use autobaud at login time so that it can detect and correctly set the port's baud rate.

```
Local> DEFINE PORT 5 AUTOBAUD ENABLED
```

**Figure 3-33: Enabling Autobaud**

### Adding Multiple Features At Once

The Set and Define commands allow you to specify several options on the same command line – up to the command line limit of 132 characters. In Figure 3-34, we want to allow port 2 to receive other users' broadcast messages, enable autoconnect on login, set even parity and change the port speed to 4800 baud on the same command line:

```
Local> DEFINE PORT 2 BROADCAST ENABLED AUTOCONNECT ENABI
        PARITY EVEN SPEED 4800
```

**Figure 3-34: Adding Multiple Features**

### Abbreviating Keywords

In any command, it is only necessary to type as many characters needed to distinguish keywords from one another. For example, this command [Figure 3-35] is essentially the same as Figure 3-34:

```
Local> DEF PORT 2 BRO ENA AUTOC ENA PAR EV SP 4800
```

**Figure 3-35: Abbreviated Command**

Extra white space (more than one consecutive space between keywords) is ignored.

### Changing Character Strings

To set passwords, names, termtypes, and identification strings, you must enter character strings. If a string contains any spaces or has any lowercase or punctuation characters, it *must* be enclosed in quotes. Whitespace indicates the end of any non-quoted string. The characters of non-quoted strings are converted to upper case.

In this example [Figure 3-36], we want to set port 7's name and the username string displayed in response to the Show Users command:

```
Local> SET PORT 7 NAME "servp_7" USERNAME "Freddy"
```

**Figure 3-36: Setting Strings**

To make this change permanent, Define the same characteristics:

```
Local> DEFINE PORT 7 NAME "servp_7" USERNAME "Freddy"
```

**Figure 3-37: Defining Strings**

### Defining A Preferred or Dedicated Service

A Preferred service (or host) is one that a Connect request will default to if no hostname is specified. Also, if Autoconnect is enabled, the Preferred service is the one connected to as soon as a user logs into the ETS on that port.

A dedicated service is one that a port is permanently tied to. The port can only log into that service, and when it logs off that service, it is logged off the ETS. If the initial connect attempt fails, the user is logged off the ETS as well. Users on that port cannot access the ETS' command line at all. For all intents and purposes, the port appears as a physical terminal line directly to the service.

A warning is in order about Dedicated and other "total control" characteristics (such as Menu and SLIP). Once these characteristics are enabled, users are prevented from reaching the ETS' command line and disabling the characteristic. If you configure all the server's ports to be Dedicated (or Menu, etc.), you will not be able to access the ETS' command line unless you log in off the network (via NCP, Telnet, etc). If you cannot log in, you will have to restore factory defaults and reconfigure the server to get to the command line. We do not recommend enabling these characteristics on all the server's ports unless you are sure you can log in from elsewhere on the network.

The first case below [Figure 3-38] will request a LAT connection to service **PHRED** at node **VAX77** when a Connect command is issued without specifying a service:

```
Local> DEFINE PORT 4 PREFERRED PHRED node VAX77
```

**Figure 3-38: Defining Preferred LAT Connection**

If a service name is specified in a Connect request, the preferred service information is not used.

To set up a preferred Telnet connection, you might enter the following command [Figure 3-39]:

```
Local> DEFINE PORT 4 PREFERRED TELNET 146.66.39.252
```

**Figure 3-39: Defining Preferred Telnet Connection**

Set/Define Preferred Telnet commands cannot use the Destination or Node parameters, but they can specify a host either by name or by its IP address. To dedicate a port to a Telnet host:

```
Local> DEFINE PORT 4 DEDICATED TELNET alex.weasel.com
```

### Setting Up Group Numbers For a Port

Each port can be given one or more authorized group numbers. By default, only group 0 is authorized, restricting a port's access to only those nodes that offer services in group 0. The server is also configured for group 0, so by default all ports can access any services local to this ETS. To allow port 4 access to groups 0, 77, and 122 through 178 [Figure 3-40]:

```
Local> DEFINE PORT 4 AUTHORIZED GROUPS 0,77,122-178
```

**Figure 3-40: Defining Port Authorized Group**

## Port Security

Between the various password protections on the server and the access control provided by service groups security should not be a problem - almost any access restriction can be enforced. For ports that need a higher level of security, the Set Port Security command allows the server manager to restrict a port user's actions on the server. Ports set this way are called secure ports. If a port is set up to be secure, it cannot see or affect other ports on the system and cannot use the Set Privileged command. For terminal connections that need to be restricted but still have access to the command line, the secure option is best.

Ports that do not need to access the command line can be put in menu mode, where they are allowed to execute only certain commands configured by the manager. Using the Set/Define Menu command, the manager can add certain commands to a menu used by ports in menu mode. These ports can only enter commands from the menu, and are not allowed access to the command line. For users who do not need or want to learn the ETS command set, Menu mode should be configured. See the discussion of menu mode later in this chapter.

If incoming modem security is a concern, ports can be restricted to dialback status. For information on Dialback, refer to page 3-34.

Due to the sequence of security checks (see list below) multiple security features can be enabled on any port. For example, a port can have be configured to require a login password, enter a dialback username, and then be put in menu mode.

| Server Checks: | To Determine: |
|---|---|
| Access is set to None? | User cannot obtain any prompt or log in. |
| Port has password enabled? | User must enter login password. |
| Port has dedicated enabled? | Port is forcibly autoconnected; user cannot access ETS command line. |
| Port has dialback enabled? | Username must be an authorized name; reconnect attempt must succeed. |
| Port has security enabled? | Command set is restricted. |
| Port has menu enabled? | Port is forced into menu command mode. |

Two other port options provide more limited security for logins. Setting the Inactivity Logout attribute causes the port to be logged out after a set period of inactivity (i.e., no keyboard or network activity). The length of time can be set with the Set Server Inactive Limit command.

Normally, a port is *not* automatically logged out when the device it is attached to is turned off or disconnected. Enabling DSRlogout will log out the port's user if *either* of these cases occurs. This will prevent unauthorized users from unplugging terminal cables and gaining access through other users' still-active sessions. DSRlogout is also useful with incoming modems. When the modem hangs up, it will logout the user and thus, the host will force subsequent users to log in. For this reason, Modem Control implicitly enables DSRlogout (although it will not show up as enabled), as well as other features.

DSRlogout, Modem Control, and hardware flow control (i.e. DSR/DTR or RTS/CTS) are mutually exclusive features on the ETS and EPS4/12 units, as the same physical serial line signals are used for each of the features.

## Creating Services

In the case of services, Set and Define refer to whether the service will be created or changed immediately or the next time the server reboots. Likewise, Clear and Purge are used, respectively, to remove a service immediately (but not permanently) or permanently (but not until the server is rebooted).

When you try to modify a service that does not exist yet, the server assumes you want to create that service. If the service already exists, the command will simply change its characteristics. The ETS will only print an error message if you try to remove a service that does not exist. Note also that connections and queueing are enabled by default whenever you create a service.

*Autobaud should not be enabled for any port that offers a service. Also, recall that a port's access must be set to Dynamic or Remote to allow service connections.*

If you are new to setting up services, you might try using the simple service below, and just use a terminal as the print device. You can then connect locally to the service and see if typed characters are displayed on the terminal device. Characters typed on the service terminal should also be sent to the original port. If it seems to be working properly, you can then try using a real print device and configure the port accordingly. Recall that you will need to use the <Break> key or your port's local key (see the Set Port Local Switch command) to return to the command line from a service connection.

The EPS1 and EPS2 are different from the ETS/EPS products in that they come with services for the serial and parallel ports already configured. These services (EPS_xxxxxx_S1, EPS_xxxxxx_P1, and EPS_xxxxxx_P2 on the EPS2) can be used as is, or deleted in favor of new ones. The services both have NetWare **enabled** and LAT **disabled**, unlike services created via the command line. The serial port service also has AppleTalk enabled by default, again unlike services set by the user. These settings may need to be changed. Note also that the EPS1 has no method of connecting to a local service - the service must be tested from another node, whether LAT, TCP, NetWare or AppleTalk.

### Creating A Simple Service (Such As A Line Printer)

In Figure 3-41, we will create a service called *dump_port* on port 4. Connections and queueing are enabled by default when you do a Set Service, so it is not necessary to do so explicitly. LAT and RTEL access are also provided by default; NetWare and AppleTalk access have to be explicitly enabled. The port providing the service must also allow remote access, so it must have Access set to Dynamic or Remote. Recall that Dynamic means both Local and Remote. After setting the service up, you should be able to connect to it locally and see it from other LAT-compatible nodes (with the Show Services command, for example).

```
Local> SET SERVICE dump_port PORT 4
Local> LOGOUT PORT 4
```

**Figure 3-41: Set Service Command**

Note that:

- As we are using Set Service, the service changes will take effect immediately but will not be permanent. Either Save Service or Define Service will make a service permanent.
- We did a Define to configure the port. Since the port is logged out after each print job or connection, a Set will be lost immediately. Define ensures that the setting will be enabled every time the port is used. In this example, we log port 4 out to force the Defined setting to take effect.

To verify the service was created and usable [Figure 3-42]:

```
Local> SHOW SERVICE LOCAL CHARACTERISTICS

Service: DUMP_PORT                Ident:
Rating: 255                       Ports: 4
Characteristics:                  Queuing Connections
Rtel Lat
Enabled Groups: 0
```

**Figure 3-42: Show Service Command**

Note that the service rating is non-zero, indicating that the service can be connected to. If the service rating is zero, it indicates that the service is not available. This can happen if the port is not configured to Dynamic or Remote access, connections are disabled, or if the service/port is in use by someone else. The Show Port All command is useful for finding out why a service is unavailable, as it shows the ports' access modes, their login status, and the services they are currently offering.

To connect to service dump_port (either locally or from another LAT node) [Figure 3-43]:

```
Local> connect dump_port
%Info: connecting to service dump_port at node...
```

**Figure 3-43: Connecting to Service**

Any output will now be displayed on port 4. If there is a terminal device on port 4, it will echo keystrokes to your port. To disconnect from the service, type the Break key (to return to the Local> prompt) and type **Disconnect**. Your session will be closed and the dump_port service will be available for connections. Enter the Show Service command from above to see this.

If this was a physical print device, the port 4's baud rate should be set to reflect the speed at which the printer is capable of printing. Parity, character size, and flow control might have to be set in this case, as well.

### Setting Up A Service With Group Codes

Local ports can be given a list of LAT service groups that they are authorized to use. This has the effect of denying the ports access to services that do not have any matching authorized groups. For example, you may wish to limit connections to a printer to a certain set of ports on a server. To start up a printer service and ports with group list access, we could enter the following commands [Figure 3-44]:

```
Local> SET SERVER GROUPS 1,2-5 ENABLE
Local> DEFINE PORT 8 SPEED ALL AUTHORIZED 1,4,7 ENABLE
Local> DEFINE PORT 8 SPEED 2400 BROADCAST DISABLED
Local> DEFINE PORT 8 ACCESS REMOTE NAME “printer” BREAK DISABLED
Local> LOGOUT PORT 8
Local> SET SERVICE PRINTER PORT 8 IDENTIFICATION “LocalPrint
Local> SAVE SERVICE PRINTER
```

**Figure 3-44: Service Configuration Example**

In order, these commands will:

- Enable groups 1 and 2 through 5 on the server and assign them to any local services.
- Define all the local ports so they can access groups 1, 4 and 7. Note that only authorized groups are defined. When a port logs in, all of its authorized groups are usable. Since groups 1 and 4 are common to both the ports and local services, all the local ports will be able to connect to any local service.

- ♦ Set up port 8 (where the printer will be connected) - reset the speed to 2400 baud, and disable broadcasts to the port. Depending on the printer, you might also want to disable everything that might send output to that port, such as Loss Notification, etc.
- ♦ Further configure port 8 - define remote access (since the printer will not need to log in), set the port's name, and disable any Break key processing.
- ♦ Logout the port, so that the Defines done above will take effect.
- ♦ Create a service "printer" on port 8; (allowing connections and queueing by default), and have the server send out an identification string in network multicasts.
- ♦ Save the service, so that it will be enabled after the next server reboot.

Group codes have no effect on the other protocols supported by the ETS (TCP/IP, NetWare, and AppleTalk). They are only applicable to LAT connections.

### Enabling other Service Options

There are 5 other options (all printer-related) that can be enabled on a service [Table 3-1, page 3-48]:

| Option | Functionality |
|---|---|
| AppleTalk | If enabled, allows AppleTalk users to see this service. Macintosh computers, for example, will see the service name in their Choosers if they are on the same zone/ network as the ETS. Note that all AppleTalk-enabled services on the ETS are assumed to be LaserWriter printers running PostScript. There is no way to support non-PostScript devices (like ImageWriters, modems, etc). |
| Banner | If enabled, the ETS will provide a banner page before any service data is sent to the port. Note that the host software may still provide its own banner page, regardless of the ETS service setting. Default is disabled. Should not be used with Postscript. |
| Binary | If enabled, the ETS will do no processing on the data stream. The default (disabled) allows the ETS to convert <LF> to <CR><LF> and possibly do tab expansion. Postscript (below) implies binary mode. |
| Formfeed | If enabled, the ETS will append a Formfeed to the end of print jobs. |
| Netware | If enabled, allows print queues on NetWare fileservers to use this service. Note that if NetWare is not enabled on any services on the ETS, the ETS will not poll the fileservers at all - it will only poll if there is a possible queue in use. |
| Postscript | If enabled, the ETS will query the printer (via Ctrl-D exchanges) to make sure it is ready for a new job before starting any job. Any service supporting only a PostScript printer should also have this flag enabled - see PostScript Notes, above. |

**Table 3-1: Service Options**

### TCP/Telnet Service Sockets

The Tcpport and Telnetport service options allows you to configure a TCP socket associated with a service. Connections to that socket number would then be mapped to the service; this allows you to create a TCP "rotary" that can apply to multiple physical ports. As long as the service rating is non-zero (i.e. a port is available to handle the TCP connection), TCP connections to the socket will be accepted. The Telnetport and Tcpport options are identical, except that Telnetport will do Telnet IAC interpretation on the data stream, while Tcpport treats the connection as a raw data pipe. Neither provides for queueing - if the service is unavailable, the TCP connection will be rejected.

### Setting Up A Modem Service

The example displayed in Figure 3-45 is detailed, but once you have created a service or two, the process should become largely intuitive. Remember that the Show Service and List Service commands are available to show you exactly what has been set up so far. If you are unsure or think you may have made a mistake, delete the service (Clear/Purge Service) and start from scratch. Refer to the section below for modem-specific configuration issues.

```
Local> DEFINE PORT 2 PARITY NONE FLOW XON SPEED 2400 BROADCAST DISA
Local> DEFINE PORT 2 ACCESS DYNAMIC NAME "modem" BREAK DISABLED
Local> DEFINE PORT 2 MODEM ENABLED PASSWORD ENABLED
Local> LOGOUT PORT 2
Local> SET SERVICE MODEM PORT 2 IDENTIFICATION "2400 BPS Modem"
Local> SET SERVICE MODEM PASSWORD "loco"
Local> SAVE SERVICE MODEM
```

**Figure 3-45: Modem Service Setup**

Because many of the commands shown are similar to the ones in the previous example, we will only comment on those that differ:

♦ The port is configured much the same. XON/XOFF flow control has been enabled for the modem's use. Access to the port is now dynamic, so that both remote and local users can access the modem to dial in or out. Finally, modem control and password protection are enabled.

♦ Users who dial in to the modem and connect to the server's serial port will have to enter the login password, due to the Define Port Password Enable. This option can be deleted if not desired.

♦ The service is set up as above, but a password has been enabled for the service as well as for the port. Users who Connect to the service (locally or from another host on the network) will have to enter the service password ("loco" in this case) to use the service to connect to the modem and dial out. Again, the option can be left out if no service password is desired.

Since modems do not log you out when you are finished with a connection, you will probably want to have the <Break> or Local key available (see Set Port Local Switch) so you can return to the ETS' command line easily. If you wish to disconnect from the modem session, use the Disconnect command or Logout.If a port and modem are to be used to allow dial in connections, you may wish to make the port Secure, which prevents dial-in users from changing the port characteristics. Otherwise users can change port settings (parity, baud rate, etc.) and prevent other users from being able to dial in properly.

# Modems

Although modems are sometimes difficult to set up, the following points should help you connect one to your ETS. Refer to the cable pinouts in the Installation Guide for more details as you read these instructions. Keep in mind two things:

- The special handling for modem connections is only done if the port in question has Modem Control enabled. See the Set/Define Port Modem command for syntax and details.
- Most modem problems are caused by cabling mistakes, or incorrect modem configuration. The ETS' handling of modem ports is rather simple and straightforward.

Modems are usually connected to data equipment (terminals, servers, etc.) with a 25 pin (DB-25)) connector that allows the equipment to see all the signals the modem provides. These signals include Ring (RI), Carrier Detect (CD), and Data Set Ready (DSR). The ETS does not have access to all of the modem control signals due to the use of the 6-pin MMJ/RJ12 jacks. Therefore, you may need to make some changes to the modem's setup to allow incoming and outgoing calls to be handled correctly. The EPS1 and EPS2 units can handle both the DSR/DTR and RTS/CTS pairs due to the RJ-45 connector.

# Modem Basics

When modem control is enabled on a port, the ETS uses the DSR input signal on the serial port to decide when there is a valid modem connection. It uses the Data Transmit Ready (DTR) output line to signal the modem that a connection is possible or acceptable. The attached modem may not be using these signals as the ETS expects, however. For a sample incoming call, the signal sequence might be:

1. Modem and ETS wait for a connection. The ETS provides DTR to the modem so that the modem will accept any incoming calls. Modem does not provide DSR to the server, since it is not connected via the phone line.
2. The modem receives a call, raises the Ring signal, and then raises Carrier Detect (CD) to signal that it accepted the connection.
3. The ETS sees its DSR input signal go high (when the modem asserts DSR and/or CD).
4. The ETS then logs the port in, the remote caller presses Return and gets a login prompt or autoconnect from the ETS.
5. The user works for a while, then decides to logout and exit.
6. If the user logs off the ETS locally, the ETS drops the DTR line long enough (roughly 3 seconds) to signal the modem to hang up the remote connection.
7. If, instead, the remote modem hangs up before the user logs out, CD (the server's DSR) drops, and the ETS closes the local session.

For outgoing calls, the session might be:

1. A local user connects to the modem service. The ETS keeps DTR high for the modem, and pays no attention to (incoming) DSR yet.
2. The user dials out and connects to a remote service.
3. The modem now raises CD (and therefore ETS' DSR signal) as a sign that it is connected. The server now begins monitoring the DSR line while the session progresses.
4. If the user logs out of the remote connection, the remote modem hangs up, and the local modem drops DSR. The ETS then logs the user off the service and makes the service available again.

   If the local connection is closed (by a Disconnect, for example) the ETS dr ops DTR to signal the modem to hang up the remote end and closes the session.

## Modem Wiring

There are 2 fundamental wiring issues to address: DTR, and DSR. Both issues are discussed in the following sections.

### DTR - Data Terminal Ready

The ETS normally asserts DTR. When Modem Control is enabled, the server will drop DTR for three seconds each time the port is logged out and each time a network user disconnects from a modem service. The modem must be configured to hang up and recycle when DTR is dropped. Security problems may result if the modem is not configured in this way.

### DSR - Data Set Ready vs. CD - Carrier Detect

The ETS must know when a connection is lost so it can log out the port and close any sessions. Otherwise security problems will result when the next user logs into the port. The server also needs to know when a valid connection does not exist so it can ignore spurious characters from the modem. Otherwise the server may interpret the characters from the modem as a user logging in, causing the port to be unavailable.

*On most modems DSR does not act like CD by default; DSR is always asserted. If DSR is to be wired to the server, the modem's configuration must be changed.*

Most modems assert CD only during a valid connection by default. In this case, the modem's CD pin may be wired to the server's DSR pin. Alternatively, many modems can be configured such that DSR acts just like CD. If this is the case, the modem's DSR pin may be wired to the server's DSR pin. Wiring the modem in this way may allow the use of preassembled connectors.

## Baud Rate Schemes

The ETS and modem must agree on the baud rate used for the serial interface. In addition, the modem may be communicating with the remote modem at a different speed, due to the use of compression and error correction techniques. The ETS and modem can be configured in one of three basic ways:

- Set the baud rate of the server and the modem to the highest expected rate, disable autobaud on the server and on the modem. Use flow control to handle connections at all slower rates. On some modems disabling autobaud is known as “locking the serial speed” or “locking the DTE speed”. This method allows nonstandard transmission rates, such as those found in V.32bis (14,440 baud), error correction and data compression. This method is generally the most reliable as the modem and the server always utilize the same baud rate. Ports that support V.32bis should be set to 19200 baud. Ports that use compression should be set one or two speeds higher than the modem's speed, i.e. 2400 baud modem with compression should be set to 4800 or 9600 baud.
- Use the autobaud feature of the server. The server will adjust its speed to that of the modem. Flow control isn't strictly required in this configuration. Applications that cannot use flow control due to limitations below may still function correctly, but if at all possible, flow control is still recommended. Nonstandard transmission rates, such as V.32bis (14,440 baud), error correction and data compression cannot be supported with this method. During a long data transfer, eventually either the ETS or the modem will run out of buffer space and overflow the other, resulting in data loss.
- If only one standard baud rate is to be supported, set the server to that rate and disable autobaud. Flow control may be disabled, subject to the same caveats discussed above.

## Flow Control

On the EPS1 and EPS2, RTS/CTS flow control is recommended. Disable XON/XOFF flow control and enable RTS/CTS flow control on the modem and the server.

The remaining servers do not have enough serial signals to support RTS/CTS flow control and modem control simultaneously. XON/XOFF flow control is therefore recommended. Enable XON/XOFF flow control on the modem and on the server. Note that some applications, such as some file transfer protocols, SLIP, and UUCP, cannot use XON/XOFF flow control. In order to support such applications, a baud rate scheme must selected that does not require flow control. Disabling flow control may preclude the use of some modem features, such as error correction, data compression and nonstandard transmission rates, such as V.32bis.

## Modem Configuration

In dialout situations, it is possible for users to reconfigure the modem with normal 'at' commands. It is possible for a user to misconfigure a modem such that it will be unavailable, unusable or cause security problems. Fortunately, many modems can reload their saved configuration each time DTR is dropped by the server. Provided the user doesn't modify the saved settings, the modem will always be configured correctly after each use.

## Modem Result Codes

If the modem is configured to return 'result codes' or status messages, the server and the modem may start to “babble”. A message from the modem will be interpreted as a user logging into the server, and the server will send a prompt to the modem. The modem will interpret this prompt as a command, and send an error message to the server. This process causes the port and modem to become unavailable, and can be cleared temporarily by logging the port out. The best solution to this potential problem is to configure the modem so that it never returns result codes. Unfortunately, the lack of status messages may make outgoing connections inconvenient. If the modem supports displaying messages only during outbound connections, babbling can usually be avoided.

## Modem Configuration Checklist

These items should be verified after any modem configuration. Note that some options might not be provided on the modem in question.

- ♦ Modem must disconnect immediately when DTR is dropped.
- ♦ Modem must assert CD (or DSR, if connected) when connected to another modem. It must not assert CD when disconnected. The modem may optionally assert CD during outbound dialing.
- ♦ Modem and server must agree on the flow control method and baud rate scheme.
- ♦ Modem must not send result codes or messages to the server except optionally during outgoing calls.
- ♦ Configuration information should be saved in non-volatile memory so that it will be retained during power outages.
- ♦ Modem should restore its configuration from non-volatile memory when DTR is dropped.
- ♦ Modem should be configured to answer the phone if incoming connections are to be supported. Generally this is done with the ats0=1 command.
- ♦ Modem should not answer the phone if DTR is not asserted by the server.

## Server Configuration Checklist

These items should be verified after any ETS port configuration.

- Modem Control must be enabled. Using modems on ports without Modem Control enabled will lead to security problems. See the Set/Define Port Modem command for syntax and details
- Autobaud **should** only be enabled if required.
- Baud, parity, and flow control settings should be verified with the modem setup.

## Testing

This section assumes that the modem is attached to port 4 and the modem and server checklists have been followed.

- Dial into the server. Issue the Test Port command. A regular cascading pattern should be printed. If characters are lost after a few dozen lines, there is a flow control problem. On another port, issue the Logout Port 4 command. The modem should immediately hang up and be ready for the next call in a few seconds. If not, the modem is ignoring DTR, or there is a wiring problem. Repeat this procedure with all expected modem speeds and options.
- Dial into the server. Connect to a network host and log in. Hang up the remote modem without logging out of the server. After the server's modem hangs up, issue the Show Port 4 command from another port. The port should be in the Idle state. Dial into the server. The port should have been reset, and should no longer be connected to the network host. If not, the modem is incorrectly asserting DSR/CD, or there is a wiring problem.
- If outbound calls are to be supported, connect to the modem's service with the Connect command. (In many cases the Connect @@PORT_x command is a quick shortcut). Dial a remote modem. In some configurations it may not be possible to see the dial commands or the modem's status messages. After the modem connects, log into the remote system.

If a modem fails to pick up the phone, DTR may not be wired correctly, or the modem may be configured to ignore incoming calls.

If modems fail to connect to each other, there could be a modem compatibility problem. For example one modem's configuration might require a special feature, such as error correction to complete connections. Verify both modem's configurations, or try swapping one or both modems with different models or different brands.

## Dtrwait

Setting the DTRwait flag on a port changes the providing of DTR when the port is idle. Normally DTR will be asserted when the port is not in use. In general, this allows modems to answer an incoming call, as many modems will not answer a call if DTR is not asserted. If you desire DTR held low, however, setting the DTRwait flag will cause the ETS to **not** assert DTR until the port is actually in use. In use refers to either a login from the attached device (terminal, etc.) or a connection from a "network" user (a LAT service connect, a Telnet/RTEL job, etc). When such a connection begins, the ETS will raise DTR; when the connection is closed, DTR will be deasserted.

## Dialup

It is frequently useful or essential for a LAT host to know whether a connection attempt is from a modem or not. The LAT host will inquire about modem status when any connect is made; you can inform the host about modem status with the Dialup flag. If Dialup is enabled on a port, that port will report that it is used for a dial-in modem if the LAT host asks. If you have any ports that will be used exclusively for modem access, we suggest you define this parameter. The Dialup characteristic has no effect on non-LAT (Telnet, Rlogin, etc) connections.

Some services or nodes may reject a connection because of its Dialup status due to the inherent security concerns with modem connections. While this may seem like an annoyance, it is a security feature, and should be implemented (via the Dialup flag) if possible.

## Dialback

The ETS also supports the use of Dialback operation. This allows a user to log in from a modem, type a pre-defined username, and have the ETS log the port out and call the user back with a preset modem command. The dialback system can be used to enhance security, or provide central telephone billing. The ETS forces ports with Dialback set to use only pre-configured usernames, and expects each username to be reachable with a certain modem command string. If the username is not in the dialback table or the reconnect attempt does not succeed, the port is logged out.

Three steps are required to configure an ETS port for Dialback status. First, the modem and port must be configured as discussed above. Second, the port has to be put into Dialback mode (see the Define Port Dialback command in the command summary for details). Note that once a port is configured for dialback, it cannot be used without giving one of the usernames in the dialback list. If the username entered at the prompt is not in the dialback list, the port will be logged out immediately.

Third, the dialback list must be created. Each dialback entry consists of a username and a modem command string that will be output to the port when that username is logged in. For example [Figure 3-46]:

```
Local> SET DIALBACK "Ronald" "atdt555-1234"
```

**Figure 3-46: Set Dialback Command**

When username "Ronald" is entered on any dialback port, the ETS will hang up the modem by cycling DTR, then output the command "atdt555-1234" to the corresponding serial port. The dial string should perform any special configuration required for the call, then dial the remote modem number (555-1234 in this case). The ETS has no knowledge of what the dialback command string should do - it simply outputs the command to the serial port and then decides whether the reconnect attempt succeeded. The server waits for up to 80 seconds for the DSR signal to go high, indicating that the modem reconnected successfully. Otherwise DTR is dropped for 3 seconds and the port is reset.

There are 4 things to note about dialback:

- The Define Port Dialback command also enables Modem Control by default. It can be disabled if it is not applicable.
- The dialback functionality only applies to logins into a port. Dialback ports can be used normally for outgoing service connections from local and network users as well.
- The reconnect command string is limited to 32 characters, and is automatically followed by a <Return> when the server echoes it to the port.
- The ETS allows 30 seconds for the user to enter a username when in Dialback mode. After 30 seconds, the port is logged out. This is to keep unauthorized users from denying access to the port to others.

## Dialback Security

The Dialback system does not absolutely guarantee security. Depending on the modem in use and its configuration, it may possible for a determined attacker to penetrate the system. There are two windows of vulnerability where an attacker could gain unauthorized access to the server. The first exists after the server hangs up the modem but before the modem dials the user back. The second window exists when a dialback attempt fails but before the server reaches the end of its 80 second timeout period. Careful configuration and testing of the system during these short vulnerable periods is required insure to a high level of security.

If a second call arrives in the few moments after the server hangs up the modem, but before the server issues the dial command, security may be breached. Until the modem goes 'off-hook', the modem may answer an incoming call and remain online, granting access. This is highly unlikely and the chances of unauthorized access can be reduced further by configuring the modem to answer only after the second or third ring. Also, the modem must not answer the phone when DTR is not asserted. If possible, the modem should be configured to only dial after detecting a dial tone, and hang up otherwise.

If the dialback call should fail for any reason before the 80 second timeout (if the number dialed is busy, for example) a second caller may be able to log into the server. This possibility can be virtually eliminated by these two steps:

1. In the dialback string itself, issue a command to ignore incoming calls An example of this is “ats0=0dt555-1234”.
2. Re-enable incoming call answering after DTR is dropped (i.e. after the dialback session has ended). To accomplish this, the modem should be configured to reload its normal configuration each time DTR is dropped.

## Modems with SLIP

If Modem Control is defined on a port that also has SLIP autostarted (see next section), SLIP is handled slightly differently. In this case, SLIP will not be started until the port sees its DSR signal high. This generally occurs when the attached local modem has formed a connection with a remote modem. Until DSR is asserted, the port will remain idle, in non-SLIP mode. It is usable as a service port in this state (until SLIP is started). After SLIP is started, the SLIP session will be closed if the attached modem device drops the DSR signal. For more information on SLIP configuration, see “Configuring SLIP Access,” next.

## Configuring SLIP Access

See the "SLIP" section of the Concepts chapter, above. SLIP allows hosts to establish TCP/IP connections over serial lines to the ETS, and then onto the network attached to the ETS. The ETS will route packets among the serial and ethernet lines transparently to the user. This allows hosts, for example, to dial in via modems and run TCP/IP sessions directly to the net, without having to go through the ETS' command line. SLIP session can be configured to Autostart, or can be manually started with the Set SLIP command.

Configuring a serial line for SLIP access involves 2 steps: the manager enabling SLIP access on the port (see the Set/Define Port SLIP command) and a user actually starting a SLIP session after logging into the serial port (see Set/Define SLIP). There are a few things to note about SLIP access:

- ♦ After the SLIP session is started, the local mode of the ETS is not reachable again without logging the port out (thus terminating the SLIP session) and then re-logging into the ETS. The port can be logged out either via a command (Logout Port) or via the serial line (DSR drops, etc). Also, a SLIP session's characteristics (IP address, MTU, etc) cannot be modified once SLIP is started.

  Until the port is actively running SLIP (i.e. SLIP was not Autostart Enabled), it can be used normally as a service port, a login port, etc. If a modem is attached to the port, users can connect to the port and dial out. The port changes to SLIP interpretation only after receiving the Set SLIP command.

- ♦ Modem control has to be enabled (see Define Port Modem Control) for SLIP connections running across a modem. Otherwise the SLIP session will not be terminated even though the user might have hung up the phone line. With modem control enabled, the SLIP session will be terminated when the locally-attached modem hangs up (either because the remote modem hung up or because the ETS port was logged out).

- SLIP data is "binary" in the sense that any 8-bit character can occur in the SLIP data stream. Because of this, any attached modems CANNOT be set to use XON/XOFF (or any other character based) flow control. Otherwise the session will hang and/or lose data. All devices (modems, etc) between the SLIP host and the ETS also must pass all characters unchanged, including XON/XOFF. Many devices can be configured to have flow control off, but they will still strip XON and/or XOFFs out of the data stream. This is not acceptable. Furthermore, the ETS will force the serial port's flow control to None for the duration of the SLIP session, and will restore it when the SLIP session ends. Hardware flow control is unaffected by SLIP operation. Note also that 7-bit characters cannot be used - the SLIP session will force 8-bit mode.

- SLIP connections can be restricted via the Set Server SLIP Password and Set IPsecurity commands. Maximum packet size (MTU) and whether or not to pass IP broadcast packets to the SLIP lines can be configured via the Set/Define SLIP commands.5.The MTU value and IP fragmentation can have a large effect on both the interoperability and performance of SLIP connections. The MTU is the size of the largest packet that can be sent unfragmented on a particular media. Packets larger than that size must be fragmented into 2 or more packets, and reassembled by the remote host. For example, the Ethernet MTU is 1536 bytes. The SLIP guidelines (RFC 1055) suggest an MTU of 1006 bytes for SLIP connections. Some SLIP implementations, however, have a lower MTU. In those cases the SLIP MTU on the ETS must be set lower, which increases IP fragmentation. This in turn decreases performance and can increase network traffic, as more packet headers are needed for the fragmented packets.

  In general, the ETS' MTU for SLIP should be as high as possible without exceeding the remote hosts' MTU values. If the remote SLIP device does not support IP packet reassembly, the MTU should be set as high as possible to prevent any fragmentation.

- No extra routing configuration should be needed by the remote SLIP hosts. The only change they may need is to make the default route point to the ETS; you may also wish to configure gateway hosts on the ETS if non-local TCP/IP sessions will be started. Providing a default route on the remote SLIP host might look like the following [Figure 3-47]:

```
% su
# route add default ETSname metric
```

**Figure 3-47: Specifying Default Route**

Before SLIP can be started on any port, the port must have SLIP enabled and SLIP parameters must be set. A SLIP session cannot be started without the ETS knowing the IP address of the remote SLIP host, the MTU to use for the SLIP connection, and whether to forward Broadcast IP packets onto the SLIP line. Each of the 3 parameters can be set one of 3 places - as a default for the entire server, as a default for a particular port, or when the SLIP session is actually started. For example, the command below specifies a default MTU and Broadcast for all ports [Figure 3-48].

*See your local host's documentation (**man** pages for route, SLIP, etc.) for more information on MTUs and routing configuration.*

```
Local> DEFINE SLIP DEFAULT MTU 512 BROADCAST DISABLED
```

**Figure 3-48: Specifying Defaults**

A configuration that applies to one port in particular might look like the following [Figure 3-49]:

```
Local> DEFINE SLIP PORT 5 MTU 640 BROADCAST ENABLED
```

**Figure 3-49: Configuration of One Port**

Defaults for a specific port take precedence over server-wide defaults. Any options not specified on the Set SLIP command line will take their values from the server or port default values.

For a particular port, an address can also be specified (if only one IP address will ever connect to that serial port) [Figure 3-50]:

```
Local> DEFINE SLIP PORT 5 IPADDRESS 192.0.1.100
```

**Figure 3-50: Specifying an IP Address**

In addition, default entries can be forced on new sessions, so that users cannot override one or more of the defaults. If the Force option is enabled in the Define SLIP Default or Define SLIP Port commands, those options will take precedence over the user's options on the Set SLIP command line. If Force is not enabled, the user's options will take precedence. For example, the following series of commands [Figure 3-51] will use an MTU of 1000, since the Force option takes precedence over the user's request.

```
Local> DEFINE SLIP DEFAULT MTU 1000 FORCE ENABLE
Local> SET SLIP ADDRESS 192.0.1.20 MTU 500
```

**Figure 3-51: SLIP Configuration Example**

Conversely, the command sequence below [Figure 3-52] will use the user's MTU value of 500.

```
Local> DEFINE SLIP DEFAULT MTU 1000 FORCE DISABLED
Local> SET SLIP ADDRESS 192.0.1.20 MTU 500
```

**Figure 3-52: SLIP Configuration Example**

The Force option applies to the MTU, Broadcast, and IPaddress options. If Force is enabled and IPaddress is not defined, the user will still be able to specify the remote IP address, but will be forced to use the other default settings. If the IP address is also specified and Force is enabled, the user's choice of IP address will be ignored in favor of the configured force value.

### Starting a SLIP Session

Three steps are needed to start a SLIP session on a serial port. First, SLIP must be enabled on any ports that might want to start a SLIP session in the future. For example [Figure 3-53]:

```
Local> DEFINE PORT 5 SLIP ENABLED
```

**Figure 3-53: Enabling SLIP**

This also allows the manager to prevent SLIP access on any particular port. As a note, doing any Define SLIP Port x does a Define Port x SLIP Enabled automatically.

Second, values for the IPaddress, MTU, and Broadcast options must be set. They can be specified on the command line or configured as server-wide or per-port defaults. As discussed above, defaults can be configured to be Forced on any SLIP session started.

Finally, any port that wants to run SLIP simply needs to start a SLIP session [Figure 3-54]:

```
Local> SET SLIP
```

**Figure 3-54: Starting SLIP Session**

Or, if an address has not been configured for the port [Figure 3-55]:

```
Local> SET SLIP IPADDRESS 192.0.1.50
```

**Figure 3-55: Setting SLIP IPaddress**

After entering the Set SLIP command, the user will be logged out and the ETS will begin treating input from the serial line as SLIP packets. Note that the Local> prompt cannot be reached again unless the SLIP session is terminated (via the port logging out) and the user re-logs into the ETS.

If a particular port should **always** be running SLIP, use the Autostart option [Figure 3-56]:

```
Local> DEFINE SLIP PORT 4 MTU 1000 AUTOSTART ENABLED
```

**Figure 3-56: Enabling Autostart**

Port 4 will then always treat serial data as SLIP packets until the Autostart option is removed and the port logged out. The exception to this case is if the port also has Modem Control enabled. If so, a drop in the DSR signal will terminate the SLIP session, and the session will not restart until DSR is re-asserted. See the "Modems" section above for more details.

A password can be configured to restrict access to SLIP sessions [Figure 3-57]:

```
Local> SET SERVER SLIP PASSWORD slpass
```

**Figure 3-57: Setting SLIP Password**

If a SLIP password has been set on the ETS, the server will prompt for it after the Set SLIP command has been entered. If the password is not entered correctly, SLIP will not be started on that port and the user continues at the Local> prompt. Note that ports with SLIP Autostart enabled will not have to enter the password - they will always be in SLIP mode. If you are concerned about SLIP security, do not use Autostart and require that all SLIP users give the SLIP password before starting.

If SLIP is being run on a serial port with a modem attached, be sure that Modem Control is enabled on that port; otherwise, the ETS will not log the port out when the modem hangs up and the port will remain in SLIP mode after the session is closed. See the "Modems" section of this chapter for details and warnings concerning modem usage. Other than a modem closing the serial connection via a DSR transition, SLIP can only be disabled on a port by explicitly logging the port out (Logout Port x).

### Other SLIP Considerations

Tips for increasing SLIP performance:

- ♦ Disable broadcasts. The extra packet traffic can be surprisingly heavy.
- ♦ Set the MTU as high as possible on the ETS and on the remote SLIP device. This lowers the ratio of header information to useful data. This will also reduce time-consuming IP fragmentation.
- ♦ Be sure that characters aren't being lost or altered by any communications equipment between the SLIP hosts. This will corrupt packets and cause retransmissions. Flow control between modems is particularly troublesome.
- ♦ When telnetting with SLIP, telnet directly to the desired host, and not to the ETS and then to the final destination.

SLIP and network security:

- ♦ If possible, restrict the IP addresses that the user can select, either via the Force option or the IP security tables. This will help prevent the remote SLIP host from "impersonating" a legitimate host on your network.
- ♦ Be very careful with .rhosts and /etc/hosts.equiv files on Unix machines.
- ♦ Disable broadcasts, because it is possible that the remote user can learn important information from packets being broadcast from your network onto the SLIP line.

Counters are available for each SLIP session via the Show SLIP Port command. The displays will show both forwarded and discarded packets for both directions of the session, and may be useful for debugging SLIP connections. The display also shows counters for fragmentation, reassembly, and packets dropped due to congestion on the SLIP line. The IP errors field shows incoming packets that were discarded due to protocol problems (checksums, header problems, incorrect source host, etc.) Recall that the ETS will discard any packets from the SLIP session that are not from the host initially configured (via the Set SLIP command, etc.)

*See your host documentation for information on enabling UDP checksumming.*

As a final note, hosts that do not check UDP checksums may send and receive corrupted packets, but accept the corrupted data as valid. Be sure that all IP hosts that send and receive packets over the SLIP line use UDP checksumming, and don't blindly set the checksums to zero. At least one major UNIX implementation does not check or set the UDP checksum by default.

### SLIP Setup Example

Figure 3-58 shows a possible setup to use SLIP with modems. Note that we have tried to show the usage of many of the options; your situation may not require any options other than the factory defaults.

```
Local> DEFINE PORT 4,5 SLIP ENABLED MODEM ENABLED FLOW NONE
Local> SET SERVER SLIP PASSWORD "slipag"
Local> DEFINE SLIP PASSWORD "slipag"
Local> DEFINE SLIP PORT 4 IPADDRESS 192.0.1.1. AUTOSTART ENA
Local> DEFINE SLIP PORT 5 MTU 512
Local> LOGOUT PORT 4
Local> LOGOUT PORT 5
```

**Figure 3-58: SLIP Setup Example**

In order, these commands will:

- ♦ Allow ports 4 & 5 to run SLIP, and set them for modem control and no flow control. Recall that XON/XOFF flow control cannot be used for SLIP, and hardware flow control is not valid with modem controls.
- ♦ Set a SLIP password that must be entered by any port issuing a Set SLIP command.
- ♦ Set the server-wide SLIP defaults to disable broadcast IP packets across SLIP sessions and set an MTU based on the SLIP specification.
- ♦ Configure port 4 to autostart SLIP to one IP address. Whenever port 4's serial connection is active (DSR asserted), it will interpret SLIP packets. The Local> prompt will not be available in this case until Autostart is disabled.
- ♦ Port 5 has a different MTU than the server default; since no IP address is specified for port 5, it will have to be specified in the Set SLIP command.
- ♦ Logout both ports to have the Define Port and Define SLIP commands take effect for them.

## Menu Mode

Ports that don't need to (or due to security concerns, cannot) have access to the ETS' command line can be put into Menu mode. In this mode, users will be given a preset menu of up to 36 commands to choose from when they log in, and are not given access to the command line. Individual ports are configured for menuing with the Define Port Menu command. The commands that users see in the menu system are configured via the Set/Define Menu command. Each Set Menu command takes 3 parameters - an entry number (which menu slot the command will be seen in) an option name (what the users will see in that command slot) and the command string that will be executed if the user selects that entry. For example, the command displayed in Figure 3-59 will put an entry in menu position 5.

```
Local> SET MENU 5 "Show net TCP hosts" "SHOW HOSTS"
```

**Figure 3-59: Set Menu Command**

The menu users will see the string **Show network TCP hosts**, and if they select this entry, the command Show Hosts will be executed for them. Note that they cannot see or change the command that is executed for any particular menu choice. The command Exit can be assigned to any menu slot; users who select that option will exit Menu Mode and be returned to the command line.

Menu commands are subject to the current privilege level of the user logged into the port. If the port has Security enabled, that may restrict the command set even if the command is listed in the menu. Also, one of the commands provided should be **Logout**, so that the port can be cleared and reset for new users. Typing <Ctrl-D> or <Ctrl-Z> at the menu prompt will also log the user out.

## Rebooting

Rebooting the server should not need to be done, but the process is quick and safe enough to be done whenever necessary. Care should be taken before rebooting, though, to see that all user sessions are logged out first, if possible. Disconnecting the sessions neatly will ensure that the network communications exit properly and may prevent failed connection attempts after the reboot. It is also courteous, but not necessary, to warn users that the server will be going down by sending a Broadcast command. The privileged user can use the Broadcast All command to notify all ports of an impending shutdown.

When all ports have been logged out, you can reboot the server with the Initialize command. The syntax is displayed below [Figure 3-60]:

```
Local> INITIALIZE SERVER DELAY 5
```

**Figure 3-60: Initialize Command**

This example will schedule a reboot in 5 minutes. The Show Server Status command will show the time remaining until any pending reboot.

Options to the Initialize command can be used to force the server to remain in the Boot Configuration Program (Initialize Noboot), reload the Flash-ROMs (for flash-equipped units, Initialize Reload) or restore the factory-default NVRAM settings (Initialize Factory, discussed next). The Noboot and Reload options requires ROM support, and may not be functional in older units (ROM versions prior to P (ETSs) or E (EPSs)).

Remember that any settings (port, server, service, host table, etc) that were Set but not defined or saved will be lost by rebooting. Use the Save command to save any settings you wish to keep across boots.

## Restoring Factory Defaults

Should it become necessary, the ETS can be restored to the default settings installed at the factory. Note that this will restore **everything** - the ETS will function as though it just came out of the packing box. Be certain you wish to do this before you start.

There are 2 ways to restore the factory settings. As discussed above, the Initialize Factory command will reboot the server with the factory settings.

The factory settings can also be restored from the Boot Configuration Program (BCP). See Appendix D, "Boot Configuration Program," for more details. To return to the Boot Configuration Program, you must either force the server to fail to download the boot file or use the Initialize Noboot command. Forcing the download to fail can be done either by disconnecting the MAU (or AUI cable) from the ETS prior to rebooting or by resetting the download filename to a filename that will not be found at load time. For example, if there is no file named JOEBOB.SYS on the load host, you could enter the following [Figure 3-61]:

```
Local> DEFINE SERVER SOFT JOEBOB
```

**Figure 3-61: Forcing a Download to Fail**

Note that if you have a BOOTP server running, you will have to remove the MAU from the server, since the BOOTP server will set the ETS' load filename after you change it.

The ETS will reboot at this point, and the Boot Configuration Program can be started by pressing <Return> or <Break> on the console if the ETS is waiting to download the file. When in the Boot Configuration Program, you can use the **Flush NVR** command to restore the ETS' default settings [Figure 3-62].

```
Boot> FLUSH NVR
```

**Figure 3-62: Flush NVR Command**

The server will reboot again, this time with the factory default settings. If you removed the MAU, you will have to replace it for the reboot to succeed. The server will attempt to load the default filename (ETS.SYS, or EPS.SYS for the EPS) and restart operation. You will need to reenter the server's IP and loadhost addresses and filename for TFTP booting, unless a BOOTP server will provide the information. The server can then be booted and reconfigured at this point.

## Reloading Flash-ROM units

This section describes the functionality of the Flash-ROM ETS/EPS units (models ETSxxF, EPSxxF, and all EPS1 and EPS2's). These parts allow the server to download code from the network and store that code in ROM, so that the server can boot without network assistance from them on. The units are shipped with the latest server download code installed at the factory, so they should boot from flash ROM (with no network support) as soon as they are powered on.

Reloading the code into the ROMs (to load a newer version of the operational code, for example) is a straightforward process. The operational code is downloaded from a network host (via TFTP or MOP), the code is compressed in RAM, and then the code is programmed into the Flash ROMs. There are 2 ways to force the unit to redownload and reprogram its stored code. First, the Initialize Reload command (from operational mode) will force the ETS to reboot and attempt to reload its code. Second, the Flash command in the Boot Configuration Program (BCP) will also force the ETS to attempt to reload its code. What you will see on the console looks like the following [Figure 3-63]:

*If using an EPS1, you will not see the compression and reboot phase - the code will be loaded directly into the flash-ROMs and the unit will start operating immediately.*

```
Lantronix ETS8 Initialization
Flash Rom Version F (October 1, 1991)
Ethernet Address: (...) Internet Address: (undefined)

Lantronix XXXx Boot Configuration Program
Type HELP for more information.

Boot> FLASH

Requesting BOOTP: no valid reply received.

Attempting MOP boot: file <ETS.SYS>
        downloading from aa-04-00-1c-32-44
        .text @80000 (1205 bytes) .
        .code @0x280000 (167430 bytes) ...............
        .data @480000 (45622 bytes) .................
Loader entry point )x280000
Last Address loaded 143234 cksum = 3452345
Loaded image from host aa-04-00-1c-32-44 (292744 bytes)

Download completed - Flash load in progress [*]
Compress ++++++ done. Compression %30.5 (292744 -> bytes)

Load Completed - Boot in Progress
        Lantronix ETS-8 Terminal Server
        Ethernet Address: (...)  Internet Address: (undefin
```

**Figure 3-63: Reloading Flash ROM**

Note that the server is rebooted out of the Flash-ROMs right after downloading and reprogramming the ROMs. This is to ensure that the download code was programmed successfully. Also note that 2 checksums are maintained to ensure that the code that is read out of the ROMs is identical to that loaded into them; if the checksums are incorrect or the Flash-ROMs can not be programmed for some reason, the ETS will wait in the BCP and not boot.

Compressing the code and loading it into the ROMs will take approximately 30 seconds, during which time the console port will be unusable. The LEDs on the front of the unit will indicate continuing activity, however. If the server is powered off or otherwise interrupted during the ROM programming phase, the code in the ROMs will be invalid and the server will have to be redownloaded from the network host when next powered up.

# EPSCON NetWare Utility

*You must be Supervisor to run the EPSCON utility.*

The EPSCON utility (provided on floppy disk) allows NetWare users to configure the ETS from their PC's. It also allows logging into the ETS from the NetWare host, similar to the Telnet and LAT logins provided. It also provides a simple way to configure the fileserver print queues without having to use PCONSOLE.

*Configurations done on the ETS via EPSCON are both Set and Defined, so they are permanent and also take effect immediately.*

To use the utility, copy the EPSCON.EXE file from the floppy disc onto your fileserver. Make sure it is in your path or your path is changed to include the directory where it resides. There are 2 basic formats of commands: EPSCON and EPSCON <ETSname>. The options for each are as follows [Table 3-2, page 3-48]:

*Installing a NetWare print queue is detailed in Chapter 7,* ***NetWare Host Setup.***

| Option | Functionality | |
|---|---|---|
| EPSCON ? | Displays a help screen | |
| EPSCON Help | Displays a help screen | |
| EPSCON | Shows a list of all available print servers on the network. Non-Lantronix units may be included in the list. | |
| EPSCON *ETSname* [option [parameter]] | Configures the server *ETSname* according to the following option fields: | |
| | RESET | Forces the ETS to rescan available fileservers for jobs. |
| | SHOW | Shows NetWare protocol information from that ETS. |
| | NETSTAT | Shows network activity from and to that ETS. |
| | SERVERNAME *name* | Changes the ETS server name to *name* |
| | PRINTSERVER *name* | Sets the remote print server name on the ETS. |
| | SERIAL | Interactively configures serial ports on the ETS. |
| | No option | a login session to the ETS will be attempted and you should get a Local> prompt on the ETS. Logging out of the ETS will end the EPSCON session and return you to the DOS prompt. |

**Table 3-2: EPSCON Options**

# EPSCON Examples

The following commands force the EPS to rescan the available fileservers [Figure 3-64]:

*Recall in these examples that you must be Supervisor to run the EPSCON utility.*

```
F:\> EPSCON <CR>
Available print servers:
EPS_23e4a1          LAB_NP          Acct_PS2
LAT_0080a312a566

F:\> EPSCON EPS_23e4a1 RESET
```

**Figure 3-64: Rescanning Available Fileservers**

The following command [Figure 3-65] changes the name of the unit from LAB_NP to "LAB_NEW_ETS".

*Any EPSCON commands now must use the unit's new name (LAB_NEW_ETS).*

```
F:\> EPSCON LAB_NP SERVERNAME LAB_NEW_ETS
```

**Figure 3-65: Changin Name of Server**

To use EPSCON to configure the serial ports, give the ETS name and the Serial parameter [Figure 3-66]:

```
F:> EPSCON ETSname SERIAL
```

**Figure 3-66: Serial Parameter**

EPSCON will then prompt for settings for baud rate, parity, character size and flow control [Figure 3-67]. It will show the choices for each parameter in ()'s, and will show the default (if nothing is specified) in []'s.

```
F:> EPSCON EPS_04d233 SERIAL
Port Number [1]:<CR>
Speed (300, 600..., 64000) [9600]: 19200<CR>
Parity (None, Odd, Even, Mark, Space) [None]:<CR>
Character Size (7,8) [8]:<CR>
Flow Control (Cts/Rts, Dsr/Dtr, Xon/Xoff) [Xon]:<CR>
Enter EPS1 privileged password>
Configuration of EPS_04d233 Completed.
```

**Figure 3-67: Configuring Serial Ports**

In this case, Port 1 would be set to 19200 baud, 8-bit, no parity, with Xon/Xoff flow control. Settings are Defined and then the port is logged out, so if there is a session on that port, it will be disconnected. The ETS password is required so that only server managers can configure the ETS via EPSCON. If the password is incorrect, or the ETS cannot be connected to, the configuration will not be done.

# MACCON AppleTalk Utility

The MACCON utility allows users on Macintosh computers to configure an ETS unit via AppleTalk. It requires a Macintosh with an Ethernet card (or Ethernet built in) and System 6.0.2 or higher. The utility is provided on a floppy disk; it can be run from the floppy or copied to the Macintosh hard drive. We recommend making a copy of the floppy in any case, for backup reasons. To install the utility on the hard disk, simply drag the file from the floppy disk to the appropriate folder on the hard drive. MACCON does not use any system or other resources, so it can be anywhere on the drive. To run the utility (either on floppy or the hard drive) simply double click on the MACCON icon.

When started, MACCON will use NBP to find all configurable Lantronix devices. Double-clicking on one of the nodes will open a configuration window for that unit. Via the edit bars and buttons, you can change the server's name, service names, and serial port settings (make sure they match with what any attached printers expect). Settings changed via MACCON are both Set and Defined, so they are permanent and also take effect immediately.

When you are done configuring, click the **Quit** button. A Help button is also provided to show basic features and any keyboard shortcuts.

# 4 User Sessions

# 4 User Sessions

# Logging into the ETS

Each terminal port on the ETS can support an interactive user or other device such as a printer. If you are using the server from a terminal, you will generally be logging in to the ETS frequently. Use the following steps to log in to the server:

1. Press the <Return> key on your terminal until the server responds. If you see garbled characters or get no response after four or five <Return>s, check cable connections and the terminal's baud rate, parity and character size settings. Both the terminal and server port must agree on these settings. See your server manager if you are not sure about this.
2. If the server manager has enabled password protection you will see "`Login Password>`". Type the login password and press the <Return> key.
3. If this port has no permanent username associated with it, the server will respond with "`Username>`". You can enter up to 16 characters as a username which will identify you, or type <Ctrl-Z> to use the port name as the user name.
4. The server will respond with one of 3 things:
   a. If you see the "`Local_x>`" prompt, where "*x*" is your port number, you are now in the ETS' **local mode**. Type Help if you need information about the server's command set.
   b. If you receive a menu screen, you are in menu mode, and you can only enter the commands listed in the menu screen.
   c. If the server logs you out and attempts to reconnect, your port is in dialback mode and the server is attempting to verify your username and connection location.
5. If you are in menu mode, you are restricted to the displayed commands. If you are at the Local prompt, you can enter any ETS commands that you are privileged for.

# Logging out

When you're through using the ETS, type **Logout** at the `Local>` prompt. You can also type either <Ctrl-D> or <Ctrl-Z> at the local prompt to log out. If you have connected sessions, they will be closed before you logout. The server will then await another login. You may, of course, remain logged into the server (with or without active sessions) perpetually. There are two special cases, however: If you leave connected sessions for long periods of time, many hosts are set to log off users after a set "inactive" time. This will not affect your login to the ETS, but you will have to log in again to that remote host. Also, the ETS supports an "inactivity logout" timer of its own. If the server manager has enabled this feature, you will be logged out of the server after a preset period of inactivity. This period can range from 15 to 120 minutes. (Inactivity is defined as no keyboard typing or received network packets.)

The other method of logging out is automatic if the DSRlogout option is set. When terminal equipment is turned off or disconnected, the server will detect it and log out your port automatically.

# Configuring Your Port

There are many options you can set up to customize your port. Refer to the Commands chapter for a full description of the following Set Port and Define Port command options:

- **Autoconnect** and **Preferred Services** - your port will try to connect to any Preferred service as soon as you log in. The preferred service must also be defined.
- **Autoprompt -** determines whether a login prompt is sent back immediately when your port connects to a LAT service. No Autoprompt means that you will have to press <Return> to obtain the login prompt. Telnet and Rlogin sessions always return a login prompt upon connecting.
- **Groups** - control which of the port's Authorized Groups are enabled. The Authorized Groups are set up by the server manager.
- **Loss Notification -** your terminal will beep if characters are dropped on the connection with the ETS.
- **Username -** set up a username for your port. Note that if you Define a username, you will not be prompted for it at login time.
- **Verification -** enables messages when you connect, disconnect, or switch from a session.

In addition, you can set the Speed (baud rate), Parity Flow Control, Type (ANSI, Softcopy, etc.) and Termtype (vt100, etc) of the port.

## Privileged Commands

There are several port options that the manager (or privileged user) can set. These usually are used when configuring the port for the first time, and are rarely changed. They include the following Set Port options:

- **Access -** whether the port is for Local equipment, is for Remote users (such as a local printer or dial-out modem), or can service both (Dynamic). Dynamic would be used for dial-in/dial-out modems, or for devices such as teletypes that can be used both interactively and as print devices.
- **Authorized Groups** - all the service groups this port can possibly enable. Use the Set Port Groups command to specify which of the Authorized groups are actually enabled.
- **Autobaud -** lets the ETS attempt to automatically set the port's speed and parity based on the first characters typed at login (they must be <Return>s). Autobaud works if the port is at least 1200 baud and is using eight bit characters or seven bit characters with even parity.
- **Dedicated -** sets the port up so that it can only connect to one specific service and nothing else. When a user logs in on the port, he will be connected; when he logs out of the remote service, he will be logged off the ETS.
- **Name -** sets the port's name (not the username, which is associated with *who* is using the port).
- **Password -** if enabled, this port must input the server password to log in.
- **Security -** if enabled, the port cannot execute restricted commands.
- **Session Limit** - sets the maximum number of sessions this port can have open at once. The session limit must be less than or equal to the server's session limit.

# Connecting to Services

Since only using your ETS in local mode is not terribly fulfilling, you will probably want to connect to remote hosts. Remote here refers to any host beyond your ETS, whether it's in another country or in the same office as you are. In addition, you can access the ETS from within your UNIX or VMS session using LAT, Telnet, or Rlogin. This means that you can remotely connect to the server to access its services just as if you had logged in locally. You can also access remote printers that are not directly connected to your system. The following section explains how to do these tasks.

## Connecting to LAT Services

To connect to a LAT service, use the Connect command and give the service name, for example [Figure 4-1]:

```
Local> CONNECT HARVEY
```

**Figure 4-1: Connecting to LAT Service**

This will connect you with the highest rated service named **Harvey** on the network (see the discussion of ratings in the ETS Concepts chapter for more information about ratings).

You can also force a connection to an explicit host and specify exactly which port (if applicable) of that host to use. This may be necessary because more than one host on a local network can provide a service. For example, the following command [Figure 4-2] will attempt to connect to the "printer" service on a port (named "port_5") of vax8.

```
Local> CONNECT PRINTER NODE vax8 DESTINATION port_5
```

**Figure 4-2: Connecting to Specific Host**

This may or may not be possible, and you will be told if the specification does not agree with reality.

If your ETS has been configured to allow incoming logins (see Set Server Incoming on page 10-58) the ETS itself will show up as a service on the network. You can then connect to the ETS from another LAT-based server. Using the Show Services command, you can see the ETS as a service that you can connect to using the Connect *ETSname* command (where *ETSname* is the name of your ETS). Connecting to the ETS will give you either the server's local prompt or a prompt for a login password. You can use the server as if you had logged in directly to the machine. Type **logout** or Ctrl-Z to exit the server and return to your local session.

Keep in mind that many LAT-supporting terminals and hosts do not have provision for a <Break> key that will be sent to the ETS. If you will need to connect out of the ETS and will need to return to local mode, set a Local Switch character that will return you to local mode.

You can connect to the ETS from a VMS host provided the ETS has been defined as a service by your system manager. Use the following VMS commands to do this [Figure 4-3]:

```
$ set host/lat ETSname

OR

$ Kermit
Kermit-32> set line LTAxxxx
Kermit-32> connect
```

**Figure 4-3: Connecting From VMS Host**

**LTAxxxx** is a LAT port that your system manager has assigned for the ETS service.

## Connecting to Telnet Hosts

To connect to Telnet hosts, you can type either one of the following commands [Figure 4-4]:

```
Local> CONNECT TELNET LABSUN
Local> TELNET LABSUN
```

**Figure 4-4: Connecting to Telnet Host**

In both cases, you can use a “text” name (e.g. **labsun**) or the numeric address of the host (such as 192.0.1.88). See the information on Host Tables and Name Servers for more information on using text Telnet host names.

You can also specify an explicit port number to log into on the remote host, if necessary, by adding “:xxx” to the end of the host name or IP address. For example, the following command [Figure 4-5] will attempt a Telnet connection to TCP port 2005 on this host, rather than the default Telnet port number.

```
Local> TELNET 192.0.1.90:2005
```

**Figure 4-5: Specifying Port Number**

On some Telnet connections, padding of Newline characters may be a problem. By default, the ETS will pad newlines with a Null character (ASCII 0x0). This is part of the Telnet specification. Two commands are provided to override this default - Set Port Telnet Pad and Set Session. **Set Port Telnet Pad** controls whether a null is sent after the newlines in an interactive session; **Set Session** controls what character is actually sent for a newline (usually Carriage Return). Refer to Chapter 10 for details on commands syntax.

## Connecting via Rlogin

If Rlogin connections are enabled on both the ETS and remote UNIX or VMS hosts, you can use the **Rlogin** command to connect from the ETS to these hosts. Note that since Rlogin can bypass the normal password-login sequence, it is a potential security problem and might not be enabled on the ETS and/or the hosts. If it is enabled, your username on the ETS is used as the name to log in under. If this name is unknown or is password protected, you will still be prompted for a login password on the UNIX host. If password protection is turned off for this username, however, you will be logged in normally on the host. The format is identical to the Telnet command [Figure 4-6]:

```
Local> CONNECT RLOGIN HARVEY
```

OR

```
Local> RLOGIN HARVEY
```

**Figure 4-6: Rlogin Command**

You can optionally specify a username to be used instead of the current one for your port [Figure 4-7]:

```
Local> RLOGIN HARVEY ROBERT
```

**Figure 4-7: Specifying Alternate Username**

## Connecting to the ETS from UNIX hosts

If incoming TCP/IP logins are enabled (again, see the Set Server Incoming command) users can log into the ETS from another UNIX host. They can also use Rlogin to connect to a service (LAT or TCP/IP) offered by the ETS. To log into the ETS using Telnet, type either of the following commands at the UNIX prompt [Figure 4-8]:

*etsname is the name of the server.*

```
% telnet etsname
% telnet IP_address
```

*IP_address is the address of the server.*

**Figure 4-8: Logging in Using Telnet**

The example assumes your system manager has set up the /etc/hosts file to include the ETS. After you type the command in, you will be connected to the server and will see the server's local prompt or will be prompted for a login password. You can use the server as if you had logged in directly to the machine. Type **logout** or Ctrl-D to exit the server and return to your UNIX session.

You can use Rlogin to connect (instead of Telnet), but the Rlogin command can also be used to connect to remote services that the ETS can see. The following command [Figure 4-9] should be used for a normal login:

```
% rlogin etsname
```

**Figure 4-9: Rlogin Command**

The command displayed below [Figure 4-10] will attempt a connection through the ETS to service **serv_name**.

```
% rlogin etsname -l serv_name
```

**Figure 4-10: Connecting to Service**

The service can be local to the ETS or on another host on the network, and can be either LAT or Telnet. LAT will be tried first.

## Default Services

You can set a default service to connect to if you do not specify a service in your Connect command. This is especially useful if you usually use one host or service regularly. Using the Set/Define Port Preferred command, you can specify the service. For example [Figure 4-11]:

```
Local> DEFINE PORT PREFERRED vax8
```

**Figure 4-11: Define Port Preferred Command**

The example above defines **vax8** as the service you will connect to if you give a Connect command without specifying the service name.

For a Telnet service, the command might be [Figure 4-12]:

```
Local> DEFINE PORT PREFERRED TELNET vax8.divc.mgmt.com
```

**Figure 4-12: Set Port Preferred Command**

Again, you can specify Telnet services as either text names or numeric IP addresses. In the case of the previous command, the line displayed in Figure 4-12 would connect you with vax8.divc.mgmt.com.

A default Telnet service is different from a default LAT service. If you have only a default Telnet service set up but type **Connect**, you will get an error, since there is no default LAT host set up (and Connect without the Telnet option tries to open a LAT connection).

TCP/IP hosts on a network are unique across the network, so there is no need to differentiate Telnet nodes with the Node and Destination parameters.

# Session Management

A connection to a service or host is called a **session** on that host. One of the useful features of the ETS is that it allows you to have multiple sessions, on different hosts, and gives you easy access to all of them. The sessions you have active can be LAT, Telnet, or Rlogin sessions or any combination of the three.

It is useful to use the term "local mode" to describe actions at the server (as opposed to being active on a remote service). While in local mode, you can switch to a different session, list which sessions you have open, change your port characteristics, etc. You can then use the Resume command (see below) to reconnect to any active session. You can switch to local mode by pressing the Break key on your keyboard. This will keep your current session active while you are in local mode. As a note, not all keyboards' Break key function alike. Some keyboards' dedicated Break key will not be recognized as such by the ETS. You can define another key to be your own local break key -with the Set Port Local Switch command.

To list the sessions you have active, use the Show Sessions command when you are in local mode. It will show you the type of connection, the host name, as well as any Node or Destination information. It will also show which service is the current one. You can use the Set Session command to change the characteristics of your connections as well.

Switching between sessions is equally straightforward. Two local mode commands, Forwards and Backwards, allow you to cycle through your sessions. The open sessions can be thought of as a list of sessions, ordered by when you started them. The first session begun will be at the head of the list, while the most recently started will be at the end. **Forward** moves you toward the most recent session, while **Backward** moves you toward the oldest. The list is circular. Both commands move to the opposite end of the list if you pass one end.

As with the Break key, Forward and Backward can be bound to specific keys (see the Set Port Forward and Set Port Backward commands). If they are bound to keys, you do not have to re-enter local mode to switch sessions; just press either key and the ETS will switch sessions for you. Pressing the Forward or Backward keys is identical to going back to local mode and typing the command explicitly. These keys also function in Local Mode - if you press the defined Forward key from the local prompt, it is the same as typing Forwards as a command. Unlike the Break key, Forwards and Backwards are not assigned to any particular key by default. Like the Break key, these should not be bound to keys that could be typed in normal use.

Here is an example of session switching:

Bob connects to host **Irvine**. He works for a while, then presses the Break key to return to local mode, and connects to host **Dayton**. He works some more, then he decides to break and connect to host **Sanfrano**.

Then he presses Break and types **Show Sessions**. At this point, his session list looks like this:

1. Irvine
2. Dayton
3. Sanfrano (where Sanfrano is the current session)

If Bob presses his Backward key or enters the Backward command, he will resume the session on Dayton because Backward resumes the previous session. Pressing it again puts him in the Irvine session. Pressing the Forward key twice from there will move him forward in the list, to the Dayton session and then Sanfrano. Pressing the Forward Key once more from Sanfrano moves him to the other end of the list, to Irvine.

Another command you need for managing sessions is Resume. From the local prompt, Resume returns you to a session. If you do not specify a session number, Resume will put you back in your current session (the last one you were in). If you specify a valid number, you will be placed back in that session. In our example above, Bob could type **Resume 2** and he would be back in the Dayton session.

Finally, the Disconnect command terminates the current session, a specified session, or all of your sessions. In our example above, Bob could type **Disconnect 1** and the Irvine session would terminate. Refer to the Commands chapter for the exact syntax and options for all of these commands.

# Multisessions

*Refer to **Multisessions** on page 2-23 for more information.*

If your terminal device supports multisessions (VT340 or VT4xx compatible), you can configure your port to support the multisession “windows.” After configuring your terminal to use multisessions, use the following command to enable multisessions on the server port [Figure 4-13]:

```
Local> SET PORT MULTISESSION ENABLED
```

**Figure 4-13: Enabling Multisessions**

Both windows (or multisessions) should then be available as local ports on the ETS. Connections can be started and stopped on each session as if each was a physical port. Multisessions can be disabled in one of two ways: explicitly disabling them at the server’s local prompt, or by disabling them on the terminal device. The latter option will log out both sessions.

# Local Server Commands

Each local port can use the following local mode commands: Broadcast, Lock, Logout, or Test Port.

## Broadcast

This command is used to send messages to other users on your ETS. The server manager can disable the Broadcast command, so the command may not be usable on your server. If it is enabled, you can Broadcast messages to either a port number or a user. If a user is specified, the username of each port will be used to compare (i.e., if more than one port has the same username, the message will go to the first user with that name). If you do not want to receive other ports' messages to you, you can enter Set Port Broadcast Disabled to refuse incoming broadcast messages. A typical broadcast command would be [Figure 4-14]:

```
Local> BROADCAST 8 “Host Vax5 is down - use vax9”
OR
Local> BROADCAST Bob “Ready for lunch ??”
```

*The message must be in quotation marks.*

**Figure 4-14: Broadcast Command Example**

## Lock

Used to disable access to your port without actually logging out. If you Lock your port, you will be asked for a password and a verification (the latter is to prevent typing errors). After that, the port will be unusable until the password is again entered. Your server manager can also unlock ports if you forget your password. Like Broadcast, locking can be disabled server-wide by the manager.

## Logout

Logs you off the ETS. If you have connected sessions, they are disconnected.

## Test Port

May be used to verify wiring and flow control setup problems with your serial port. Test Port will send an identical stream of characters to the terminal, continuously or for a specified number of repetitions. If your connection is regularly dropping characters, you may be able to see it. If multisessions are enabled on a port, only that port can issue a Test Port for itself.

# Status Displays

Various display commands are provided on the server to show the availability of hosts and services as well as the state of the ETS, its ports, and the local services.

The Monitor command is also provided to show a continuous display, rather than a one-time look. The List command is provided to show the current setup in permanent memory (i.e., what will take effect after the next reboot or logout).

Note that the Node, Host and Service displays allow you to use “wildcards” for the host names. Wildcards are characters that can be substituted for a single character or a series of characters.

The List command is a special case of the Show commands, as some displays make no sense to list. Sessions and Users, for example, have no meaning in the permanent configuration data. List Nodes also has no meaning, as it only refers to the remote nodes.

- **Netstat** — Shows currently active UDP/TCP/IP and LAT sessions.
- **Show Dialback** — Shows the currently configured dialback strings.
- **Show Hosts** — Displays the TCP/IP hosts the ETS knows about. These are hosts that were resolved with a connect attempt, were seen with Rwho broadcasts, or were explicitly added to the host table with the Set Host command. If the Local option is given, only the hosts on the local network segment (based on IP subnet mask) will be shown. Show Protocols shows protocol-related counters and settings. There are sub-screens for LAT, TCP/IP, Netware and AppleTalk.
- **Show IPsecurity** — Shows any restriction on incoming or outgoing TCP/IP sessions and the ports to which those restrictions apply.
- **Show Menu** — Shows the currently configured menu strings.
- **Show Nodes** — Shows the LAT nodes on the network that offer services, along with their LAT protocol version. The Status option shows hardware addresses and any services the node offers.
- **Show Port** — Shows the current setup of your port. Settings such as flow control, parity, default preferred hosts, and username are shown, as well as enabled settings such as Broadcast, Lock, Messages, etc.

- **Show Server** Shows the state of the ETS, including hardware and software addresses, LAT timers, session and queue limits, etc. There are also counters (network events), status (active users, ports, and services) and boot (reloading information) displays available.
- **Show Services** Shows the services that are available to your port. Note that this is subject to any group restrictions that are in place (i.e., you cannot see services that you are not eligible to connect to). It shows local as well as remote services. The Characteristics option displays more detailed information on the service, such as its rating and groups. The Status option shows availability as well as the offering node. Using the Local option instead of a service name shows only the services provided by this ETS.
- **Show Sessions** Displays your active sessions (if any). The command displays the host/service you are connected to as well as the session number and type of connection.
- **Show SLIP** Shows current and default SLIP connection information It also shows packet counters for current SLIP sessions.
- **Show Users** Displays the users on the ETS. Each user is shown by name, along with his/her port number and the current connection status.

# 5 AppleTalk Host Setup

# 5

# AppleTalk Host Setup

# AppleTalk Configuration

AppleTalk configuration is extremely simple for EtherTalk nodes such as Macintoshes. If AppleTalk is enabled on an ETS service, the service should be visible via the Chooser on all networked Macintoshes. This assumes that the Macintosh using the Chooser has access to the AppleTalk zone that the ETS is a part of. If there are no AppleTalk routers on the network, or the router's default zone is the correct one, the Macintosh should see the service under the LaserWriter selection of the Chooser. Selecting this ETS service will cause any future printing to be sent to the ETS. If there are multiple zones running on the network, the ETS can be configured to be part of a specific zone via the **Set/Define Protocol AppleTalk Zone** command.

Macintoshes that do not support EtherTalk will need either an Ethernet interface or an EtherTalk-to-LocalTalk router to use the ETS. If a router is used, it will provide an EtherTalk zone on the LocalTalk network, and the ETS devices will appear under that zone. If an Ethernet board is installed in the Macintosh, the user needs to change from **Built-In** to **EtherTalk** on the Networking Control Panel. In both cases, the ETS queue(s) should be visible and usable like any other LocalTalk device.

Since ETS printers aren't directly connected to the network, any programs or utilities that attempt to modify the printer's AppleTalk settings will fail. The ETS controls all AppleTalk parameters, such as zone name and job timeouts, so modifying these settings on the printer itself will have no effect. Changes to the printer's serial port will need to be reflected on the ETS port setting, as well. Note that many printers are set to use 7-bit serial characters by default; this should be changed to 8 if possible. Some special characters (the trademark symbol and Apple logo, for example) and bitmapped graphics require 8-bit characters and will not print legibly if 7-bit mode is used.

Printing to a LaserWriter via AppleTalk requires bi-directional PostScript data flow. The EPS4 and EPS12 parallel ports are not usable for AppleTalk because the parallel port is an output-only port. The EPS1 and EPS2 support the Bitronics bi-directional parallel standard, so they can be used if the attached printer also supports Bitronics. Also, if the PostScript printer attached to the ETS can not provide interactive responses to the printing host, AppleTalk printing will likely fail. Macintoshes typically require responses to PostScript queries sent to the printer.

If you are using a third-party software that provides AppleTalk for Unix (or VMS, etc.), the ETS services should be visible like any other AppleTalk printer. Print queues should be able to access the ETS as any other AppleTalk printer. Due to the variety of software packages and their configurations, setup details cannot be shown here - refer to the local documentation for details. Note that native Unix and TCP/IP printing methods (lp, lpr, etc.) are generally easier to setup and administer than non-native AppleTalk printing, and should be used when possible.

# Troubleshooting Macintosh Printing

AppleTalk printing is extremely easy to setup, but can also be very hard to debug. Since the Macintosh is expecting PostScript replies from the print device, you cannot just attach a terminal to the ETS and watch the output. Also, there is no way to "connect" to the ETS port and talk to the printer via AppleTalk - you need to do so via LAT or Telnet. There are a few options:

1. If the printer prints a banner page upon startup, examine it. It will typically tell whether the printer is listening to LocalTalk or the serial port (or both, see Step 5 below) and what the serial settings are. This may explain why printing is not working correctly.
2. See if the printer is getting any data. Most printers will blink an LED as they receive data. When the job is finished and the printer is idle, the LED will typically be on solid. If it blinks when a job is sent, it means the printer is getting something from the Macintosh/ETS. If not, see step 5.
3. Try to get an error message from the printer.

   If the Macintosh is running a print spooler (PrintMonitor, LaserSpool, etc.), it will typically report errors. Many will show a "distressed printer" icon in the upper right of the screen; opening the spooler window may show a helpful error message from the printer.

   Or, if you can connect to the printer service on the ETS via LAT or TCP/IP, try typing some keys to the printer and pressing Return. The printer will typically respond with a PostScript error message [Figure 5-1]:

```
%% Unknown command <stuff>.
%% Flushing to end of job.
```

**Figure 5-1: PostScript Error Message**

   If the message is legible, the printer is apparently receiving and sending data correctly. Type Ctrl-D to signal the end of job (and clear the error condition), and then disconnect from the service without typing any more data to the printer.

*Be sure to check that any serial adapters conform to the pinouts shown in your Installation Guide.*

4. The most likely causes of errors are incorrect serial port configurations. If the printer's error message is unreadable (or there are serial errors on the ETS' Show Port Counters display) it may indicate that the baud/parity settings don't agree. If the printer error is readable, and says things like "Unknown command <garbage>...", it may indicate a flow-control problem. For example, the ETS never sees the printer flow control, so it overruns the printer buffer and corrupted data results.

   Many printers can be configured via PostScript as well as by switches on the printer itself. This PostScript configuration would then take precedence over the switches. If you are unsure of the current saved setup, check your printer manual for a way to force the printer to obey the switch settings. This typically involves switching the printer to LocalTalk mode (see step 5, next) for a minute and then back to serial mode.

5. Many printers can have to be configured to use LocalTalk or the serial port (i.e. they do not autoselect). If the printer is set to LocalTalk, then no errors, replies, or acknowledgments will be sent from the printer to the ETS. If you have serial line analyzer, this would be visible. You can also watch the input/output counters in the Show Port Counters display. If the printer is set to LocalTalk instead of the serial port, there are typically DIP switches to change to force it to the serial port.

*See your printer's documentation for more information.*

6. Make sure Postscript is enabled on the ETS service - this ensures that the ETS will not start a new job before the last job is finished. It will force a Ctrl-D to the printer before beginning a new job, and wait for an acknowledgment before continuing. This can show up as a problem if, for example, the first 1 or more jobs print fine but later jobs are garbled or lost.

# 6 LanMager Host Setup

# 6 LanManager Host Setup

# LanManager Configuration

The following sections cover print configuration for LanManager hosts. These instructions are designed for Windows NT and OS/2 users.

Please note that sending print jobs to the ETS from Windows for Workgroups hosts via LanManager will not work reliably. It is recommended that Windows for Workgroups users spool their print jobs to a Windows NT host and print from that machine.

## Windows NT Users

To send print jobs from a Windows NT host to the ETS, the ETS must be added as a Windows NT printer. The ETS can communicate with the NT host using either the DLC or NetBios protocols, or both. DLC configuration is simpler than NetBios configuration; however, NetBios is more flexible. The ETS can queue jobs from other protocols with NetBios connections; under DLC, there is no provision for queueing on the ETS. The ETS either allows or rejects the DLC connection; if it rejects it, the NT host will retry the connection within 10 seconds. Both methods are discussed in the following sections.

## DLC Configuration

To install DLC, a service must be created with DLC enabled. Figure 6-1, below, displays an example.

```
Local_1> SET PRIVILEGED
Local_1> system (not echoed)
Local_1>> SET SERVICE NTX DLC ENABLED PORT 4
```

**Figure 6-1: Enabling DLC**

Note that only one service on the ETS can have DLC enabled, since there is no way to differentiate by service name within the DLC connection itself.

*If the printer driver isn't already installed, you'll need the Windows NT installation disks.*

Open the NT PrintManager; its icon is located in the Main window on the desktop. Choose **Create Printer** from the Printer menu. In the dialog that is displayed, choose the following:

- Name of the queue on the NT host
- A description string
- Type of printer driver required

If applicable, choose the **Share this printer on the network option**. (This is not recommended until the print queue is confirmed to be running properly.)

Click the Print To menu arrow to view its pull-down menu. Scroll to the **Network Printer** option. From the dialog that appears, choose the **Hewlett Packard Network Port** option and click the OK button.

Click the Card Address menu arrow to view its pull-down menu; all known DLC nodes will be listed. Select the ETS' hardware address. Enter a name to be associated with this address at the prompt.

Click the **Timers** button. In the dialog that appears, click the **Job Based** radio button. Click the OK button in each print dialog displayed.

If you'd like this printer to be the default printer, click the **Default** menu arrow on the Create Printer window's title bar and scroll to the printer's name.

### NetBios Configuration

To install NetBios, create a service with LanManager enabled [Figure 6-2]. Note that in the example below, we've also used the Set Server command to give the server a name.

```
Local_1> SET PRIVILEGED
Local_1> system (not echoed)
Local_1>> SET SERVER NAME GRAPHICSERVER
Local_1>> SET SERVICE NTX LANMANAGER ENABLED PORT 4
```

**Figure 6-2: Enabling LanManager**

Choose one of the lpt ports to redirect (typically the ports to choose from will be lpt1, lpt2, or lpt3.) Before you select a port to redirect, ensure that it doesn't currently have a printer connected to it.

Double-click the DOS Command Prompt icon to open a DOS session. At the DOS prompt, use the NET USE command to indicate that the lpt port will be redirected to the ETS service [Figure 6-3].

```
C:> NET USE LPTn: \\GRAPHICSERVER\NTX
```

**Figure 6-3: Net Use Command**

The Windows NT node will attempt to connect to the ETS. (If it cannot connect to the ETS, it will inform you of the problem.) At this point, all references to LPTn (the port specified with the NET USE command above) will go to the service that you created. For testing purposes, you can try to use a copy command, if the ETS is ready to accept data.

Exit the MS/DOS shell. Open the NT Print Manager; its icon is located in the Main window on the desktop. Choose **Create Printer** from the Printer menu.

In the dialog that is displayed, choose the following:

- Name of the queue on the NT host
- A description string
- Type of printer driver required

Click the **Print To** menu arrow to view its pull-down menu, and select the LPT port you wish to redirect. Click the **OK** button in each print dialog displayed.

Please note that Windows NT does not ship with a basic, text-only printer driver. If you wish to print to a terminal, you'll need to use some type of line printer driver and will see the embedded print codes.

If you'd like this printer to be the default printer, click the **Default** menu arrow on the Create Printer window's title bar and scroll to the printer's name.

## OS/2 Users

OS/2 does not support the DLC protocol; NetBios must be used. To install NetBios, create a service with LanManager enabled [Figure 6-4]. Note that in the example below, we've also used the Set Server command to give the server a name and defined port 4 to remote access.

```
Local_1> SET PRIVILEGED
Local_1> system (not echoed)
Local_1>> SET SERVER NAME GRAPHICSERVER
Local_1>> DEFINE PORT 4 ACCESS REMOTE
Local_1>> LOGOUT PORT 4
Local_1>> SET SERVICE NTX LANMANAGER ENABLED PORT 4
```

**Figure 6-4: Enabling LanManager**

Choose one of the lpt ports to redirect (typically the ports to choose from will be lpt1, lpt2, or lpt3.) Before you select a port to redirect, ensure that it doesn't currently have a printer connected to it.

Open an OS/2 session. At the OS/2 prompt, use the NET USE command to indicate that the lpt port will be redirected to the ETS service [Figure 6-5].

```
C:> NET USE LPTn: \\GRAPHICSERVER\NTX
```

**Figure 6-5: Redirecting LPT Port**

The OS/2 node will attempt to connect to the ETS. (If it cannot connect to the ETS, it will inform you of the problem.) At this point, all references to LPTn (the port specified with the NET USE command above) will go to the service that you created. For testing purposes, you can try to use a copy command, if the ETS is ready to accept data.

Exit the OS/2 shell. Open the OS/2 PrintManager. Choose **Printer install** from the Setup pull-down menu. In the Printer Installer dialog box, select the printer driver to use, enter a queue name, and enter a description string. Click the **Connect To** menu arrow to view its pull-down menu, and scroll down to the name of the redirected lpt port. Click the OK button.

A dialog box will appear containing configuration options. Change the configuration if desired; when you're satisfied with your changes, click the OK button.

# 7

# Netware Host Setup

# 7 Netware Host Setup

The ETS can process NetWare print jobs two different ways, as a printserver or via rprinter. Under the pserver method, the ETS logs into a fileserver and then queries the server for pending print jobs that it can service. When using the rprinter method, the ETS service is actually dedicated to a particular fileserver, and the service is not usable by other fileservers (or protocols). The significant benefit of the rprinter method is that the rprinter does not use a login slot on the fileserver, while the pserver method does.

The pserver login is done under the ETS's name, not the names of the ETS services. For this reason, the ETS name will be entered with the service name in the steps below. The ETS name also has a password associated with it; the ETS will provide this password when it tries to log into the file server. This is explained below.

Configuration can be done in one of three ways. The first two both configure the ETS services as pserver devices, while the third covers rprinter setup:

1. The QINST utility (on the fileserver) is the preferred way to set up a pserver device. It will interactively prompt you for the names of the fileserver print queue to create and the ETS and service names to use. There is also a non-interactive method to create queues from within a DOS script, for example.
2. Pserver configuration can also be done via the PCONSOLE utility on the file server(s). If you are more comfortable with PSCONSOLE, or want to see exactly what is being configured by the QINST utility, this method is described below. Some basic familiarity with PCONSOLE is assumed in the examples.
3. Rprinter setup is done via the PCONSOLE utility on the file server(s). Again, basic familiarity with PCONSOLE is assumed in the examples.

# Using EPSCON to Communicate with the ETS

For NetWare, the EPSCON utility is provided on the MS-DOS distribution diskette. EPSCON can be used to either log into an EPS or send individual commands to it.

Using the EPSCON utility requires logging into the appropriate Novell file server as supervisor for NetWare 3.xx or admin for NetWare 4.xx.

To use EPSCON, type EPSCON at the prompt [Figure 7-1].

```
F:\PUBLIC> EPSCON
```

**Figure 7-1: EPSCON Command**

A list of all available commands will be displayed [Figure 7-2].

```
Epscon          Ver. 1.2
-------------------------
epscon list                    list available print servers
epscon <PsName> [port]         establish a console connection
epscon <PsName> reset          force rescan of queues
epscon <PsName> show           show Netware protocol info
epscon <PsName> show [...]     send show command to server
epscon <PsName> netstat        display server network status
epscon <PsName> name newname   set the server name
epscon <PsName> serial         set serial port parameters
epscon <PsName> reload         reload a print server

F: PUBLIC> EPSCON EPS_xxxxxx
Connection established (Ctrl-break to terminate)
# access (not echoed)

Lantronix EPS4 Version n.n/n (yymmdd)

Type Help at the 'Local_4' prompt for assistance.

Enter Username> xxxx
```

**Figure 7-2: EPSCON Connection**

Note that the "#" prompt requires the login password be entered before the connection is allowed to succeed. The default login password is **access**. This password can be changed using the Define Server Login Password command.

To end the EPSCON connection (and log out of the ETS), press Ctrl-D or type **logout**.

# NetWare Configuration Using QINST

The QINST utility, provided on floppy disk, is the recommended way to configure NetWare print queues. The QINST utility requires being logged into the SUPERVISOR (NetWare 2.2/3.1x) or ADMIN (NetWare 4.x) account on the target file server. To use the utility, simply copy the QINST.EXE file from the included floppy disk into the "PUBLIC" directory on the file server, ensure it is in the executable path and type QINST.

Note, when using NetWare Version 4.0 and greater, the QINST program requires access to UNICODE tables to provide character translation. To ensure the utility runs properly, please copy it into the NetWare "PUBLIC" directory or ensure that the PATH variable includes the required UNICODE tables.

In addition, when installing in a NetWare Version 4.x environment, the ETS requires that Binary emulation be enabled. A future version of the ETS software will directly support Novell's Network Directory Services.

The following example [Figure 7-3] shows creating a Novell print queue named TESTQ on a server named LAB_ETS using a service named LABPRT.

```
F:\> QINST
Q-Install      Ver. 1.0

Logged in as ADMIN
Installing on GONZO, NetWare V4.01
Select a volume to install queue on: (only for NetWare 4.x)

Number    Name
--------------
0         SYS
Volume number: 0 <CR>
Enter the name of the print server.
: LAB_ETS <CR>
Enter the name of the queue to create.
: TESTQ <CR>
Adding print queue TESTQ on volume GONZO_SYS
Enter the service name on LAB_ETS which will service this
queue.
[EPS_ETS_P1]: LABPRT
Adding print server LAB_ETS. Please wait...
Attaching LAB_ETS to TESTQ
Adding print server LABPRT. Please wait...
Attaching LABPRT to TESTQ

Print queue installed successfully. Resetting LAB_ETS.
Resetting print server.

Install another queue [y/n]? n

F:\>
```

**Figure 7-3: Creating Novell Print Queue**

Use the NPRINT command to print a job to the EPS [Figure 7-4].

```
F:\> NPRINT C:\AUTOEXEC.BAT /QUEUE=TESTQ_PRT
```

**Figure 7-4: NPRINT Command**

# NetWare Access List

By default, the EPS will only scan local file servers (file servers one hop away) for print queues to service. File servers on non-local Ethernets, i.e. with a IPX router between them and the EPS, can be enabled using the following command [Figure 7-5]:

```
Local_1> SET PRIVILEGED
Password> SYSTEM (not echoed)
Local_1> SET PROTOCOL NETWARE ACCESS fileserver
Local_1> DEFINE PROTOCOL NETWARE ACCESS fileserver
Local_1> SET SERVER NETWARE RESET
```

**Figure 7-5: Enabling Non-local File Servers**

To enable scanning for jobs on all file servers in the extended network, replace the file server name with the keyword "**All**". However, on an extended (wide or local-area) network with many file servers, specifying "**All**" can severely impact the time between jobs and the overall printing performance.

## NetWare Queue Password

The default login password on the EPS is used as the password that the EPS logs into NetWare file servers with. If the login password is changed, NetWare print queue setups must also be changed to reflect the new password.

## PCL Printing From NetWare

Printing PCL jobs requires an 8-bit clean data path between the NetWare file server and the printer. If printing via the serial port, hardware flow control must be used. In addition, the Binary attribute must be enabled on the service being used [Figure 7-6].

```
Local_1> SET PRIVILEGED
Password> SYSTEM (not echoed)
Local_1> DEFINE PORT 1 FLOW CTS
Local_1> SET SERVICE EPS_xxxxxx_S1 BINARY ENABLED
Local_1> DEFINE SERVICE EPS_xxxxxx_S1 BINARY ENABLED
```

**Figure 7-6: Enabling Binary**

## PostScript Printing From NetWare

PostScript printing from Novell requires the Silent/Verbose switch (Switch 2 on the EPS) be set to Silent mode. Leaving the switch in Verbose mode will result in boot-time diagnostic messages which may cause the printer to be unavailable. In addition, the PostScript attribute must be enabled on the service being used [Figure 7-7]:

```
Local_1> SET PRIVILEGED
Password> SYSTEM (not echoed)
Local_1>> SET SERVICE EPS_xxxxxx_S1 POSTSCRIPT ENABLED
Local_1>> DEFINE SERVICE EPS_xxxxxx_S1 POSTSCRIPT ENABLE
```

**Figure 7-7: Enabling Postscript**

# Pserver Configuration Using PCONSOLE

NetWare configuration involves 3 major steps:

1. Tell the NetWare file server which print resources (services) are on the ETS.
2. Assign print queues on the file server to use the ETS resources.
3. Tell the ETS to re-poll the file servers because of the updated queue information.

These steps must be done on each file server that will need access to the ETS queues. Users on each file server wishing to print will specify the file server's queue name; the file server and ETS will decide which, if any, of the ETS's queues can service the requests.

You will need to note the name of the ETS from the back label. The following steps refer to NetWare v3.11, but are similar for v2.x:

1. Log in as Supervisor on the file server you will be changing.
2. The remaining steps are done via the PCONSOLE utility. Type PCONSOLE at the F: prompt to start the utility [Figure 7-8].

```
F:> PCONSOLE<CR>
```

**Figure 7-8: Starting PCONSOLE**

a. You must tell the file server what print resources are on the network. Select **Print Server Information** from the **Available Options** menu by using the cursor keys. Press Enter [Figure 7-9].

```
Available Options

Change Current File Server
Print Queue Information
Print Server Information
```

**Figure 7-9: Print Server Information Option**

**b.** You will be shown a list of current print servers. You need to enter the ETS's server name and the names of the resources on the ETS that will be visible to the file server. After adding the ETS resources, they will be visible in this list.

Press **Insert** to create a new entry and add the ETS name and press **Enter** [Figure 7-10].

```
Enter Print Server Name: ETS_LAB
```

**Figure 7-10: Entering ETS Name**

This is the name that the ETS will log in as when querying the file server's print queues. Note that if you later change the ETS's name, you will need to update the file server(s).

If you ever change the **login password** on the ETS, you will need to add a password for the entry you just added. Highlighting the ETS name and pressing **Enter** shows the "Print Server Information" menu, where you change the password for the ETS name. If the ETS password is the factory default, it is not send to the fileserver and therefore does not need to be added explicitly.

**c.** Use **Insert** to add the name of the service on the ETS, and press **Enter**. We'll assume a service named **ETS_LAB_S1** below. Repeat for all services on the ETS that will have NetWare access. Then return to the main **Available Options** menu by pressing the Escape key.

**3.** The next step is to create a print queue on the file server and associate it with the ETS services. Select the **Print Queue Information** menu option under "Available Options" and press the Enter key [Figure 7-11].

```
Available Options
-----------------------------
Change Current File Server
Print Queue Information
Print Server Information
```

**Figure 7-11: Print Queue Information Option**

This shows a list of configured print queues on the file server. Press **Insert** to create a new queue on the file server, type the new queue name and press **Enter**. Figure 7-12 displays an example of a queue name.

```
New Print Queue Name: ETSQ
```

**Figure 7-12: Queue Name Example**

The name does not have to be related to the name of the ETS resources, but should be short and convenient for users to remember.

**a.** Highlight the queue you just entered and press **Enter** to configure the queue itself. The following menu will appear [Figure 7-13]:

```
Print Queue Information
-----------------------------
Current Print job entries
Current Queue Status
Currently Attached Servers
Print Queue ID
Queue Operators
Queue Servers
Queue Users
```

**Figure 7-13: Print Queue Information Menu**

**b.** Select **Queue Servers** and press **Enter** to specify which network print servers can print jobs from this print queue. The list will be empty, as none have been selected yet. If you press **Insert**, the resources entered above should appear in a selection list [Figure 7-14]:

```
Queue Server Candidates
-----------------------------
...
ETS_LAB
ETS_LAB_S1
...
```

**Figure 7-14: Queue Servers Option**

You need to add BOTH the ETS name (ETS_LAB) and the ETS print service (ETS_LAB_S1). The ETS name is used to log in, and the print service determines which ETS port the jobs will be sent to.

**c.** Highlight the ETS_LAB name and press **Enter**.

**d.** Press **Insert** again, highlight service name (ETS_LAB_S1 in our example) and press **Enter** again.

**e.** Press **Escape** to return to the list of print queues and repeat step 3, as needed, to create any other file server queues. Each queue must include the ETS name as one of the Queue Servers.

**f.** Press **Escape** to return to the Available Options menu.

4. When no other configuration is desired, we need to tell the ETS to re-scan the file server queues since we modified them. Return to the main Available Options menu, highlight **Print Server Information**, and press **Enter** [Figure 7-15].

```
Available Options
-----------------------------
Change Current File Server
Print Queue Information
Print Server Information
```

**Figure 7-15: Print Server Information Option**

*Use* ***Define*** *to make the setting permanent.*

Select the ETS name (ETS_LAB) and press **Enter**. The following menu will appear [Figure 7-16]. Highlight **Print Server Status/Control** and press **Enter**.

```
Print Server Information
-----------------------------
Change Password
Full Name
Print Server Configuration
Print Server ID
Print Server Operators
Print Server Status/Control
Print Server Users
```

**Figure 7-16: Print Server Information Menu**

Highlight **Server Info** and press **Enter** [Figure 7-17].

```
Print Server Status/Control
-----------------------------
File servers being serviced
Notify list for printer
Printer Status
Queues Serviced by Printer
Server Info
```

**Figure 7-17: Server Info Option**

If you don't get the menu below [Figure 7-18], the fileserver could not contact the ETS for some reason.

```
Print Server Info/Status
--------------------------------------------
Print Server Version:  3.0xx
Print Server Type:     Dedicated DOS
# of Printers          n
Queue service nodes    0
Current Server Status  Running
Serial Number          066497
```

**Figure 7-18: Print Server Info/Status Menu**

Highlight **Current Server Status** and press Enter. Select **Down** and press Enter. This will not reboot the ETS - it will only force it to re-scan the available file servers for new queue entries.

5. Press **Escape** repeatedly to exit the PCONSOLE utility.
6. Test the queue by using nprint where etsq is one of the file server queues created in step 3 [Figure 7-19]:

```
C:> nprint c:\autoexec.bat /queue=etsq
```

**Figure 7-19: Nprint Command**

The file will be spooled to the ETS for printing and should appear on the proper physical port. If the print port is in use, the NetWare job should be visible in the Show Queue display on the ETS.

# Rprinter Configuration Using PCONSOLE

As mentioned before, configuring an ETS service as an rprinter device prevents any other users from using the service. It will be tied to the rprinter node for as long as the node is running and PSERVER is executing. Only one rprinter node can be configured on the ETS, but multiple ports on the ETS can be tied to queues on that node. Before starting, you will need to know the name of the printserver process you will be starting on the fileserver, and the name of the print service on the ETS.

Rprinter support involves 4 distinct steps:

1. Set the rprinter printserver on the ETS. From the ETS local prompt, type the following text [Figure 7-20] where printservername is the name that the PSERVER NLM/VAP/standalone program will be started with (under step 4, page 7-14).

```
Local> SET PROTOCOL NETWARE PRINTSERVER printservername
```

**Figure 7-20: Setting the Printserver**

2. Create a queue on the fileserver. Steps 3a and b in the preceding section discuss creating a queue under PCONSOLE. The same process is used, with one major difference. Do **not** add the ETS node and service names, as is done in step 3c. Instead, add the print server name you will use in step 4 below.

3. Tell the fileserver about the service on the ETS that will be used. Return to the main PCONSOLE menu (**Available Options**), select **Print Server Information**, and press Enter [Figure 7-21].

```
Available Options
Change Current File Server
Print Queue Information
Print Server Information
```

**Figure 7-21: Print Server Information Option**

a. Select the ETS name from the displayed list and press Enter. The following menu will appear [Figure 7-22]:

```
Print Server Information
------------------------
Change Password
Full Name
Print Server Configuration
Print Server ID
Print Server Operators
Print Server Status/Control
Print Server Users
```

**Figure 7-22: Print Server Configuration Option**

b. Select **Print Server Configuration** and press Enter [Figure 7-23]:

```
Print Server Configuration Menu
-------------------------------
File Servers to be Serviced
Notify List for Printer
Printer Configuration
Queues Serviced by Printer
```

**Figure 7-23: Printer Configuration Option**

c. Select **Printer Configuration** and press Enter. You will be shown a list of configured printer devices (if any). Highlight the first **Not Installed** entry and press Enter [Figure 7-24]:

```
Configured Printers
-------------------
<printer name>          0
<printer name>          1
Not Installed           2
...                     3
...                     4
...                     5
```

**Figure 7-24: Configured Printers Menu**

**d.** Highlight the **Name** field and press Enter [Figure 7-25].

```
Printer 2 Configuration
Name:
Type:
...
```

**Figure 7-25: Name Field**

**e.** Highlight the **Type:** field and press Enter. You should get this menu [Figure 7-26]:

```
Printer Types
...:
Full Name
Remote Serial, Com 1
...
```

**Figure 7-26: Printer Types Menu**

**f.** Highlight **Remote Serial, Com 1** and press Enter. The printer ports under PCONSOLE are always configured as **Remote Serial** even if they are parallel ports on the ETS itself.

Pressing Escape will bring up the Save Changes menu - select **Yes** and press Enter.

**g.** Press Escape to return to the menu in step 3b [Figure 7-27]:

```
Print Server Configuration Menu
File Servers to be Serviced
Notify List for Printer
Printer Configuration
Queues Serviced by Printer
```

**Figure 7-27: Print Server Configuration Menu**

**h.** Select **Queues Serviced by Printer** and press Enter. A list of configured rprinters appears, and the name configured in step 3e should appear [Figure 7-28]:

```
Defined Printers
----------------
Change Password
....:
RPRT_SVC
...
```

**Figure 7-28: Defined Printers Menu**

Highlight that name and press Enter [Figure 7-29].

```
Fileserver    Print Queue    Priority
-------------------------------------
server1       ...            1
server2       ...            1
...           ...            1
thisserver    ...            1
...           ...            1
```

**Figure 7-29: Selecting Fileserver**

Select the name of this fileserver and press Enter (or just press Enter if no fileservers appear) [Figure 7-30].

```
Available Queues
----------------
...:
your_queue
...
```

**Figure 7-30: Available Queues Menu**

Highlight the name of the queue created in step 2, and press Enter twice (leaving the priority at +1).

**i.** Exit PCONSOLE by repeatedly pressing Escape.

4. Restart the PSERVER VAP/NLM/process on the fileserver or PC.

   a. If PSERVER is running as a VAP, enter the following [Figure 7-31] on the fileserver where pserver_name is the name of the pserver process running.

```
: STOP pserver_name
: START pserver_name
```

**Figure 7-31: Restarting PSERVER**

   The name must match the name you entered on the ETS for the PRINTSERVER.

   b. If PSERVER is running as an NLM, do the following on the fileserver [Figure 7-32]:

```
: UNLOAD PSERVER pserver_name
: LOAD PSERVER pserver_name
```

**Figure 7-32: Restarting PSERVER**

   c. If PSERVER is running on a dedicated PC, start it by doing the following [Figure 7-33]:

```
F:> PSERVER pserver_name
```

**Figure 7-33: Restarting PSERVER**

   At this point, the ETS will connect the port to the rprinter fileserver, and the port will be ready to print from NetWare and unavailable for other protocols' use. If the ETS does not connect within one minute, use the Set Protocol Netware Reset command to force the ETS to re-scan the netware connections.

# NetWare Troubleshooting

If the jobs are not printed properly, complete the following steps:

**13.** Make sure the server's name and login password were entered correctly under the PCONSOLE/EPSCON setup. If the ETS cannot log into the file server correctly, printing will not take place. If you used EPSCON, you can use PCONSOLE to verify the settings by stepping through the install above.

**14.** Make sure the ETS server name is one of the Queue Servers for each queue - if it is not, the login attempt will succeed but the EPS1 will be unable to obtain information about queued jobs and therefore never print the jobs.

**15.** Check the Show Protocols Netware screen on the ETS for any obvious errors or misconfiguration. Recall that if an access list was set up incorrectly, it will prevent the ETS from querying some fileservers.

**16.** Make sure the service(s) on the ETS have NetWare enabled. If they are not, the ETS will not query the fileservers and the print jobs will remain queued on the fileserver.

# 8 UNIX/RTEL Host Setup

# 8

# UNIX/RTEL Host Setup

This section explains how to set up your UNIX host to take advantage of the Berkeley remote printing software (LPR) and/or the Reverse Telnet software. Both protocols allow printing from the host's print spooling software.

*Please read through the entire installation/setup procedures before beginning.*

Berkeley remote printing software is very easy to setup and is the recommended method of printing if available. Not all machines (notably machines running Unix based on System V, instead of BSD) support the lpr system. No configuration (other than setting up services) is needed on the ETS to use the LPR software. In the case of the EPS1 and EPS2, the default services (EPS_xxxxxx_S1 and EPS_xxxxxx_P1) are already created and usable.

Reverse Telnet allows your UNIX host to form connections to the ETS using either a printer backend program or a named pipe interface.

All Unix printing features depend on having the ETS name resolvable. This can be accomplished by using the system's host table, using DNS or using NIS. The host table is usually a file called "/etc/hosts" that should contain the name and IP address for your server, as well as all the hosts on your network. For example, to add server *ets1* at IP address 192.0.1.22, include this line in your /etc/hosts file [Figure 8-5]:

*One of the most common causes of IP network problems is duplicate IP addresses. Please make sure that your ETS has a unique IP address.*

```
192.0.1.22        ets1
```

**Figure 8-1: Example Host File Entry**

# Selecting A Printing Method

There are four main Unix printing methods supported by the ETS. They include: the Berkeley LPD interface, the RTEL backend filter, the RTEL named pipe interface and the TCP socket interface. The advantages and drawbacks of each of these approaches will be discussed.

**LPD:** Printing with LPD is very easy to set up and requires no additional host software (assuming that the host supports printing with LPD) but has several significant restrictions. Control information about the print job is sent after the print data has been sent, so things like binary characteristics and formatting options cannot be applied. The banner page information is also sent after the print data, so a banner page is printed after the job instead of before it. In addition, since the ETS is not actually spooling the print data locally, only one copy of the job will be printed, regardless of the number of copies requested.

*The ETS can not implement all the lpr options; the print job information is not available until the print job is completed.*

**RTEL Backend Filter:** The RTEL backend filter is a program that interfaces with the host based spooling system. It receives data from the spooling system and sends the data to the ETS. Advantages of the backend filter approach include: the banner page is printed at the front of a job, multiple copies can be printed, and simple reformatting (i.e. <lf> to <cr><lf> conversion and tab expansion can be performed. However, the backend filter does not support any complicated output filtering or conversions.

**RTEL Named Pipe Interface:** The RTEL Pipe Daemon process (RTELPD) uses a Unix named pipe as its interface to the host. This allows any host or user supplied backend filter to be used for printing. Any data that is sent into the pipe is simply moved to the ETS. However, the data flow is one-way from the host system to the ETS and this approach does require one RTELPD daemon process for each print queue configured.

**TCP Socket Connection:** The ETS supports direct TCP connections to its ports. These connections provide 8-bit clean full-duplex data communication. They do require that the user provide their own software. The RTEL software includes an example program showing how to form these connections. Note that the TCP Socket interface does not support queueing. If the resource is busy when a connection attempt is make, the connection will be rejected.

# LPR on Generic Unix Hosts

The Berkeley remote printing system is supported on many machines, and is simple to configure for the EPS. Add the host print queue name into /etc/printcap, and then specify the remote nodename (the host name of the EPS) and the service name on the EPS.

For example, to add a serial port print queue for an EPS add the EPS's name and IP address to the /etc/hosts file [Figure 8-2]:

```
xxx.xxx.xxx                EPS_xxxxxx
```

**Figure 8-2: Example hosts File Entry**

Edit the /etc/printcap file and add an entry of the following form [Figure 8-3]:

```
eps_prt|Printer on LAB EPS:\
       :rm=EPS_xxxxxx:\
       :rp=EPS_xxxxxx_S1:\
       :sd=/usr/spool/lpd/eps_prt:
```

**Figure 8-3: Example printcap File Entry**

*See your host’s lpr documentation (or man pages) for details.*

Note that the punctuation shown is required, and whitespace should be avoided within each option. This will create queue named eps_prt. The “rm” parameter is the name of the EPS in the host's address file, the “rp” parameter is the name of the service as it exists on the EPS, and the “sd” parameter specifies the name of a directory used to hold temporary spooling files.

The spooling directory will have to be created using the mkdir command and should be world writable [Figure 8-8].

```
# mkdir /usr/spool/lpd/eps_prt
# chmod 777 /usr/spool/lpd/eps_prt
```

**Figure 8-5: Creating a Spooling Directory**

In addition, the "mx" option may be used to allow unlimited size files to be printed and the "sh" option may be used to prevent header pages from being generated. See the host's documentation or man pages for more information on the format of the printcap file and how to create the spool directory.

After adding the queue entry to the printcap file, it should be visible via the lpc status command [Figure 8-9]:

```
% lpc status

eps_prt:
     queuing is enabled
     printing is enabled
     no entries
     no daemon present
```

**Figure 8-6: BSD Printer Status Display**

Print to the queue using normal lpr commands [Figure 8-10]:

```
% lpr -Peps_prt /etc/hosts
```

**Figure 8-7: Printing from a BSD system**

## Notes about LPR

There are a number of important things to note about the LPR printing method:

1. Because of the way the LPR protocol is typically implemented on the host, the processing options and the banner page are sent after the job data itself. Because of this, **the EPS will print a banner page at the end of a job, and cannot support most of the LPR options**. If it is necessary to have the banner page at the beginning of the printout, install and use the RTEL software.
2. If banners are not needed, they can be disabled using the EPS commands [Figure 8-11]:

```
Local_1>> SET SERVICE service_name BANNER DISABLED
Local_1>> DEF SERVICE service_name BANNER DISABLED
```

**Figure 8-8: Disabling the LPD Banner Page**

3. By default, a formfeed is added to the end of each LPD job to force a page eject on laser printers. If this formfeed is not wanted, it can be disabled using the EPS commands [Figure 8-12]:

```
Local_1>> SET SERVICE service_name FORMFEED DISABLED
Local_1>> DEF SERVICE service_name FORMFEED DISABLED
```

**Figure 8-9: Disabling the LPD Formfeed Characteristic**

4. Many LPR spoolers are not intelligent about using multiple queues on one host. If two queues on the print host refer to two services on the same EPS, they must use separate spooling directories. If only the default directory is used, data from the 2 queues can be intermixed or sent to the wrong EPS service.

5. No special purpose input or output filters can be used when printing via LPR. If this functionality is necessary, use the named pipe interface program in the RTEL software.

## LPR on Ultrix Hosts

Ultrix hosts will need the following additional information added to the printcap entry [Figure 8-13].

```
eps_prt|Printer on LAB EPS:\
      :lp=:ct=remote:\
      :rm=EPS_xxxxxx:\
      :rp=EPS_xxxxxx_S1:\
      :sd=/usr/spool/lpd/eps_prt:
```

**Figure 8-10: Example Ultrix printcap Entry**

The additional options will show that there is no physical device for this queue and tell the host that this is a remote connection.

# LPR on AIX Hosts

LPR is available on machines running IBM's AIX operating system, but has only been tested on AIX versions 3.2 and higher. Using LPR on AIX hosts involves a slightly different configuration procedure. The queue configuration file is /etc/qconfig and the format of the entry is different. Note the lack of colons (:) and the required white space [Figure 8-14]:

```
eps_prt:
      device = eps_prtd
      up = TRUE
      host = EPS_xxxxxx
      s_statfilter = /usr/lpd/aixshort
      l_statfilter = /usr/lpd/aixlong
      rq = EPS_xxxxxx_S1
eps_prtd:
      backend = /usr/lpd/rembak
```

**Figure 8-11: Example AIX qconfig entry**

Note that the device name is simply the queue name with a “d” appended. To print, the normal lp syntax is used [Figure 8-15]:

```
% lp -deps_prt filename
```

**Figure 8-12: Printing From an AIX System**

# LPR on HP Hosts

LPR is supported in HP/UX Version 9.0 and greater. To configure a print queue using LPR, become the superuser and issue the following commands [Figure 8-16].

```
# /usr/lib/lpshut
# /usr/lib/lpadmin -peps_prt -v/dev/null /mrmodel \
 -ocmrcmodel -osmrsmodel -ormEPS_xxxxxx -orpEPS_xxxxxx_S1
# /usr/lib/accept eps_prt
# /usr/bin/enable eps_prt
# /usr/lib/lpsched
```

**Figure 8-13: Installing a LPD Queue on HP/UX**

Note that issuing the “lpshut” command will stop the HP spooling system, so this command should not be performed when print jobs are active. To print to this queue, normal lp syntax is used [Figure 8-17]:

```
# lp -deps_prt filename
```

**Figure 8-14: Printing From a HP/UX System**

## LPR on SCO Unix Hosts

LPR is supported in SCO V3.2 release 4 with TCP/IP Version 1.2 and greater. To configure a print queue using LPR first issue the following command [Figure 8-18]:

```
# mkdev rlp
```

**Figure 8-15: Using mkdev on SCO Systems**

This will install the Berkeley remote printing files and executable programs. Note that this should only be done once. To actually create a remote printer, use the command [Figure 8-19]:

```
# rlpconf
```

**Figure 8-16: Using rlpconf on SCO Systems**

This command will ask a series of questions and create a printcap entry for the specified queue. To print to this queue, normal lp syntax is used [Figure 8-20]:

```
# lp –deps_prt filename
```

**Figure 8-17: Printing from a SCO system**

# Reverse Telnet

The RTEL software is a set of programs designed to allow host initiated connections to devices attached to your terminal/print server. The RTEL software is simply a transport mechanism, and as such makes no assumptions about the actual devices attached to the server.

RTEL provides two main types of interfaces to the server. The first method provides a printer backend filter for BSD, SYSV and AIX type printing subsystems. The backend filter simply accepts data from the queue manager (LPD for BSD, lpsched for SYSV and qdaemon on AIX) and moves that data over the network to the terminal/print server.

The backend filter can perform simple reformatting of the job, i.e. tab expansion, <cr> to <cr><lf> expansion, printing a banner page, but cannot perform complicated things like postscript conversion. See the RTEL backend man page for more information.

The second method of printing with RTEL uses a Unix named pipe as the actual interface to the host system. This allows host supplied or third party software to send output to a device (the pipe) and then the RTELPD daemon process will send that data to the server. See the RTELPD man page for more information.

The RTEL software is supported on seven platforms: AIX on IBM RS6000's, Ultrix on DEC workstations, HP/UX on HP platforms, RISC/OS on MIPS workstations, SCO on PC platforms, IRIX on SGI machines, Solaris on SUN workstations and SUN/OS on SUN workstations. If the target machine is not one of these platforms, some modification may be necessary to get the RTEL software to compile, link and run properly. See the RTEL build man page for hints on building on unsupported platforms.

### Queueing with the RTEL Software

There is one important thing to note about queueing jobs to RTEL services. There are two places that a job can be queued: by the host's print system (lp or lpr, etc) or by the ETS itself. If you are only printing to one queue from one host to one ETS service, the queueing is provided by the host, as it will not release a new job to the ETS until the previous job has completed.

The second case occurs when multiple print queues (possibly on more than one host) point to the same ETS queue. In this case, the hosts will provide local queueing, but you may still have multiple jobs attempting to access the ETS. In this case, the ETS will provide queueing among the hosts. Jobs from multiple hosts are handled in order, without regard to size or any priority, so if a particular host's queue seems to stop for an extended time, this may be an explanation.

The Show Queue display on the ETS will show the order of jobs pending on the ETS. The host's lp or lpr utilities must be used to show jobs queued by the host.

## Installing the Reverse Telnet Software

Installing the RTEL software is a fairly simple process. The first step is getting the software onto the target machine. Typically, the distribution will be on a magnetic tape in tar format and can be restored into a local directory.

Set your current directory to the RTEL directory [Figure 8-21]:

```
# mkdir /tmp/rtel
# cd /tmp/rtel
# tar xvf /dev/rst0
```

**Figure 8-18: Restoring RTEL software from tape**

If the distribution arrived on a MS DOS floppy, the source code will be in a single tar archive file. (rtel_src.tar) This file contains all the source code, but does not contain any executables. Simply copy the tar archive file onto the host system, remembering to use binary mode during the copy, and untar it. Since the rtel_src.tar file does not contain any executables, it will be necessary to recompile the software on the target machine. Note that this will be done automatically for supported systems.

Once all the files are in place, issue the command ./lpinstall. See the following example [Figure 8-22].

```
# ./lpinstall

Installing the RTEL software package requires root priv-
ileges. Do you want to continue the installation [yes]

The RTEL binaries are provided for the following machine

      AIX ---- IBM RS6000 based machines
      DEC ---- Ultrix RISC (non-VAX) machines
      HP ----- HP/UX HP/PA based machines
      MIPS --- RISC/OS
      SCO ---- SCO Unix
      SGI ---- IRIX
      SOL ---- Solaris Sparc machines
      SPARC -- Sun/OS Sparc machines
      other -- (input file extension of other)

Please select your machine type: sparc
Please select type of spooling system to install [lpr]

Installation of RTEL software package complete.

#
```

**Figure 8-19: RTEL Installation**

This script moves everything necessary to run the RTEL software into the /usr/spool/rtel directory tree. The source code can now be removed from the system. The next step is to create and install backend filter program, or setup a named pipe and start the RTELPD daemon process.

# Setting up the RTEL Backend Filter

RTEL backend filters are designed to work with BSD, SYSV and AIX type printing subsystems. The backend filter simply accepts data from the queue manager and moves that data over the network to the terminal/print server. The backend filter can perform simple reformatting of the job, i.e. tab expansion, <cr> to <cr><lf> expansion, printing a banner page, but it cannot perform complicated things like postscript conversion.

*Become superuser before running these commands.*

Installing a print queue using the backend filter is very simple. First install the RTEL software according to Figure 8-23. Then execute the **mkprt** script. See the following example from a BSD type system:

```
# cd /usr/spool/rtel
# ./mkprt

Installing a print queue requires root privileged. Do you wan
to continue the installation [yes]

Please enter the name of the print queue to create: hp4

A printcap entry for hp4 needs to be created.

la75|RTEL printer:\
      :lp=/usr/spool/rtel/hp4_dev:\
      :of=/usr/spool/rtel/hp4:\
      :sd=/usr/spool/rtel/hp4_sd:

Should this entry be added to your printcap file [y]: yes

In order to establish RTEL connections, an IP address and
a host name must be assigned to the terminal server.

Please enter the server’s host name []: ps3

A connection on the server can be specified by either a
port number or a service name. A port number is simpler,
but a service allows more flexibility.

Would you like to specify a (p)ort number or a (s)ervice
name [p]: p
Please enter the port number []: 2

By default, a password is not needed to connect to a service
a port. (Note that the word null actually means no password.)
Please enter the password [null]:

The following options are available:

      banner ------- Print a banner page before every job
      binary ------- Do not format data for printing
      expandtabs --- Expand tabs into spaces
      formfeed ----- Append formfeed to end of job
      none --------- No options are to be used
      postscript --- Append <cntrl-d> to postscript job

Please enter a comma separated option string [none]: none

The following entry has been added to the server_host file

queue  server  service  password  option
hp4    ps3     :2       null      none

Installation of queue hp4 complete
```

*There must be an entry in the options field. Use* ***none*** *if no options are desired.*

**Figure 8-20: Installing Backend Filter**

The queue created (hp4) should now be ready to process requests.

The options field specifies which (if any) of the print options are enabled. The choices are one or more of the following [Table 8-1]:

| Option | Functionality |
|---|---|
| Banner | A banner page is printed before each job. The banner will show the date, name of the job, and name of the user who requested it. |
| Binary | No Line Feed processing is performed. This is useful for non-text files (such as plotter or graphics files). |
| Expandtabs | Change tabs into space characters. Eight space tabs are assumed. |
| Formfeed | Appends a formfeed (ASCII 0xC, Ctrl-L) to the end of the job. Should not be used with the PostScript or Binary options. |
| None | No options are enabled |
| Postscript | A Ctrl-D (ASCII 4) is used to terminate a print job, rather than the standard formfeed (ASCII 12). |

**Table 8-1: Print Options**

There must be an entry in the options field (use **none** if no options are desired). Multiple options can be specified, separated by commas. Options must be specified in lower case characters.

There must be an entry in the server_hosts file for each ETS print queue you wish to use. You may have multiple UNIX print queues using the same ETS print queue (using different sets of options, for example) but a UNIX print queue cannot use multiple ETS queues.

Use the **lpstat -t** command (for lp spooling systems) or the **lpc status** command (for lpr spooling systems) to make sure the printer was created correctly. For example, if we created printer **hp4** on our lp spooling system, entering the lpstat -t command might display the following information [Figure 8-24]:

```
% lpstat -t
scheduler is running
device for hp4: /usr/spool/rtel/hp4_dev
hp4 accepting requests since Apr 18 15:44
printer hp4 is idle. enabled since Apr 18 15:44
```

**Figure 8-21: lpstat -t Information**

For an LPR type spooling system, the corresponding output would look like the following [Figure 8-25]:

```
% lpc status hp4
hp4:
     queueing is enabled
     printing is enabled
     no entries
     no daemons present
```

**Figure 8-22: lpc status Information**

# Setting up the RTEL Named Pipe Daemon

The basic concept behind the RTELPD daemon is to allow a host unix system to use its own output filter programs that provide special purpose post-processing and then write the data into a named pipe device. The RTELPD daemon will read the data from the pipe and send it out over the network to a terminal or print server port.

Note that the output filter in this case is provided by the host system or is custom software. It is not provided as part of the RTEL software package.

When starting the rtelpd daemon the command line parameters will be verified. The daemon process will then wait for input to arrive on the pipe device. When input data is detected, a connection attempt will be made to the target terminal server port. If this is successful, data will start moving through the path.

If the specified service or port is currently busy, the daemon process will wait until the service or port is available and then once again try a connection attempt. This waiting appears to the process dumping data into the pipe device as if the connection is flow controlled. Eventually, the connection attempt should succeed, and the user data will flow through the path. When the user process closes the pipe device, the rtelpd daemon will detect the end-of-file, close the network connection and then restart the job cycle by waiting for input to arrive on the pipe.

Since named pipes are uni-directional devices, data will only move in one direction, i.e. data will flow from the host to the printer, but not from the printer back to the host. If setting up host application packages like NewsPrint, configure them as if they were talking to parallel attached devices. This will ensure that the software is not expecting a response back from the printer.

Installing the RTELPD daemon process and creating a pipe is very simple. First create the named pipe and setup the server_hosts file using the mkpipe script [Figure 8-26, page 8-12].

*See the supplied rtelpd man page for a complete list and description of the rtelpd options.*

```
# cd /usr/spool/rtel
# ./mkpipe

Installing a pipe device requires root privileges.
Do you want to continue the installation [yes]

This install script will create a named pipe as the in-
terface to the RTEL software. This pipe will be used as
the output device for your printing system.

Note that the full directory path must be specified for
the pipe name, i.e. /dev/rtelp1.

Note that the full directory path must be specified for
the pipe name, i.e. /dev/rtelp1.

Please enter a pipe name: /dev/rtp1

Created pip /dev/rtp1 with root ownership and world
read/write privileges. If this is not appropriate for
your application, please change these characteristics.

In order to establish RTEL connections, an IP address ar
a host name must be assigned to the terminal server.

Please enter the server's host name []: ps3

A connection on the server can be specified by either a
port number or a service name. A port number is simpler,
but a service allows more flexibility.

Would you like to specify a (p)ort nubmer or a (s)ervice
name [p]: p
Please enter the port number []: 2

By default, a password is not needed to connect to a ser
vice or a port. (Note that the word null actually means
no password.)

Please enter the password [null]:

The following entry has been added to the server_host
file

queue      server   service   password   option
/dev/rtp1  ps3      :2         null       -b

Installation of pip /dev/rtp1 complete.

#
```

**Figure 8-23: Installing RTELPD Daemon Process**

Note that this example has not actually setup a print queue. It has simply created the named pipe and put an entry in the server_hosts file that says how to startup the RTELPD daemon process.

To actually start the RTELPD daemon process, issue the command "/usr/spool/rtel/rtelpd -s" from a shell prompt. The RTEL setup can then be checked by sending data into the named pipe and seeing if it appears at the printer [Figure 8-27].

```
# cat /etc/hosts > /dev/rtp1
```

**Figure 8-24: Checking RTEL Setup**

Of course, if the printer is a postscript printer, a postscript job should be sent into the pipe. Remember to add the "/usr/spool/rtel/rtelpd -s" command to the host startup files so the RTELPD daemon will be started automatically each time the system boots.

Once the RTELPD daemon is running, a unix print queue can be configured and should use the named pipe as its output device. This allows all the normal option processing associated with host supplied or third-party backend programs.

### RTELPD Command Line Parameters and Options

The RTELPD daemon process has a set of command line options [Table 8-2]:

| Option | Description | Usage |
|---|---|---|
| b | rteld spawned as a system daemon | -b |
| c | specify a host and service to connect to | **-c***service@host* |
| e | convert <lf> into <cr><lf> | -e |
| h | help screen | *-h* |
| k | kill daemon for the specified named pipe | -k/dev/pipe |
| n | named pipe to use | -n/dev/pipe |
| p | specify a password for the remote service | -ppassword |
| r | shows all active rteld sessions | *-r* |
| s | start rtelpd daemons in server_hosts file | -s |
| v | Show version of RTEL software | -v |

*See the RTELPD man page for more information about each of these options and what they do.*

**Table 8-2: RTELPD Command Line Options**

### Creating a BSD Print Queue Using RTELPD

*See the RTELPD man pages for specific examples of configuring a print queue for each supported system type.*

To create a generic BSD print queue using the RTELPD pipe daemon, first create a named pipe as documented above. Then edit the file /etc/printcap and insert an entry of the following form [Figure 8-28]:

```
rtel_prt:\
     :lp=/dev/rtp1:|
     :sd=/usr/spool/lpd:
```

**Figure 8-25: Editing /etc/printcap File**

This will create a very simple queue. Ensure that the RTELPD daemon is running and print a job. The output should appear on the printer.

### Creating a SYSV Print Queue Using RTELPD

To create a generic SYSV print queue using the RTEL pipe daemon, first create a named pipe as documented above. Then issue the following commands [Figure 8-29]:

```
# /usr/lib/lpshut
# /usr/lib/lpadmin -prtel_prt \
                   -i/usr/spool/lp/model/dumb \
                   -v/dev/rtp1
# /usr/bin/enable rtel_prt
# /usr/lib/accept rtel_prt
# /usr/lib/lpsched
```

**Figure 8-26: Creating SYSV Print Queue**

This will create a very simple queue. Ensure that the RTELPD daemon is running and print a job. The output should appear on the printer.

See the RTELPD man pages for specific examples of configuring a print queue for each supported system type.

## RTEL Troubleshooting

There is a diagnostic file in the **/usr/spool/rtel** directory that may be helpful in finding problems. The **job_status** file contains any errors that occurred on a particular print job, such as job aborted, invalid service name, etc.This file should be examined if the RTEL queue is not performing correctly. Entries in this file are time-stamped to indicate when the error(s) took place; make sure when looking at these error logs that the errors shown are for roughly the same time as the problem(s) occurred. In addition, each error name in the log files has a name that identifies roughly where the error took place, as follows [Table 8-3]:

| Error Name | Error |
|---|---|
| %Exxxxxx | Error reported by the host operating system itself. Error names generally correspond to values for **errno**. |
| %LAT_xxxx | A reject code from the server. Generally indicates a problem with the ETS setup or the server_hosts file. |
| %RTEL_xxxxx | The host system could not initiate the connection to the server properly. The message should indicate the reason. |

*See the **job_status** file in the **/usr/spool/rtel** directory for RTEL error messages.*

**Table 8-3: RTEL Errors**

# TCP Socket Connections

If custom queueing software has been designed, raw TCP/IP (or Telnet) connections can be made directly to the ETS ports. Opening a TCP session to port 30nn will attempt a direct connection to port nn on the ETS. For example, port 3002 will connect to the parallel port on a EPS1. The 30nn range of ports is 8-bit clean.

If the port is busy or unavailable when the TCP connection attempt is made, a TCP RST will be sent back to the host and the connection will be closed. It is up to the application software to detect this condition and retry the connection.

If Telnet IAC interpretation is needed, form a connection to the 20nn range of ports.

In addition, connections can be formed to a specific port on the ETS by using the Telnet command with the “port number” parameter. For example, to connect to serial port 3 on the ETS, first make sure the port is set to Access Remote or Access Dynamic, then issue the following command [Figure 8-30]:

```
% telnet server_name 2003
```

**Figure 8-27: Telnet Command to Connect to Specific Port**

*See **Set Service TCPport** or **Set Service Telnet-Port** on page 10-78 for more information.*

A TCP socket number can also be associated with a multi-port service. This allows an application program to request a connection to a single socket and be given the next available port.

# Logging into the ETS

A Unix host machine can actually "log in" to the server from a TCP/IP host using Telnet or Rlogin. For example, if the server's IP address is 192.0.1.92, Telnetting to that address will connect and return the local prompt as if the user had logged in on a hardwired port. Incoming logins can be selectively disabled with the Set Server Incoming command. Password protection can also be enabled or disabled for all virtual logins with the Set Server Incoming [Password, Nopassword] command. If passwords are enabled, users wishing to log into the server over the network will have to enter the server's login password before they are given a command line prompt. Any login also creates a "virtual port," and thus has to conform to the 16-virtual user maximum on the server.

Once an incoming connection has been started, the user can use the server as if he or she were connected on a local port. The user can start outgoing connections, connect to local services, configure the server, etc. Users logging in to the ETS will receive their port setups from the "template port". Because of the nature of these "virtual" port connections, users cannot perform many of the port configuration functions. For example, baud rate has no applicability to network connections (neither do parity, flow control, etc.). Users are also not allowed to permanently configure any of their port characteristics, since the "port" disappears when the user logs out. Only the privileged user can configure the permanent template for the virtual ports.

Note that this login capability gives TCP/IP users a convenient way to remotely "manage" the server - managers can simply log in, change server parameters, and log out again. Recall that Telnetting to port 7000 on the ETS bypasses the Incoming restrictions, but requires entering the ETS' login password (like the NCP console facility). Note also that most network connections will not have a Break key that the ETS will receive, so setting a local switch character (either by hand or via the template port) is a good idea before connecting to services from a network login port.

## Rlogin

The ETS protocol enhancements also allow users on the ETS to **Rlogin** to remote UNIX hosts. Note that for both incoming and outgoing rlogins, the ETS must be in the host's local host table.

Rlogin can be configured to allow trusted users to log into a host without password verification. For this reason, Rlogin can be a security problem.

If login security is a concern at your site, your ETS should have Rlogin disabled and your UNIX hosts probably should not allow Rlogin access. Telnet will automatically prompt for a username and password, so it does not have this security problem.

Rlogin generally allows two methods of user verification. There may be a **hosts.equiv** file on your system that tells UNIX that some hosts are "equivalent" to yours, and that their users are safe to use this system. Thus any user from a host in this file with a corresponding username will be allowed onto your system. Users can also have a **.rhosts** file in their home directory that describes other usernames and hosts that are allowed to access their account. If the username is unknown or if neither of these files exist, the normal login/password sequence is used.

If Rlogin is enabled on your system, be very careful when setting up these two files. If Rlogin is enabled on your ETS, users can Rlogin to a host; their usernames on the ETS will become their usernames in the Rlogin request. The syntax of the Rlogin command is also identical to that of Telnet except that the Rlogin command allows users to specify a username other than the current one for logging in.

# Translating Between LAT and Telnet Protocols

The service capabilities of the ETS allow translation between LAT and Telnet protocols. A LAT host can connect to a Telnet host, and a TCP/IP user can "Telnet" or "Rlogin" to a LAT host. Note what this implies: the TCP/IP host does not need LAT software, and the LAT host does not need TCP/IP software to communicate with each other. Both translation directions create a "virtual port" to handle the sessions, and thus have to conform to the 16-virtual user maximum on the server. The two different directions (Telnet to LAT and LAT to Telnet) are set up differently, however.

*See **Configuring Virtual Ports** on page 3-11 for information on giving network logins default characteristics.*

### Setting Up a TCP/IP Host as a LAT Service

For the LAT to Telnet direction, the server must be configured to offer the Telnet host as a proxy "service." This is done with the Set Service Virtual command. Once this is done, a service can be made of a Telnet host by entering the command below [Figure 8-31] where *hostname* is the name of a Telnet host.

```
Local> SET SERVICE hostname VIRTUAL ENABLED
```

**Figure 8-28: Set Service Virtual Command**

This will set up the host as a **service** that LAT hosts can connect to. The service should now be visible to LAT hosts (if it was Set; if it was Defined, it will not appear until the server is rebooted). Setting this host to Virtual Disabled will remove it as a service. If the hostname is not in the ETS' host table, the name will be resolved when a user attempts to connect to the service. If it cannot be resolved, the service connect attempt will fail.

For example, to create a permanent LAT service for host **Beeker**, enter the following command on the ETS (you must be privileged) [Figure 8-32]:

```
Local> DEFINE SERVICE BEEKER VIRTUAL ENABLED
```

**Figure 8-29: Creating LAT Service**

*To disable Rlogin, see **Set/Define Server Incoming** on page 10-58.*

If host **Beeker** is not in the local host table, the ETS will attempt to resolve it when a user attempts to connect to the service.

Note that the Telnet proxy services cannot be used the same as regular services. The Test Service command cannot be used to test a proxy service. The Set Service Identification command can be used to change the identification string of the created service from the default **Telnet to *hostname***. If the virtual service was created using Set, using a Show Service Local command at this point would show **Beeker** as a virtual Telnet service. As mentioned above, this is a normal service as far as network users are concerned. It can be connected to from other terminal servers as well as from LAT hosts. Note that network users cannot connect directly to the ETS command line in this manner - when they close the connection to the remote Telnet host, they will be disconnected from the service on the ETS as well. If the Telnet connection fails for some reason, they will also be logged off the ETS.

### Connecting to a LAT Service from a TCP/IP host

Connecting to a LAT service from a TCP/IP host can be accomplished in one of two ways. Users can Telnet into the ETS (actually log in) and then CONNECT normally, for example, when they have a local session (see Logging In, below). If users want to avoid using the local ETS commands, they can use the UNIX **Rlogin** command to connect to a LAT service. If the Rlogin command is given without a specified username (such as **Rlogin etsbox**) the user will obtain the local prompt on the ETS (if Rlogins are enabled and there is no password protection, etc.). If, however, the Rlogin is specified with a remote username (usually the **-l** option), that “username” will be treated as a service to connect to from the server. For example, if we have host **etsbox** (the ETS) and a LAT service named **hoopla** somewhere on the network, a UNIX user could type the following text [Figure 8-33] and the ETS will attempt to connect the user to service **hoopla**.

```
% Rlogin etsbox –l hoopla
```

**Figure 8-30: Rlogin Command**

The service does not have to be local to the ETS, as the ETS will look at remote services as well when attempting the connection. Like the Telnet proxy services, the user will not have access to the ETS command line in this case; when they log out of the LAT service, they are disconnected from the ETS as well. If the connect attempt fails the user will be logged off the server. Note that the service name is not case-sensitive as far as the ETS is concerned. Users can also connect to TCP/IP hosts using this feature (although it’s obviously more inefficient than just Telnetting directly). The ETS will search both LAT services (first) and Telnet hosts (second) when it tries to match the service name specified in the Rlogin command.

In the case of a transparent connections, where the ETS local prompt is never reached, a way to disconnect from the remote service should be provided. Rlogin and Telnet both support “escape characters” (~. and ^], respectively) to return to the local TCP/IP host. Most LAT implementations do as well. If you are unsure whether an escape character (or local switch) exists, you can log into the ETS and set one explicitly with the Set Port Local Switch command, and then connect to the remote host or service.

# 9 VMS Host Setup

# 9 VMS Host Setup

This section explains how to set up your VMS host to connect to the ETS, including creating an application port and setting up a printer under LAT control. Most VMS applications that can utilize the ETS will require the setup of a LAT "application port." This is a device that allows programs to treat a LAT connection as a physical port for input and output. For example, a printer might be configured to use port LTA3419:, which might be connected ("mapped") to port 3 on the ETS. The LAT port can be mapped to either a service or an exact port on the ETS. Refer to your LATCP documentation as you read the following sections. Note that you need system privileges to create and change LAT ports.

# Creating a LAT Application Port

To configure LAT on your VMS host machine so that users can connect to a remote service, use the LATCP program to create a new, unique port, then set it to the appropriate node and service. Enter the following commands at the VMS system prompt [Figure 9-1]:

```
$ RUN SYS$SYSTEM:LATCP
LCP> CREATE PORT LTAnnnn/APPLICATION
LCP> SET PORT LTAnnnn/node=nodename/service=servname
LCP> SHOW PORT LTAnnnn
LCP> EXIT
```

***nodename*** *and* ***servname*** *are the names of the terminal server and service name, respectively.*

**Figure 9-1: LATCP Program**

The *nnnn* is any unused LAT port number (use the Show Ports command to see which port numbers are in use). VMS users can then use port LTAnnnn as a port to the ETS. When the port is used as the target of an application, such as a print queue or Kermit connection, a LAT connection with the service will be attempted.

If heavy input or output loads are expected on the LTA port, you can set alternate type ahead to reduce flow control problems on the ports [Figure 9-2]:

```
$ SET TERM/PERM/ALTYPEAHD LTAnnnn
```

**Figure 9-2: Set Term Command**

The following examples [Figure 9-3, page 9-2] show how you might create application ports for incoming logins, modems, and Telnet (proxy) services, respectively. In each case, the name of the ETS server (from the Set Server Name command) is **etsbox**.

```
$ RUN SYS$SYSTEM:LATCP
LCP> CREATE PORT LTA45/APPLICATION
LCP> SET PORT LTA45/node=etsbox/service=etsbox
LCP> SET PORT LTA46/node=etsbox/service=modem
LCP> SET PORT LTA47/node=etsbox/service=hydra
LCP> EXIT
```

**Figure 9-3: Creating Application Ports**

Notes:

- ♦ In the first example (for logins from the VMS machine to the ETS) the ETS creates a service with the same name as the ETS itself - thus the service and nodename are the same.
- ♦ In the second example, the ETS must be configured to provide a service called "modem" on one of its ports.
- ♦ In the last example, the "service" (hydra) is not an ETS-provided service; rather, it is a Telnet host that the ETS is offering as a proxy service (see below) that LAT users can connect to. Users who access the "service" will be connected to the Telnet host **hydra**. In this case, the ETS manager must configure host hydra as a "virtual service".

*See **Set Service Virtual** on page 10-79 for more information.*

To connect to the ETS service from the VMS host, use a command similar to the following [Figure 9-4]:

```
$ SET HOST/DTE lta45
```

**Figure 9-4: Connecting From VMS Host**

# Setting Up a Printer Under LAT Service

The following is an example of how a VMS host might be configured to access the ETS as a printer device. The following VMS commands [Figure 9-5] will create a new LAT device and start a print queue on it. Below, *node_name* is the name of the ETS and *serv_name* is the name of the print service on the ETS. *Port_name* is of the form **LTAxxxx**. *queue_name* is the name of the VMS queue to create and start.

*Refer to the LATCP and VMS documentation for exact syntax and details.*

```
$ RUN SYS$SYSTEM:LATCP
LCP> CREATE PORT port_name/APPLICATION
LCP> SET PORT port_name/NODE=node_name/SERV=serv_name
LCP> EXIT
$ SET DEVICE-
/SPOOL=(queue_name, SYS$SYSDEVICE) port_name
```

**Figure 9-5: Creating LAT Device**

In addition, you may want to use the following commands [Figure 9-6] on the ETS to set up terminal characteristics for the print device.

*Remember that you must set up the print service on the ETS to complete the configuration.*

```
$ SET TERM/PERM/NOBROAD/DEV=LA75/FORM/WIDTH=132 -
             port_name
```

**Figure 9-6: Setting Up Terminal Characteristics**

For example, if you want to create a new LAT device *LTA1234* that accesses print service *printer1* on the ETS named *server1* using VMS queue *remote_prq*, enter the following commands [Figure 9-7]:

```
$ RUN SYS$SYSTEM:LATCP
LCP> CREATE PORT LTA1234/APPLICATION
LCP> SET PORT LTA1234/NODE=server1/SERV=printer1
LCP> EXIT

$ SET TERM/PERM/NOBROADCAST/DEV=LA75-
/FORM/WIDTH=132 LTA1234
$ SET DEVICE/SPOOL=(remote_prq, SYS$SYSDEVICE) LTA1234
```

**Figure 9-7: Creating LAT Device**

The ETS's name must match that in the **/Node** field in the LATCP Set Port command above and the service names specified must also match. The Server Management chapter explains the ETS commands used to do this task. Example commands for the above service might be [Figure 9-8]:

```
Local> SET SERVER NAME server1
Local> SAVE SERVER
Local> SET SERVICE printer1 PORT 7
Local> SAVE SERVICE printer1
Local> DEFINE PORT 7 ACCESS REMOTE
Local> LOGOUT PORT 7
```

**Figure 9-8: Sample Commands For Service**

When the configuration is compete, create and start a VMS queue with the command [Figure 9-9]:

```
$ INIT/QUEUE/START/ON=LTA1234:/PROCESSOR=LATSYM/ -
      RETAIN=ERROR remote_prq
```

**Figure 9-9: Creating and Starting VMS Queue**

A print request might look like the following [Figure 9-10]:

```
$ PRINT/QUE=remote_prq filename.txt
```

**Figure 9-10: Print Request**

# Tips for VMS Printer Troubleshooting

If a remote print queue is not functioning properly, check the following items:

- Verify that the LTAxxxx device is mapped to the correct server and service name. Use the LATCP command **Show Port LTAxxxx** to see what server/service combination the LTA device is mapped to.
- On the terminal server, verify that the server name matches the information obtained in step one. Note that the server name should be both Set and Defined to the correct value.
- On the terminal server, verify that the service name specified matches the name obtained in step 1.

  In addition, verify that the service is available. Issue the following server command [Figure 9-11]:

```
Local> SHOW SERVICE serv_name CHARACTERISTICS
```

**Figure 9-11: Show Service Command**

  The service port should be the port the printer is physically connected to and the service rating should be 255 (or at least non-zero).

  If the service rating is zero, the service is not available. Verify that the port on the ETS is Defined access remote or dynamic. (Specified using the server commands **Define Port Access {Remote, Dynamic}**).

  Log the port out using the command **Logout Port n**. This will force the server to re-evaluate the service rating for the port.

- Verify that the LAT characteristic is enabled on the service.
- Verify that the port characteristics (baud rate, parity, flow control) match the settings needed for the printer. Remember that these characteristics need to be both set and defined.
- On the server, issue the following command to see if the host is attempting to make a LAT connection [Figure 9-12]:

```
Local> MONITOR QUEUE
```

**Figure 9-12: Monitor Queue Command**

  When a job is active, a queue entry from the VMS host to the specified service should appear.

- As a last resort, connect a terminal to the port and see if data is appearing on the terminal when a print job is attempted. You can also try connecting to the ETS service locally, to see if the ETS is configured properly.

If the printer still does not function properly after verifying these conditions, contact Lantronix technical support for assistance.

# Translating RLAT Jobs to RTEL

The ETS provides the ability to translate RLAT jobs into RTEL jobs to a third terminal server. The LAT host and local ETS are on the same network, but the remote ETS (with the print device attached) only needs to be accessible via TCP/IP. This allows LAT print jobs to traverse a router-based network (via TCP/IP) that LAT itself would not otherwise be able to do. The procedure for setting up such a case is as follows. The VMS host uses a print queue on a LAT device as usual, but the LAT device will have a destination port containing a capital "Q" followed by the name of the print service on the remote ETS. Note that the destination port is really an "environment string." See *Environment for TCP/IP and LAT* on page 2-21 for more information on the environment strings.

On the local ETS, create a virtual service mapping a LAT service to a TCP/IP "host" where the host is the name of the remote ETS with the print device. When the VMS host forms an RLAT connection to this service on the local ETS, that ETS will look at the destination port name and the service name and create an RTEL connection to the remote ETS. An example setup is displayed in Figure 9-13, below.

```
VMS> MCR LATCP CREATE PORT/APPL LTA5000
VMS> SET PORT LTA5000/NODE=LOCAL_ETS/
SERVICE=REM_ETS/PORT=QPRINTER
```

**Figure 9-13: Sample Service Setup**

This maps the LAT application port LTA5000 on the VMS host to point to a service ("rem_ets") provided by the local ETS ("local_ets"), and tells the local ETS which remote service to connect to ("printer"). The "Q" preceding the service name tells the local ETS to use the RTEL protocol to connect to the remote ETS' service. On the local ETS, enter the following commands to create a service ("rem_ets") that will connect to ETS "rem_ets" [Figure 9-14]:

```
Local>> SET SERVICE rem_ets VIRTUAL ENABLED
```

**Figure 9-14: Creating "rem_ets" Service**

Recall that service names are not case-sensitive. On the remote ETS (with the print device), enter the following commands [Figure 9-15]:

```
Local>> SET SERVICE printer PORT 7
Local>> DEFINE PORT 7 ACCESS DYNAMIC
Local>> LOGOUT PORT 7
```

**Figure 9-15: Creating Target Service**

This will create the target service (on port 7 in this example) and configure it to accept network connections.

Note that the local ETS must be able to resolve the IP address of the remote ETS (either via a nameserver, host table, etc) for the RTEL connection to work.

# The ETS$CONFIGD Download Server

The download server process needs to be installed if you plan to use the Startupfile option to download a configuration file into your ETS. This process runs on a VMS machine and waits for a download connections; it then tries to download the requested file to the ETS.

There are 2 steps required to use the download server. First, the server code must be compiled on your VMS host. The code is written in C and should build on any VMS system. Second, the download process has to be started on the VMS machine, either by hand or as part of the system startup.

To compile the download server, enter the following commands [Figure 9-16]:

```
VMS> cc ets$configd
VMS> link ets$configd,sys$input/opt
sys$share:vaxcrtl.exe/share
<ctrl-z>
VMS> copy ets$configd.exe sys$startup
```

**Figure 9-16: Compiling Download Server**

Then start the ets$configd server. The format for the command looks like the following [Figure 9-17]:

```
VMS> @SYS$STARTUP:ETS$STARTUP [cmd] [dir] [dev] [log]
```

**Figure 9-17: Starting “ets$configd” Server**

T he parameters are as follows:

- ♦ *cmd* is either Start, Show, or Stop. Start will create the server process, Show will show the process's current status, and Stop will kill the server process.
- ♦ *dir* is the directory that download files will be loaded out of.
- ♦ *dev* is the lat device (if different from the default LTA9777).
- ♦ *log* is the name of the file to write status data from the server.

The parameters must be specified in order, and cannot be skipped unless you are ending the command line. For example [Figure 9-18]:

```
VMS> @SYS$STARTUP:ETS$STARTUP START –
VMS> _DUAO:[ETS] LTA5000 SYS$MANAGER:ETSLOG.DAT
```

**Figure 9-18: Syntax Example**

You can use the VMS command **Show System** to ensure the download process started properly. To shutdown the server process [Figure 9-19]:

```
VMS> @SYS$STARTUP:ETS$STARTUP STOP
```

**Figure 9-19: Shutting Down Server Process**

Managers can use the **Source** command on the ETS to attempt to download a configuration file and test the functionality of the download server before actually using the download file at boot time. See the Source command for more details.

The ETS must be able to find the VMS host to utilize the download process. If the node limit on the ETS is set to zero, it may not be able to locate the VMS host. For downloading to work reliably, one of two cases must be true for the ETS to access the VMS configd process:

- ♦ The ETS must have a non-zero node limit configured - see Set Server Node Limit.
- ♦ The VMS must be explicitly told to reply to the ETS if a request is sent. Use the **Set Node** command from within LATCP to set this [Figure 9-20]:

```
LATCP> SET NODE /CONNECTIONS=BOTH
```

**Figure 9-20: Set Node Command**

If neither of these two condition is met, downloading may fail.

# 10 Command Reference

This chapter describes the ETS/EPS command set. Each command and its syntax is shown, as well as whether or not the command is restricted to the privileged user. (See *Set Privileged* on page 10-47 for information on changing your port to privileged status.)

ETS commands and keywords are displayed in **bold** for clarity, but may be entered in upper, lower, or mixed case. Optional parameters are enclosed in brackets **[ ]**. Curly braces **{ }** indicate that one and only one of the items enclosed within the braces must be used. User-supplied parameters, such as a particular port ***number*** or host ***name***, are shown in italics.

*Refer to the Index for a listing of every command keyword explained in this manual.*

The EPS1 and EPS2 do not support the full ETS/EPS commands set; commands that are **not** supported on the EPS1 are marked with an asterisk ( * ).

*See **Set Port Command Completion** for details.*

Command completion is inactive by default. When enabled, pressing the space bar will complete a partially-typed command for you (often saving time and reducing errors.)

Table 10-1 on page 10-2 lists all special keys used for editing the command line.

# Command Line Editing

The following table lists the special keys used for command line editing. Note that line editing is disabled on Hardcopy ports.

*Overstrike is on by default.*

| Key | Purpose |
|---|---|
| Return | Executes the current command line |
| Delete | Deletes the character before the cursor |
| Ctrl-A | Toggles insert mode (insert or overstrike.) |
| Ctrl-D | Logs out of the server |
| Ctrl-E | Moves cursor to end of line |
| Ctrl-H or Backspace | Moves cursor to the beginning of the line |
| Ctrl-R | Redisplays the current command |
| Ctrl-U | Deletes the entire current line |
| Ctrl-Z | Logs out of the server |
| Left Arrow | Moves cursor left |
| Right Arrow | Moves cursor right |
| Up Arrow or Ctrl-P | Recalls the previous command |
| Down Arrow or Ctrl-N | Recalls the next command |
| !*text* | Recalls command starting with *text* |
| !! | Recalls last command |

**Table 10-1: Command Line Editing Keys**

# Attach Port (*)

```
                ┌service [NODE node] [DESTINATION port] [ENVIRONMENT envstring]┐
ATTACH PORT n  <                      TELNET hostname                          >
                └                     RLOGIN hostname                          ┘
```

Forces a connection from a port on the ETS to a LAT or Telnet service. Attach is similar to the Connect command, except that Connect works with the port your terminal is attached to; Attach works with a different port. If the *service* or *hostname* is not given, the dedicated service or hostname is used. Logged-in, virtual, and multisession ports cannot be Attached. Note that environment strings can be used with the Attach command.

**Restrictions** This command requires privileged status.

**Examples** `Local> ATTACH PORT 2 fileserver NODE opus DES port_2`

**See Also** Connect, page 10-9, Set/Define Port Dedicated, page 10-31.

# Backwards (*)

BACKWARDS

Switches the current session to be the session started previous to this one. The session list is shown with the Show Sessions command. Repeating this command will cycle you “backward” (in time) through the active sessions. It is similar to the Forward command, but cycles in the opposite direction. If there is only one active session, the command has no effect; if the beginning of the session list (the earliest active session) is reached, the most recent session (the other end of the list) is resumed.

**See Also** Set/Define Backward Switch, page 10-28, Session Management, page 4-9.

# Bg (*)

BG

Bg is functionally equivalent to the Backwards command (see above). It is provided for Unix compatibility.

# Broadcast (*)

$$\text{BROADCAST}\begin{Bmatrix}\text{ALL}\\ \text{PORTS } \textit{number}\\ \textit{username}\end{Bmatrix}\textit{message}$$

Sends a message to another port, all ports, or a specific user on the server. Broadcast can only be used if broadcasts have been enabled on the server. The *message* may be one word, or several words enclosed in quotes. If quotes are used, the message will be sent exactly as typed, otherwise it will be sent in uppercase.

**Restrictions** The All parameter requires privileged status.

**Errors**
The port broadcasted to is flow controlled.
The server does not have broadcast enabled.
The session broadcasted to is set to Pasthru or Passall.
The sender is notified if a message was not received.

**Examples**

```
Local> BROADCAST PORT 7 “Ready for lunch??”
Local> BROADCAST Fred “Meeting in 10 minutes.”
```

**See Also** Set/Define Server Broadcast, page 10-53, LAT and Services, page 2-2.

# Clear/Purge Dialback (*)

$$\begin{Bmatrix}\text{CLEAR}\\ \text{PURGE}\end{Bmatrix}\text{ DIALBACK }\begin{Bmatrix}\text{ALL}\\ \textit{username}\end{Bmatrix}$$

Clears a dialback setting for a particular username, or for all usernames if the All parameter is specified.

**Restrictions** This command requires privileged status.

**Errors** Clear Dialback will return an error if the specified username isn’t found, or if All is specified and no entries are configured.

**Examples**

```
Local> PURGE DIALBACK robert
Local> CLEAR DIALBACK ALL
```

**See Also** Set/Define Dialback, page 10-21, Show/List Dialback, page 10-85.

# Clear/Purge Hosts

See Clear/Purge Telnet Hosts, page 10-8.

# Clear/Purge Ipsecurity

$\left\{\begin{matrix}\text{CLEAR}\\ \text{PURGE}\end{matrix}\right\}$ IPSECURITY $\left\{\begin{matrix}\text{ALL}\\ \textit{IPaddress}\end{matrix}\right\}$

Removes an entry for a particular IP address from the IP security table (or all the entries if the All parameter is specified).

**Restrictions** This command requires privileged status.

**Errors** Clear IPsecurity will return an error if the specified entry isn't found, or if All is specified and no IP security entries have been configured.

**Examples**

```
Local> PURGE IPSECURITY 192.0.1.255
Local> CLEAR IPSECURITY ALL
```

**See Also** Set/Define Ipsecurity, page 10-22, Show/Monitor/List Ipsecurity, page 10-85.

# Clear/Purge Menu

$\left\{\begin{matrix}\text{CLEAR}\\ \text{PURGE}\end{matrix}\right\}$ MENU $\left\{\begin{matrix}\text{ALL}\\ \textit{MenuNum}\end{matrix}\right\}$

Removes the specified menu entry (or all the entries if the All parameter is specified).

**Restrictions** This command requires privileged status.

**Examples**

```
Local> CLEAR MENU ALL
Local> CLEAR MENU 2
```

# Clear/Purge Protocol Netware Access

{CLEAR<br>PURGE} PROTOCOL NETWARE ACCESS {ALL<br>*fileserver*}

Clears one or more of the currently specified entries in the NetWare access list. Specify either All or a fileserver name, and that entry will be removed from the access list.

**Restrictions** This command requires privileged status.

**Errors** Clear Protocol will return an error if the specified entry isn't found, or if ALL is specified and no access list entries have been configured.

**Examples**

```
Local> PURGE PROTO NET ACC ALL
Local> CLEAR PROTO NET ACC LAB_FS4
```

**See Also** Set/Define Protocol Netware Access, page 10-49, Show/List Protocol Netware, page 10-89.

# Clear/Purge Service

{CLEAR<br>PURGE} SERVICE {LOCAL<br>*service*}

Removes the definition of a service from the ETS. Clear Service disables the service immediately, but it may return when the ETS is reinitialized if the service was previously Defined. Purge Service removes a service permanently but not until the ETS is reinitialized.

Note that the Login service can not be deleted via the Clear/Purge commands - you must use the Set/Define Server Incoming command to remove them.

**Restrictions** This command requires privileged status.

**Parameters** **Local**
Local removes the definitions of all local services.

**Errors** Clear Service fails when there are sessions connected to the service or when there are connect requests in the service's queue. These conditions can be corrected with the Logout Port and Remove Queue commands.

**Examples**

```
Local> PURGE SERVICE LOCAL
Local> CLEAR SERVICE fileserver
```

**See Also** Set/Define Service, page 10-74, Show/List Services, page 10-91.

# Clear/Purge SLIP (*)

Clear SLIP is not a valid command. Purge Slip is detailed under Purge, on page 10-16.

# Clear/Purge SNMP

$$\left\{\begin{matrix}\text{CLEAR}\\\text{PURGE}\end{matrix}\right\}\ \text{SNMP}\ \left\{\begin{matrix}\text{ALL}\\\textit{CommunityName}\end{matrix}\right\}$$

Used to remove entries from the SNMP security table. The parameter is either the name of a community name in the table or ALL to remove all entries in the table.

**Restrictions** This command requires privileged status.

**Parameters** **All**
Removes all SNMP table entries.

**Example**

```
Local> CLEAR SNMP public
```

**See Also** Set/Define SNMP, page 10-84, Show/List SNMP, page 10-92.

# Clear/Purge Telnet Hosts

$$\begin{Bmatrix}\text{CLEAR}\\ \text{PURGE}\end{Bmatrix} \text{[TELNET] HOSTS} \begin{Bmatrix}\text{LOCAL}\\ \textit{HostName}\end{Bmatrix}$$

Removes a TCP/IP host entry from the ETS' table of known hosts. Clear Telnet Hosts removes the entry immediately, but not permanently. If the host was seen through the **rwho** facility, it will reappear as soon as that machine broadcasts again. A host will Also reappear if a user Connects to it. Purge Telnet Hosts removes the entry from the permanent database, but it remains in the active list until the server is rebooted.

**Restrictions** This command requires privileged status.

**Parameters** **Local**
Removes the names of all known hosts.

**Errors** Clear Telnet Hosts will fail if there are any active Telnet connections on the server.

**Example**

```
Local> CLEAR HOSTS alex
```

**See Also** Set/Define Telnet Hosts, page 10-84, Show/List Hosts, page 10-85.

# Close (*)

CLOSE

Close is functionally equivalent to the Disconnect command, discussed on page 10-10.

# Cls

CLS

Clears the screen on your terminal device if the port is configured as Type ANSI.

**See Also** Set/Define Port Type, page 10-45.

# Connect (*)

```
CONNECT  [LAT] service [NODE nodename] [DESTINATION port] [ENVIRONMENT EnvString]
         @@PortName
         TELNET [hostname]
         RLOGIN [hostname]
         TCP [hostname]
```

Establishes a session with a LAT service or TCP/IP host. If no service or host-name is specified, a connection to any *preferred* service or host is attempted. In the case where more than one node or port offers the same LAT service, a particular node and/or port name can be specified with the Node and Destination parameters. Otherwise, the node offering the service with the highest rating will be accessed. Telnet, TCP and Rlogin connections have no such parameters; the host-name in this case can also be a numeric IP address.

The LAT option is provided for cases where a LAT service happens to be named Telnet or Rlogin, etc. Otherwise, the service name would be used as a command keyword instead of the target service.

The *@@port_name* option is a shorthand method for connecting to a local port without having to create a service. It works only for local port connections.

A session environment string can be added to the connect request (refer to *Environment for TCP/IP and LAT* on page 2-21). A colon and port number can Also be added to the hostname for TCP/Telnet/Rlogin sessions; in this case the specified port number will be used for the connection. The TCP option creates a raw TCP connection to the host (i.e. no Telnet data interpretation is performed).

**Examples**

```
Local> CONNECT
Local> CONNECT @@Port_3
Local> CONNECT vax6 ENVIRON +e
Local> CONNECT wampum NODE opus DESTINATION port_2
Local> CONNECT RLOGIN 145.34.35.14
Local> CONNECT TELNET 145.34.35.11:245
Local> CONNECT TCP labsun
```

**See Also** Session Management, page 4-9, Set/Define Port Preferred, page 10-41.

# Crash 451

```
CRASH 451
```

Crash 451 **immediately** reboots the server. All users are logged off and sessions are disconnected. If upline dumping is enabled the server will generate a dump file on the load host before rebooting.

**Restrictions** This command requires privileged status.

# Define

Define is equivalent to Set, but Define changes the *permanent* characteristics of ports, servers, and services. Define Port and Define SLIP settings do not take effect until after the current user logs out. Define Server, Define Telnet Host, and Define Service settings remain unchanged until the ETS is rebooted. To make a permanent change that takes effect immediately requires both a Define and Set command with the same parameters. All Define commands are documented together with their corresponding Set commands.

# Disconnect (*)

```
DISCONNECT
```

Terminates the *current* session (if no session is specified), the *specified* session, or *all* your sessions.

**Examples**

```
Local> DISCONNECT
Local> DISCONNECT SESSION 3
```

**See Also** Session Management, page 4-9.

# Fg (*)

```
FG
```

Fg is functionally equivalent to the Forwards command, discussed on page 10-11. It is provided for Unix compatibility.

# Finger

FINGER [ [*username*] [@*host*] ]

This command is an implementation of the UNIX Finger command that shows local and remote users. The finger command by itself will show the ports on the ETS. If a ***username*** is specified, information about user ***username*** on the ETS is shown. If the finger command is given with a parameter such as ***user@host***, information regarding user ***user*** on TCP/IP host ***host*** will be shown. The username can be omitted, in which case all the users on the host will be displayed.

To see a list of processes running on the ETS, use the command "finger finger".

**Errors** An error is displayed if the host cannot be accessed.

**Examples**

```
Local> FINGER (shows all users)
Local> FINGER bob(shows user bob on ETS)
Local> FINGER @hydra(shows users on host hydra
Local> FINGER bob@hydra(shows user bob on hydra)
Local> FINGER FINGER (Shows ETS process list)
```

**See Also** Show Users, page 10-93.

# Forwards (*)

FORWARDS

Cycles forward through your sessions in the order displayed by the Show Sessions command. The next session on the list becomes the active session. Forward has no effect if there is only one session; if the end of the session list (most recently started session) is reached, the oldest session (at the other end of the session list) is resumed.

**Errors** An error is displayed if no sessions are active.

**See Also** Set/Define Port Forward Switch, page 10-35, Sessions, page 2-19, Multiple Session Support, page 1-5.

# Help

HELP [*command* [*parameter*] ]

Lets you access the online Help system. **Help** with no parameters lists all available commands. Specifying **Help *command*** gives information about that command and a list of its parameters. **Help** with a command and a parameter gives information about the parameter and possibly a list of sub-parameters, etc.

The help processor only shows the help text for the commands that the user is privileged to use. To see all help text, become the privileged user.

**Examples**

```
Local> HELP
Local> HELP CONNECT
Local> HELP DEFINE SERVER BROADCAST
```

# Initialize Server

INITIALIZE [SERVER] [CANCEL | NOBOOT | RELOAD | FACTORY] [DELAY *delay*]

Resets the server after an optional delay, or cancels a pending Initialize. When reset, the server loses all changes made using Set commands unless corresponding Defines or Saves were also made.

The Delay option allows you to schedule the Initialize after a specified number of minutes. (Show Server Status will show the time pending until a scheduled reboot.)

The Cancel parameter cancels any pending Initialize command. Note that Cancel and Delay cannot be used together.

The remaining 3 options control after-boot behavior. The Noboot option will force the server to remain in the Boot Configuration Program instead of booting; this option is only supported on ROM versions 2.0 or later. The Reload option will, on Flash-ROM equipped servers, force the server to attempt to redownload the operational code and reprogram the Flash-ROMs. The Factory command will cause the server to reload the factory settings; all Defined and Saved settings will be cleared and will have to be re-configured.

**Restrictions** This command requires privileged status.

# Initialize Server (cont.)

**Parameters** **Delay**

Set the number of minutes to delay the reboot. Range is 0 - 120 minutes.

**Examples**

```
Local> INITIALIZE DELAY 12
Local> INITIALIZE RELOAD DELAY 12
Local> INITIALIZE FACTORY
Local> INITIALIZE CANCEL
```

# Jobs (*)

The jobs command is functionally identical to the Show Sessions command, discussed on page 10-92. It is provided for UNIX compatibility.

# Kill (*)

The Kill command is functionally identical to the Disconnect command, discussed on page 10-10. It is provided for UNIX compatibility.

# Lat (*)

LAT is a shorthand for the Connect LAT command, and is provided for simplicity. If a preferred service has been defined, the service parameter is optional. An environment string can also be added to the command - see *Environment for TCP/IP and LAT* on page 2-21 for details.

**See Also** Connect, page 10-9, Set Port Preferred, page 10-41.

# List

List is similar to the Show command, except that List displays Defined (permanent) characteristics, which may or may not be the same as those currently Set (temporary). In other words, List shows settings that will take effect the next time the ETS is reinitialized. Each List command is documented together with its corresponding Show command.

# Lock (*)

LOCK

Locks a port without disconnecting sessions. The user is queried for a password (6 characters maximum, alphanumeric only) and asked to verify it. The password and verification are not displayed as the user types them. The port is then locked until the correct password is used to unlock it. If a user forgets his password, the privileged user must either Logout the port (disconnecting all sessions) or use the Unlock Port command.

**Restrictions** Secure users may not lock their ports.

**Examples**

```
Local> LOCK
Password> donut(the passwords will not be echoed)
Verification> donut
Unlock password> donut
Local>
```

**See Also** Set Server Lock, page 10-59, Unlock Port, page 10-97.

# Logout

LOGOUT [PORT *PortNum*]

Logs out the current port by default, or another port if specified. Disconnects all of the port's open sessions. Closes any SLIP connections, if SLIP was enabled.

**Restrictions** Only privileged users can log out a port other than their own.

**Examples**

```
Local> LOGOUT
Local> LOGOUT PORT 4
```

# Man

The Man command is functionally identical to the Help command, discussed on page 10-12. It is provided for Unix compatibility.

# Mode

MODE [COM *SerPort*:] *baudrate* [,*parity* [,*charsize* [,*stopbits*] ] ]

The Mode command is provided for DOS compatibility, and configures the serial port parameters. Note that Mode does both a Set and Define of the parameters.

**Parameters**

**serport**
Configure this serial port instead of the current port

**baudrate**
One of the settings shown under Set Port Speed

**parity**
One of the settings shown under Set Port Parity

**charsize**
7 or 8

**stopbits**
1 or 2 (EPS1/EPS2 only)

**See Also** Set/Define Port, page 10-25.

# Monitor

Monitor commands are the same as Show commands, except that the displayed information is continuously updated approximately every 3 seconds. The display can be stopped by pressing any key. See the corresponding Show commands for details and options.

### Restrictions

This command requires privileged status.

# Netstat

NETSTAT

The Netstat command shows the currently active network connections. Information is displayed for the AppleTalk, LAT, LanManager, Netware and TCP/IP protocols. This information is primarily meant for debugging network problems.

# Open (*)

The Open command is functionally identical to the Connect Telnet command, discussed on page 10-9.

# Ping

PING *hostname*

Ping sends a request (using TCP/IP) for an echo packet to another network host. This provides an easy way to test network connections to other TCP/IP hosts. In general, any host that supports TCP/IP will respond if it is able, regardless of login restrictions, job load, or operating system. If there is no reply from the host, this may indicate a network or TCP/IP configuration problem. Note that either a text hostname or IP address can be provided.

**Parameters** ***hostname***
Name or IP address of the network host

**Examples**

```
Local> PING 192.0.1.23
Local> PING hydra.local.net
```

# Purge

Purge is comparable to Clear, but Purge removes an entry (service or IP host) from the server's permanent database. Note that Purge SLIP is the only exception, and is shown below. Purge does not affect the current operating characteristics.

# Purge Slip (*)

```
             ALL
PURGE SLIP { PORT portnum }
             DEFAULT
```

Removes a previously configured Define SLIP entry for a port. Note that Clear SLIP is not a valid command.

Using the keyword parameter returns the server wide slip settings back to their default values.

**Restrictions** This command requires privileged status.

**Errors** Purge SLIP will return an error if the specified port does not have SLIP enabled, or if the IP address specified is not associated with that port.

**Example** `Local> PURGE SLIP Port 4`

**See Also** Show/List Slip, page 10-92, SLIP, page 2-9, Server Management, page 10-17.

# Remove Queue

REMOVE QUEUE { ENTRY *number* | NODE *name* | SERVICE *name* | ALL }

This command removes requests for local services from that service's queue. A particular request or all requests may be specified. The Node parameter removes requests for that node. The Service parameter removes all requests queued for that particular service. Use the Show Queue command to display the queues and entries therein.

**Restrictions** This command requires privileged status.

**Examples**

```
Local> REMOVE QUEUE NODE opus
Local> REMOVE QUEUE ENTRY 5
Local> REMOVE QUEUE SERVICE MODEM
Local> REMOVE QUEUE ALL
```

**See Also** Show/Monitor Queue, page 10-90.

# Resume (*)

| |
|---|
| RESUME [ [SESSION] *number*] |

Leaves local mode and resumes the current session (the active session). To resume a session other than the current one, specify a session number .

**Errors** An error is returned if there are no active sessions.

**Examples**

```
Local> RESUME
Local> RESUME SESSION 4
```

# Rlogin (*)

| |
|---|
| RLOGIN [*hostname* [*username*] ]<br>CONNECT RLOGIN [*hostname* [*username*] ] |

Rlogin is a shorthand for the Connect Rlogin command, and is provided for simplicity. The hostname can be either text or an IP address. If a preferred Telnet service has been defined, the hostname parameter is optional. If the optional *user_name* is specified, it will be used as the login name.

**See Also** Connect, page 10-9, Set Port Preferred, page 10-41.

# Save

$$\text{SAVE}\left\{\begin{array}{c} \text{IPSECURITY}\\ \text{MENU}\\ \text{PORT}\begin{bmatrix} \textit{portnum}\\ \text{ALL}\end{bmatrix}\\ \text{PRINTER}\\ \text{SERVICE}\left\{\begin{array}{c}\textit{name}\\ \text{ALL}\end{array}\right\}\\ \text{SERVER}\\ \text{SNMP}\end{array}\right\}$$

This command saves current configurations into the permanent database, essentially Define-ing everything you have Set so far. Instead of issuing a Define for each Set command in order to make your changes permanent, you can just issue the Save command after you have configured the port, service, server or printer.

Note that you cannot Save the login service - you must use Define Server Incoming to make it permanent.

**Restrictions** This command requires privileged status.

**Parameters**

***number***
Optional. If Port parameter is used and number is omitted, the current port is Saved.

**All**
The settings for all the ports (or services) are Saved.

***name***
All Set Service commands for the specified service become Defined.

**IPsecurity**
The current IPsecurity table is Saved.

**Menu**
Saves all menu items set up with Set Menu commands.

**Server**
The current Server settings are Saved.

**PRINTER**
(EPS only) The current printer port settings are Saved.

**SNMP**
Saves all parameters associated with SNMP.

**Errors** Save without a parameter is invalid.

# Send (*)

SEND *command*

The Send command is used to send Telnet and LAT commands through a session. This command is nonfunctional for non-Telnet TCP connections (Rlogin, raw TCP, etc).

### Parameters

Command is one of the following:

| | | |
|---|---|---|
| AO | Abort Output | (Telnet, LAT) |
| AYT | Are You There | (Telnet, LAT) |
| BRK | Break | (Telnet, LAT) |
| EC | Erase Character | (Telnet) |
| EL | Erase Line | (Telnet) |
| GA | Go Ahead | (Telnet) |
| IP | Interrupt Process | (Telnet, LAT) |
| NOP | No Operation | (Telnet) |
| SYNCH | Synchronize | (Telnet) |

### Examples

```
Local>   SEND AYT
Local>   SEND SYNCH
```

# Set/Define Dialback (*)

{SET / DEFINE} DIALBACK *username command*

Configures the (username, command) strings for the dialback handler. Each username is specified with a command string that is echoed to the serial port after that user logs in. If a modem is connected to the port, a user can dial into the ETS, enter his username, and then have the ETS log him out and call him back at a particular phone number, for example.

**Restrictions** This command requires privileged status.

**Parameters**

***username***
A text name, up to 16 characters long.

***command***
A string of text, up to 30 characters long.

**Example**

```
Local> SET DIALBACK "robert" "atdt555-1234"
```

**See Also** Set/Define Port Dialback, page 10-21, Clear/Purge Ipsecurity, page 10-5.

# Set/Define Hosts

See Set/Define Telnet Hosts, page 10-84.

# Set/Define Ipsecurity

{SET / DEFINE} IPSECURITY [ADDRESS] *IPaddress* [*Parameters*]

This command is used to add or change entries in the IP security table. Any address with a 0 in any segment implies Incoming and Outgoing Disabled for all ports. An address with a 255 in any segment means the restriction applies to all the addresses in that range. A port number of 0 is used to apply to the virtual (incoming login) ports.

**Restrictions** This command requires privileged status.

**Parameters**

***IPaddress***
The IP address to be restricted. The address can be a full IP address, such as 192.0.1.80, to restrict one address; it can also be expressed as a partial address, such as 192.0.1.255, to restrict whole sub-networks.

**Incoming**
Restricts logins from the network into the server.

**Outgoing**
Restricts telnet sessions to the network from the server.

**Slip**
Restricts SLIP sessions from the server.

**Both**
Restricts both logins from the network to the server and telnet sessions to the network from the server.

**Ports** ***portlist***
An optional list of ports for which the restriction applies. If portlist is not specified, all physical and virtual ports apply.

**Examples**

```
Local> SET IPSEC ADDR 192.0.1.255 INC ENA OUT DIS
Local> SET IPSEC 134.0.1.255 Port 3,5-7
```

**See Also** Show/Monitor/List Ipsecurity, page 10-85, Clear/Purge Ipsecurity, page 10-5.

# Set/Define Logging

```
{ SET    } LOGGING { HOST hostname }
{ DEFINE }         { NONE          }

                   [ COMMANDS ]
                   [ MODEM    ]
{ SET    } LOGGING { NETWORK  } { ENABLED  }
{ DEFINE }         { PRINTER  } { DISABLED }
                   [ SECURITY ]
                   [ SYSTEM   ]
```

This command is used to control error and event logging on the ETS. Events can be logged to a network host (via LAT, TCP/IP, or NetWare) or to the console of the ETS. See chapter 4 for a detail description of the events that can be logged.

**Restrictions** This command requires privileged status.

**Parameters**

***hostname***

Specifies the host to send event logs to. Specified as

| | |
|---|---|
| hostname:: | for LAT hosts |
| hostname: | for TCP/IP hosts |
| hostname | for NetWare fileservers |
| CONSOLE | to send events to the ETS console. |

**None**

Clears any enabled event logs and the log host name.

**See Also** Show/Monitor/List Logging, page 10-86, *Configuring Event Logging*, page 3-14.

# Set/Define Menu (*)

{ SET / DEFINE } MENU { *ItemNum StringCommand* / TITLE *TitleString* }

Configures the menu and command strings for the menu handler. Each menu string is specified with a command string that is executed if the user selects that menu entry. Individual ports are placed in menu mode the Define Port Menu command.

**Restrictions** This command requires privileged status.

**Parameters**

**itemnum**
A number (1 through 32) and corresponds to the menu entry you are changing.

**string**
A text string, up to 32 characters long, that is shown to the users in the menu screen.

**command**
A string of text, up to 32 characters long, that is the actual command executed when the user selects this entry.

**title_string**
An optional title for the entire menu, up to 48 characters long.

**Example**

```
Local> Set MENU 5 “show net nodes” “SHOW HOSTS”
```

**See Also** Menu Mode, page 2-24, Show/List Menu, page 10-86, Clear/Purge Menu, page 10-5.

# Set Noprivileged

See Set Privileged, page 10-47.

# Set/Define Port Access

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{ACCESS}\left\{\begin{matrix}\text{DYNAMIC}\\ \text{LOCAL}\\ \text{NONE}\\ \text{REMOTE}\end{matrix}\right\}$$

Sets the type of connections allowed to this port. Autobaud must be disabled for Remote and Dynamic ports.

**Parameters**

**Local**
The default. Port may originate connections to services, and only local logins are permitted.

**Remote**
Port accepts network connection requests. No local logins are permitted.

**Dynamic**
Port can originate and receive connection requests.

**None**
Port is unusable.

**Restrictions** This command requires privileged status.

**Errors** If a port is active, its access cannot be set.

**Example**

```
Local> DEFINE PORT ACCESS LOCAL
```

# Set/Define Port Authorized Groups

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{AUTHORIZED [GROUPS]}\left\{\begin{matrix}\text{ALL}\\\textit{grouplist}\end{matrix}\right\}\begin{bmatrix}\text{ENABLED}\\\text{DISABLED}\end{bmatrix}$$

Determines which service groups are accessible to a port. Use service groups to restrict access to services on a per port basis. The authorized groups are the only groups a port can see. Use Set Port Groups to determine which of the Authorized groups are actually visible.

**Parameters** ***grouplist***
Group numbers range from 0 to 255. They can be listed individually (for example 1, 3, 12) and by range (for example 3-25), separated by a commas. The default is group 0 enabled, 1-255 disabled.

**Restrictions** This command requires privileged status.

**Example** `Local> SET PORT AUTHORIZED 0,2-5,101`

**See Also** Set/Define Server Groups, page 10-55, Set Port Groups, page 10-35.

# Set/Define Port Autobaud (*)

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{AUTOBAUD}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Allows a port to detect the incoming baud rate and change its own to match at login time. When autobaud is set, you may have to press Return twice or more to allow the port to determine the baud rate. Autobaud must be disabled for Remote and Dynamic port access and for any port that offers a service.

**Restrictions** This command requires privileged status.

**Errors** Autobaud will not work for all parity and baud combinations - it will only function if the port is using either 8-bit characters with no parity, or 7-bit characters with even parity. Autobaud will only function for baud rates from 1200 through 38400.

**Example** `Local> DEF PORT AUTOBAUD DISABLED`

# Set/Define Port Autoconnect (*)

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix}\right]\text{AUTOCONNECT}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

If enabled, the port connects automatically to the preferred service (LAT or Telnet) upon login. You can still Break to local mode and use the Connect command to attach to other services. Autoconnect is disabled if there is no preferred service defined. Also, if both preferred LAT and preferred Telnet services have been set, the LAT connection will be attempted.

**Example** `Local> SET PORT AUTOCONNECT ENABLED`

**See Also** Set/Define Port Preferred, page 10-41.

# Set/Define Port Autoprompt (*)

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix}\right]\text{AUTOPROMPT}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

If enabled, the ETS will automatically display a login prompt on connection to a LAT login service. If disabled, the user must press the <Return> key first. This option is not applicable to TCP/IP connections, which should always display a login prompt upon connection.

**Example** `Local> SET PORT AUTOPROMPT DISABLED`

# Set/Define Port Backward Switch (*)

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{BACKWARD [SWITCH] }\left\{\begin{matrix}\textit{character}\\ \text{NONE}\end{matrix}\right\}$$

Allows the user to define a key which switches to the previous session without entering local mode. From local mode, typing the key functions as if the Backward command was entered. Any key can be specified, unless it conflicts with the line editing, <Break> or <Forward> keys. You should not use a key that will be used by a remote operating system or application you will be using, as the ETS will interpret that key and the remote service will not see it. Also, line editing characters will be interpreted by the ETS. Control characters can be specified as a character preceded by a caret('^').

**Parameters** ***character***

To specify a control character, hold down the control key while typing the letter, or use the caret (see previous paragraph.)

**Example**

```
Local> SET PORT BACKWARD SWITCH  ^K
```

**See Also** Backwards, page 10-3, *Sessions*, page 2-19.

# Set/Define Port Bitronics (*)

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{BITRONICS }\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

Enables or disables bidirectional parallel port functionality. Valid only on the EPS1 and EPS2. Note that the attached printer must also support Bitronics mode.

**Example**

```
Local> DEFINE PORT 2 BITRONICS ENABLED
```

# Set/Define Port Break (*)

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix}\right]\text{BREAK}\left\{\begin{matrix}\text{LOCAL}\\ \text{REMOTE}\\ \text{DISABLED}\end{matrix}\right\}$$

Determines where processing of the Break key will take place.

**Parameters**

**Local**
The Break key returns you to local mode, i.e. the key is processed by the ETS.

**Remote**
The Break key is ignored by the ETS and passed through to the remote service.

**Disabled**
The Break key does nothing.

**Example**

```
Local>  SET PORT BREAK REMOTE
```

**See Also** Set/Define Port Local Switch, page 10-37.

# Set/Define Port Broadcast

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix}\right]\text{BROADCAST}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

Enables or disables other users' broadcasts to this port. This would typically be set whenever extra messages are not desired on the port's output device.

**Example**

```
Local>  SET PORT BROADCAST DISABLED
```

**See Also** Broadcast, page 10-4, Set/Define Server Broadcast, page 10-53.

# Set/Define Port Character Size

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{port}\\\text{ALL}\end{matrix}\right]\right]\text{CHARACTER [SIZE]}\left\{\begin{matrix}7\\8\end{matrix}\right\}$$

Sets the number of bits per character for the serial port. Character size defaults to 8 bits.

**Errors** Autobaud will only work for 8 bits, or for 7 bits with Even parity.

**Example** `Local>  SET PORT CHARACTER SIZE 7`

**See Also** Set/Define Port Autobaud, page 10-26, Set/Define Port Parity, page 10-40.

# Set/Define Port Command Completion

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\\text{ALL}\end{matrix}\right]\right]\text{COMMAND [COMPLETION]}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Enables or disables the command completion feature. If enabled, the ETS will attempt to complete partially typed command words when the user presses the Space or Tab keys. If the command to that point is ambiguous (or if the user is typing in an option), the ETS will send a beep to the terminal.

**Example** `Local>  SET PORT COMMAND ENABLED`

# Set/Define Port Dedicated (*)

$$\text{DEFINE}\left[\text{PORT}\left[\begin{matrix}\textit{portnum}\\ \text{ALL}\end{matrix}\right]\right]\text{DEDICATED}\left\{\begin{matrix}\text{NONE}\\ [\text{SERVICE } \textit{service}]\\ \text{TELNET } \textit{hostname}\\ \text{RLOGIN } \textit{hostname}\end{matrix}\right\}\left[\begin{matrix}\text{NODE } \textit{node}\\ \text{ENVIRONMENT}\end{matrix}\right][\text{DESTINATION } \textit{port}]$$

Dedicates a port to a service. Once logged into the port, you are automatically connected to the service. You cannot return to local mode, so the user never issues server commands. When you exit the remote service, you are logged out of the ETS. You can dedicate a port to a Telnet host as well. Setting any dedicated service or host will clear any Preferred services and hosts, however, Set Dedicated has no effect; Dedicated must be Defined. An environment string can be part of the dedicated hostname; see the "Environment" section of the Concepts chapter for more information. Note that via the environment string, a dedicated host can be connected to via Rlogin, not just Telnet.

**WARNING:** Defining dedicated on all the server's ports is dangerous, as it leaves no easy way to log into the server. You must either connect via the NCP or Telnet console ports (see Chapter 2, *Concepts*), or have incoming logins enabled (see Set Server Incoming, page 10-58).

**Restrictions** This command requires privileged status.

**Parameters**

**None**
Clears any existing Dedicated service.

**service**
The port is dedicated to this service.

**Telnet *hostname***
The port is dedicated to this Telnet host.

**Rlogin *hostname***
A synonym for Telnet hostname. Ports set up to use Rlogin will still use Telnet for the connection.

**Errors** If the port is logged in, you can only issue this command as a Define.

**Examples**

```
Local> DEFINE PORT DEDICATED Fileserver NODE opus
       DESTINATION port_4

Local> DEFINE PORT 5 DEDICATED TELNET 192.0.1.221
```

**See Also** Set/Define Port Preferred, page 10-41.

# Set/Define Port Dialback (*)

$$\begin{Bmatrix}\text{SET}\\\text{DEFINE}\end{Bmatrix}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{DIALBACK}\begin{Bmatrix}\text{ENABLED}\\\text{DISABLED}\end{Bmatrix}$$

Turning on Dialback causes the ETS to check the dialback table (see the Set Dialback command) each time a user logs in. If that username is not in the table, the port is logged out. If it is in the table the port is logged out and the ETS sends the dialback string (typically modem commands) to the port and awaits a second login. Typically the dialback string will cause the a modem attached to the port to call the user back and a certain telephone number for security reasons. You will probably want to use Define Port Modem Control as well. Note that ports with dialback enabled have a 30-second time limit for entering the username when logging in (for security reasons).

**Restrictions** This command requires privileged status.

**See Also** Clear/Purge Dialback, page 10-4, Show/List Dialback, page 10-85, Set/Define Port Modem Control,page 10-38.

# Set/Define Port Dialup (*)

$$\begin{Bmatrix}\text{SET}\\\text{DEFINE}\end{Bmatrix}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{DSRLOGOUT}\begin{Bmatrix}\text{ENABLED}\\\text{DISABLED}\end{Bmatrix}$$

Sets the Dialup characteristic for a port. The ETS does not pay attention to this flag, but it is passed on to the host for LAT connections, and the host can use it appropriately.

**Restrictions** This command requires privileged status.

# Set/Define Port Dsrlogout

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\left[\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\right]\text{DSRLOGOUT}\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$$

When enabled, the port will be logged out when the port's DSR signal is dropped. This usually happens only when the attached terminal device is powered off or disconnected, and is intended to keep users from switching terminal lines to access other sessions. Any open connections will be closed before logging out.

**Restrictions** This command requires privileged status.

**Errors** Hardware flow control (RTS/CTS and DTR/DSR) and DSRlogout cannot be active simultaneously on the ETS 8/16 and EPS 4/12. When Modem Control is enabled on the port, DSRlogout is enabled, although it will not show up as such.

**See Also** Set/Define Port Modem Control, page 10-38, Set/Define Port Flow Control, page 10-34.

# Set/Define Port Dtrwait

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\text{DTRWAIT}\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$$

If enabled, the ETS will not assert the DTR signal on the serial port until either a user logs into the port or the port is connected to via a service or Telnet connect. When the port is idle, DTR will not be asserted. The default is to assert DTR at idle (i.e. DTRwait disabled).

**Restrictions** This command requires privileged status.

**Errors** Hardware flow control (RTS/CTS and DTR/DSR) and DTRwait cannot be active simultaneously on the ETS8/16 and EPS4/12. DTRwait also cannot be specified with Modem Control.

**See Also** Set/Define Port Modem Control, page 10-38, Set/Define Port Flow Control, page 10-34.

# Set/Define Port Flow Control

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{FLOW [CONTROL]}\left\{\begin{matrix}\text{NONE}\\ \text{CTS}\\ \text{DSR}\\ \text{XON}\\ \text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

Sets the type of flow control on the port, and whether flow control should be enabled for the port. For example, you can set the type of flow control to DSR/DTR, but have it disabled so flow control is not performed.

**Parameters**

**None**
Perform no flow control

**CTS**
CTS/RTS flow control.

**DSR**
DSR/DTR flow control.

**XON**
XON/XOFF flow control.

**Disabled**
No flow control is enabled.

**Enabled**
Enable flow control.

**Errors**

On the ETS8/16 and EPS4/12, Modem Control and DSRlogout must be disabled if CTS/RTS or DTR/DSR flow control is selected.

**Example**

```
Local>  SET PORT FLOW CONTROL DSR
```

# Set/Define Port Forward Switch (*)

$$\begin{Bmatrix}\text{SET}\\ \text{DEFINE}\end{Bmatrix} \left[\text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix}\right] \text{FORWARD [SWITCH]} \begin{Bmatrix}\textit{character}\\ \text{NONE}\end{Bmatrix}$$

Allows the user to define a key which switches to the next session without entering local mode. Any key can be specified, as long as it does not conflict with the line editing, Local or Backward keys. In local mode, the forward key functions as if the Forward command had been typed. To specify a control character, hold down the control key while typing the letter (or use a caret '^' and the letter). You should not use a key that will be used by a remote operating system or application you will be using. Also, the ETS will interpret line editing control characters before they can be used for switches - use the ^ notation in this case.

**Parameters**

**Character**
Any character.

**Example**

```
Local> SET PORT FORWARD SWITCH ^X
```

**See Also**

Set/Define Port Local Switch, page 10-37, Set/Define Port Backward, page 10-28, Forwards, page 10-11, Sessions, page 2-19.

# Set/Define Port Groups

$$\begin{Bmatrix}\text{SET}\\ \text{DEFINE}\end{Bmatrix} \text{PORT}\begin{bmatrix}\textit{portlist}\\ \text{ALL}\end{bmatrix} \text{GROUPS} \begin{Bmatrix}\textit{grouplist}\\ \text{ALL}\end{Bmatrix} \begin{Bmatrix}\text{ENABLED}\\ \text{DISABLED}\end{Bmatrix}$$

Limits which of the Authorized groups the port has access (i.e., the groups must already be in the Set Port Authorized Groups list). If the Enabled or Disabled modifiers are used, the group list is added to or removed from the current list. Otherwise, the group list given replaces the current one. Groups lists are applicable to LAT connections only.

**Parameters**

**Enabled**
Add the specified groups

**Disabled**
Do not show the specified groups.

**Errors**

This option can only be Set (it is only temporary).

**Example**

```
Local>  SET PORT GROUPS 4,5-9
```

**See Also**

Set/Define Port Authorized Groups, page 10-26.

# Set/Define Port Inactivity Logout

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{INACTIVITY [LOGOUT]}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Enables automatic logout of the port if it has been "inactive" for a set period of time. This timer period is set with the Set Server Inactivity Limit command. Inactive is defined as having no keyboard or network activity on the port. The port's open connections (if any) will be closed before logging out.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Server Inactivity Limit, page 10-57.

# Set/Define Port Input/Output Speed

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\begin{bmatrix}\text{INPUT}\\\text{OUTPUT}\end{bmatrix}\text{SPEED }\textit{speed}$$

Specifies the baud rate of the port. The optional Input and Output qualifiers can be used to set split baud rates; if not specified, the baud rate applies to both. The number of stop bits is set according to the baud rate as follows: 2 for 150, and 1 for all other speeds. Due to hardware restrictions, split speed (input speed not equal to output speed) cannot be run on ports 3 and 4 of the EPS. There are no such restrictions on the ETS units.

**Parameters**

**Input**
The speed of input from the connected device

**Output**
The speed of output from the server to the connected device

**speed**
300, 600, 1200, 2400, 4800, 9600, 19200 or 38400 baud. The EPS1/2 also support 56000, 57600, 64000, and 115200.

**Errors** An error is displayed for illegal baud rates.

**Examples**

```
Local>  SET PORT INPUT SPEED 19200 OUTPUT SPEED 4800
Local>  SET PORT SPEED 2400
```

# Set/Define Port Interrupts

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{INTERRUPTS}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Specifies whether or not to allow a port with Access Dynamic which has a remote connection to be interrupted and changed to the Local state. The Enabled option makes it possible for a local user to take over a port that would otherwise be busy with a remote connection.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Port Access, page 10-26.

# Set/Define Port Local Switch (*)

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{LOCAL [SWITCH]}\left\{\begin{matrix}\textit{character}\\\text{NONE}\end{matrix}\right\}$$

Allows the user to define a key that switches the port to local mode from a remote connection. Any key can be specified, provided it does not conflict with the line editing, Forward and Backward keys. To specify a control character, hold down the control key while typing the letter (or use a caret '^' and the letter). You should not use a key that will be used by a remote operating system or application you will be using. The Local Switch key will function the same as the Break key, except that it *will* be ignored by the ETS if a session is set up as Passall or Passthru. The local switch is especially useful for connections into the ETS, where a local break key may not be passed to the ETS.

**Parameters** ***Character***
The character that will switch the port to local mode.

**Example**

```
Local>  SET PORT LOCAL ^V
```

**See Also** Set Session Passall, page 10-81, Set/Define Port Break, page 10-29.

# Set/Define Port Loss Notification

{SET | DEFINE} [PORT [*portlist* | ALL]] LOSS [NOTIFICATION] {ENABLED | DISABLED}

Sends the terminal device a Ctrl-G (Bell) when a typed character is lost due to a data error or an overrun on the ETS.

# Set/Define Port Menu (*)

{SET | DEFINE} [PORT [*portlist* | ALL]] MENU {ENABLED | DISABLED}

Specifies whether the port is placed in menu mode at login or the regular local prompt. If in menu mode, the user is given a list of commands he is allowed to enter, and cannot access the command line at all.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Menu, page 10-24, Clear/Purge Menu, page 10-5, Show/List Menu, page 10-86.

# Define Port Modem Control

{SET | DEFINE} [PORT [*portlist* | ALL]] MODEM [CONTROL] {ENABLED | DISABLED}

Specifies whether the modem control lines are to be used. The ETS only supports partial modem control (DSR/DTR). Modem Control must be **disabled** to use DSRlogout. Modem Control implies DSRlogout, in that the ETS will attempt to log out any connections if the port's DSR signal drops. Note that Modem Control cannot be Set - it must be Defined.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Port Dsrlogout, page 10-33.

# Set/Define Port Multisession (*)

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{MULTISESSION}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Enables or disables multisessions on the port. Note that Multisession should be enabled on the terminal device before it is enabled on the port. Multisession can be disabled on the ETS; disabling Multisession on the terminal device will also disable it on the port and logout both sessions. Note also that Multisession must be explicitly turned off - it is not cleared by logging out of the ETS port.

**See Also** Multisessions, page 2-23.

# Set/Define Port Name

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\begin{bmatrix}\textit{portlist}\\\text{ALL}\end{bmatrix}\right]\text{NAME}\ \textit{portname}$$

Used to set a unique name for each port. Remote LAT connections to the ETS can use either a service name or the port name to identify the port to connect to. Names can be up to 16 characters long.

**Restrictions** This command requires privileged status.

**Options** ***portname***
A 16 character name composed of alphanumeric or the underscore (“_”) character. Default is Port_n, where n is the port number. If the name is not enclosed in quotation marks, it will be converted to uppercase.

**Example**

```
Local>  SET PORT NAME "HighSpeed_Modem"
```

# Set/Define Port Parity

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{PARITY}\left\{\begin{matrix}\text{ODD}\\ \text{EVEN}\\ \text{MARK}\\ \text{SPACE}\\ \text{NONE}\end{matrix}\right\}$$

Sets the serial port's parity.

**Errors** Autobaud will not work unless the port is using 8 bit characters, or 7 bit characters with Even parity.

**See Also** Set/Define Port Autobaud, page 10-26.

# Set/Define Port Passflow

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{PASSFLOW}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

Specifies that Xon/Xoff characters will be added to the data stream as well as used for flow control. These commands apply only to the Xon/Xoff flow control.

# Set/Define Port Password

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right]\text{PASSWORD}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

Controls whether a password is needed to log in to the server from this port. The Set Server Login Password command is used to set the password itself.

**Restrictions** This command requires privileged status.

# Set/Define Port Preferred (*)

```
{ SET    } [PORT [portlist]] PREFERRED
{ DEFINE }       [ALL     ]

{ NONE                                                                              }
{ TELNET hostname                                                                   }
{ [SERVICE] service [NODE nodename] [ENVIRONMENT] [DESTINATION port]                }
```

Specifies a default service (either LAT or Telnet) for this port. The ETS will attempt to use the preferred service for Autoconnecting, as well as when no service name is specified in a Connect, Telnet, or Rlogin command. Note that via the environment string, a preferred host can be connected to via Rlogin, not just Telnet.

**Parameters**

**None**
Clears any previous preferred host.

**hostname**
Telnet host name or IP address.

**Service**
LAT service to set.

**Examples**

```
Local>  SET PORT PREFERRED Vax5
Local>  SET PORT PREFERRED TELNET 192.0.1.3
Local>  SET PORT PREFERRED TELNET TODD
```

**See Also**

Connect, page 10-9, Telnet, page 10-95, Rlogin, page 10-19, Set/Define Port Autoconnect, page 10-27, Set/Define Port Dedicated, page 10-31.

# Set/Define Port Queue (*)

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\\text{ALL}\end{matrix}\right]\right]\text{QUEUE}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

If queueing is enabled on a port, that port will be "queued" when it requests a connection to a LAT service that is busy. If queueing is not enabled, the connection will be rejected. If the port is queued, the connection will be accepted in the order in which it was received by the target node.

**Restrictions** Secure users cannot use this command.

# Set/Define Port Remote Configure

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\\text{ALL}\end{matrix}\right]\right]\text{REMOTE [CONFIGURE] }\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Controls whether actions on a remote LAT/VMS host are allowed to affect the local port on the ETS. The VMS command Set Term/Speed, for instance, will attempt to configure the local server port (using LAT) as well as the remote Term setting if this is enabled. Certain application programs, such as EMACS, will also need to configure the port.

# Set/Define Port Security

$$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}\left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\\text{ALL}\end{matrix}\right]\right]\text{SECURITY}\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$$

Setting a port to Secure status restricts its access to terminal commands and the ability to get information about other ports using Show/List commands.

**Restrictions** This command requires privileged status.

**See Also** Port Security, page 3-22.

# Set/Define Port Session Limit (*)

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\left[\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\right]\text{SESSION [LIMIT]}\left\{\begin{array}{c}\textit{limit}\\\text{NONE}\end{array}\right\}$$

Limits the number of active sessions on a port. The maximum on any port cannot exceed the server session limit.

**Restrictions** This command requires privileged status.

**Parameters**

**None**
Allows the maximum number of sessions (set with the Set Server Session Limit command).

**limit**
A value between 0 and 8. The default is 4.

**See Also** Set Server Session Limit, page 10-70.

# Set/Define Port Signal Check

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\left[\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\right]\text{SIGNAL CHECK}\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$$

Set Port Signal Check and Define Port Signal Check determine whether or not the DSR signal will be checked for when remote connections to the port are made. If enabled, remote connections to the port will not be permitted unless the DSR signal is asserted.

**Restrictions** This command requires privileged status.

# Set/Define Port Slip (*)

```
{ SET    } [PORT [portlist]] SLIP { ENABLED  }
{ DEFINE }       [ALL     ]       { DISABLED }
```

Allows or prohibits this port from starting SLIP sessions (via the Set SLIP command). Note that Define SLIP Port n implicitly does a Define Port n SLIP Enabled.

**Restrictions** This command requires privileged status.

**See Also** Clear/Purge Slip, page 10-7, Show/List Slip, page 10-92, SLIP, page 2-9.

# Set/Define Port Telnet Pad

```
{ SET    } [PORT [portlist]] TELNET [PAD] { ENABLED  }
{ DEFINE }       [ALL     ]               { DISABLED }
```

If Telnet Pad is enabled (the default), the server will automatically pad Carriage Returns with null characters for Telnet sessions.

**Restrictions** This command requires privileged status.

**Example** `Local>  SET PORT TELNET ENABLE`

# Set/Define Port Termtype (*)

```
{ SET    } [PORT [portlist]] TERMTYPE { TermString }
{ DEFINE }       [ALL     ]           { NONE       }
```

Used to specify a terminal type for the port. This terminal type is reported to the destination node in Telnet and Rlogin commands, and is a completely arbitrary string. Example terminal types might be VT100 or IBM1000.

**Options**

**TermString**
Up to 16 characters long. The string is converted to all uppercase unless it is enclosed in quotes. The default is no terminal type. Use None to clear the field.

**Example** `Local>  DEF Port 4 TERMTYPE HP2492`

# Set/Define Port Type

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\left[\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\right]\text{TYPE}\left\{\begin{array}{c}\text{ANSI}\\\text{SOFTCOPY}\\\text{HARDCOPY}\end{array}\right\}$$

Describes the type of device connected to the port.

**Parameters**

**ANSI**
VT100 compatible devices.

**Softcopy**
VT100 without clear screen or cursor controls (default).

**Hardcopy**
Deleted characters are echoed between backslashes, no cursor movement.

# Set/Define Port Username

$$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}\left[\text{PORT}\left[\begin{array}{c}\textit{portlist}\\\text{ALL}\end{array}\right]\right]\text{USERNAME}\left\{\begin{array}{c}\textit{username}\\\text{NONE}\end{array}\right\}$$

Used to specify a username for the port. When the username is Defined, you will not be asked for one when logging in to the port. To clear the username, enter **Set Port Username None**.

**Parameters**

***username***
Up to 16 characters long. The name is converted to all uppercase unless it is enclosed in quotes. The default is no username. Enter the keyword None to clear the field.

# Set/Define Port Verification

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\} \left[\text{PORT}\left[\begin{matrix}\textit{portlist}\\ \text{ALL}\end{matrix}\right]\right] \text{VERIFICATION} \left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}$$

If enabled, the server will issue informational messages whenever a session is connected, disconnected, or switched.

### Restrictions

Only privileged users can change the characteristics of other ports, and some options require privileges to Set or Define the characteristics of any port.

### Examples

```
Local> SET PORT USERNAME "Ted Kennedy" COMMAND
           COMPLETION ENABLED
Local> DEFINE PORT 7-8 SPEED 9600 AUTOBAUD ENABLED
```

# Set/Define Printer

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\} \text{PRINTER} \left\{\begin{matrix}\text{AUTO}\\ \text{CENTRONICS}\\ \text{DATAPRODUCTS}\\ \text{MPX}\left\{\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right\}\end{matrix}\right\}$$

Configures the type of parallel port interface to be used for the server's parallel port. Auto will attempt to configure the port automatically based on the signals seen from the print device at connection time. MPX Enabled configures the server to use the MPX serial port (port 3 for EPS4/12 or port 1 for the EPS1/2) as a back channel to the printer. The parallel port will send data to the print device, while the print device can use the serial port to send data to the server (and to a network host, presumably).

### Restrictions

This command requires privileged status.

### Parameters

**Auto**
Printer port will try to determine the type of print device attached automatically.

**Centronics**
Printer port is dedicated to being a Centronics-type port.

**Dataproducts**
Printer port is dedicated to being a Dataproducts-type port.

# Set/Define Printer (cont.)

**Parameters** **MPX**

The server will treat serial port 1 (on EPS1/2) or 3 (on EPS4/12 and ETS8/16) as a back channel for printer control information. Can be used with Define only.

**Example**

```
Local> DEFINE PRINTER CENTRONICS MPX DISABLED
```

**See Also** Show Printer, page 10-88.

# Set Privileged/Noprivileged

```
SET { PRIVILEGED [OVERRIDE] }
          NOPRIVELEGE
```

Changes the current port's privilege status. Only one port on the server can be privileged at any time. The Override parameter is provided to force your current port to become the privileged port (and the previously privileged port loses the privilege).

When changing your port to privileged status, you will be queried for the privileged password. The factory default privileged password is **system**; this password can be changed with the Set Server Prvileged Password command. If the password is forgotten, the server can be reset to factory defaults.

**Restrictions** To use Privileged, the user must know the privileged password. Secure users cannot become privileged.

**Examples**

```
Local> SET NOPRIVILEGED
Local> SET PRIVILEGED OVERRIDE
Password> ETHERN  (not displayed when typed)
```

**See Also** Restoring Factory Defaults, page 3-45.

# Set/Define Protocol AppleTalk Zone

{SET / DEFINE} PROTOCOL APPLETALK ZONE *ZoneName*

Allows the specification of a zone other than the default. If an AppleTalk router is present, it may offer more than one zone; this command is used to put the ETS into a zone other than the default. If the zone specified cannot be confirmed by a router, the command will fail.

**Restrictions** This command requires privileged status.

**Parameters** ***ZoneName***
Name of the zone to move the ETS into.

**Example** `Local> DEFINE PROTOCOL APPLETALK ZONE "AcctZone"`

**See Also** Show/List Protocol AppleTalk Zone, page 10-89, AppleTalk Server Parameters, page 3-6.

# Set/Define Protocol IP

{SET / DEFINE} PROTOCOL IP { IPADDRESS *x.y.z.a* | [SECONDARY] { GATEWAY *host* | LOADHOST | NAMESERVER *x.y.z.*a } }

All configurable IP parameters available under Set/Define Server are also available under Set/Define Protocol IP. The functionality is identical.

# Set/Define Protocol NetWare Access

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\} \text{PROTOCOL NETWARE ACCESS} \left\{\begin{matrix}\text{ALL}\\ \text{LOCAL}\\ \textit{fileserver}\end{matrix}\right\}$$

Configures a list of fileservers the ETS will contact for print jobs. By default, only fileservers on the local network will be queried; this command can be used to add to or restrict that category. Individual fileservers can be specified with Local, to allow Local access plus the specified non-local servers. If neither Local nor All are specified, only the listed fileservers will be contacted.

**Restrictions** This command requires privileged status.

**Parameters**

**Local**
Only fileservers on the local network will be queried.

***fileserver***
Individual fileserver name, 8 characters or less.

**All**
All fileservers, including those on other networks, will be contacted.

**Example**

```
Local> DEFINE PROTOCOL NETWARE ACCESS LAB_FS4
```

**See Also** Set/Define Server Software, page 10-71, Show/List Protocol Netware Access, page 10-89, *NetWare Server Parameters*, page 3-5.

# Set/Define Protocol NetWare Encapsulation

```
                                                  ┌ETHER_II┐
  SET                                             │  SNAP  │  ENABLED
{ DEFINE } PROTOCOL NETWARE ENCAPSULATION        {│ NATIVE │} { DISABLED }
                                                  └ 802_2  ┘
```

In conjunction with Set Protocol Netware Routing, configures the frame types that the ETS will pay attention to. This setting is ignored if routing is Enabled, since all frame types are accepted in router mode. When routing is disabled, this setting is used to control which frame formats the ETS will accept and generate traffic for.

Note that multiple frame types can be enabled at once; this may cause error messages from some NetWare fileservers due to the same network number being on different frame types. To correct this, enable NetWare routing.

**Restrictions** This command requires privileged status.

**Parameters**

**ETHER_II**
Use the Ethernet v2 frame format.

**NATIVE**
Use the "native mode" NetWare frame format.

**SNAP**
Use the 802.2 frame type with SNAP SAPs.

**802_2**
Use the 802.2 frame type with NetWare SAPs.

**Example**

```
Local> DEFINE PROTOCOL NETWARE ENCAPSULATION ETHER_II
          DISABLED
```

**See Also** Show/List Protocol Netware, page 10-89, Set/Define Protocol Netware Routing, page 10-52, *NetWare Server Parameters*, page 3-5.

# Set/Define Protocol NetWare Loadhost

| |
|---|
| { SET / DEFINE } PROTOCOL NETWARE LOADHOST *fileserver* |

Specify the name of the fileserver to attempt to download from when the unit is booted. This parameter is only useful if it is Defined; if it is Set, it will be cleared/ reset at boot time.

**Restrictions** This command requires privileged status.

**Parameters** ***fileserver***
Fileserver name, 8 characters or less.

**Example** `Local> DEFINE PROTOCOL NETWARE LOADHOST LAB_FS4`

**See Also** Set/Define Server Software, page 10-71.

# Set/Define Protocol NetWare Printserver

| |
|---|
| { SET / DEFINE } PROTOCOL NETWARE PRINTSERVER *fileserver* |

Configure the fileserver that the ETS will be dedicated to for Rprinter operation.

**Restrictions** This command requires privileged status.

**Parameters** **fileserver**
Fileserver name, 8 characters or less.

**Example** `Local> DEFINE PROTOCOL NETWARE PRINTSERVER LAB_FS4`

# Set/Define Protocol Netware Reset

| |
|---|
| { SET / DEFINE } PROTOCOL NETWARE RESET |

**Restrictions** This command requires privileged status.

Instructs the Print Server module to immediately go to the network and rescan for new connections. This is typically necessary when setting up queues or print servers using PCONSOLE.

# Set/Define Protocol NetWare Routing

```
  SET                                  ENABLED
{       } PROTOCOL NETWARE ROUTING {           }
  DEFINE                               DISABLED
```

Configures whether the ETS will act as an internal router if there are multiple NetWare frame types on the LAN. If enabled, the ETS will advertise all its NetWare services as part of an internal network, and will advertise itself as a "router" to that network. If disabled, the setting of Set Protocol Netware Encapsulation is used to decide which frame types to accept.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Protocol Netware Encapsulation, page 10-50, Show/List Protocol Netware, page 10-89.

# Set/Define Server Announcements

```
  SET                              ENABLED
{       } SERVER ANNOUNCEMENTS {           }
  DEFINE                           DISABLED
```

Sets whether or not the ETS sends messages over the network advertising available local services. If no services are active, no messages are sent in either case.

**Restrictions** This command requires privileged status.

# Set/Define Server BOOTP

```
  SET                      ENABLED
{       } SERVER BOOTP {           }
  DEFINE                   DISABLED
```

Enables or disables querying for a BOOTP host at system boot time.

**Restrictions** This command requires privileged status.

# Set/Define Server Broadcast

| { SET / DEFINE } SERVER BROADCAST { ENABLED / DISABLED } |
|---|

Enables or disables broadcasts from the server's ports.

**Restrictions** This command requires privileged status.

**See Also** Broadcast, page 10-4.

# Set/Define Server Buffering

| { SET / DEFINE } SERVER BUFFERING *buffersize* |
|---|

Specifies the size of buffer (in bytes) to use for network (LAT and TCP/IP) connections. The default is 512 bytes and should be sufficient for most cases. The size can be increased for large data transfers (file transfers, for example).

**Restrictions** This command requires privileged status.

**Parameters** **buffersize**
Buffer size in bytes: minimum 128, maximum 4096.

**Example**

```
Local> SET SERVER BUFFERING 1024
```

# Set/Define Server Circuit Timer

| { SET / DEFINE } SERVER CIRCUIT [TIMER] *TimerValue* |
|---|

Specifies the delay between messages from the server to other nodes. Do not alter this parameter while there are active sessions (and it should not need to be changed at all under normal circumstances).

**Restrictions** This command requires privileged status.

**Parameters** **TimerValue**
Timer value in milliseconds (from 30 to 200).

**Example**

```
Local> SET SERVER CIRCUIT TIMER 55
```

**See Also** *LAT and Services*, page 2-2.

# Set/Define Server Console

| |
|---|
| { SET / DEFINE } SERVER CONSOLE [PORT] *portnum* |

Specifies which of the server's ports acts as the console port. The console receives all error messages and is the only port that is enabled at boot time. Note that the console at boot time will be configured as if it were a real port, i.e. all baud, parity and flow control settings will be set as if the port were real.

**Restrictions** This command requires privileged status.

**Parameters** **port_num**
Port number of the new console port.

**Example** `Local> SET SERVER CONSOLE 5`

# Set/Define Server Domain

| |
|---|
| { SET / DEFINE } SERVER DOMAIN *DomainName* |

Specifies the default domain name to use when attempting to resolve text TCP/IP host names. For example, a domain name of "weasel.ctcorp.com" would result in a name resolution attempt for "alex.weasel.ctcorp.com".

**Restrictions** This command requires privileged status.

**Parameters** **DomainName**
64 characters maximum length

**Example** `Local> SET SERVER DOMAIN weasel.ctcorp.com`

**See Also** *TCP/IP*, page 2-5.

# Set/Define Server Dump

$\begin{Bmatrix}\text{SET}\\\text{DEFINE}\end{Bmatrix}$ SERVER DUMP $\begin{Bmatrix}\text{ENABLED}\\\text{DISABLED}\end{Bmatrix}$

Enables or disables upload dumping of a memory file (when either a serious error occurs or when signalled by a remote NCP request or Crash command).

**Restrictions** This command requires privileged status.

**See Also** Crash 451, page 10-10.

# Set/Define Server Gateway

$\begin{Bmatrix}\text{SET}\\\text{DEFINE}\end{Bmatrix}$ SERVER GATEWAY *IPaddress*

Specifies the host to be used as a TCP/IP gateway between networks. If a connection to a machine on a different network is attempted, messages will be directed to the gateway for forwarding. If the gateway is undefined in this case, an error message will be returned. (This command has no effect on LAT connections.)

**Restrictions** This command requires privileged status.

**Parameters** **IPaddress**
In numeric IP format

**Example** `Local>  SET SERVER GATEWAY 192.0.1.27`

**See Also** *TCP/IP*, page 2-5, Set/Define Server Subnet Mask, page 10-72.

# Set/Define Server Groups

See Set/Define Server Service Groups, page 10-69.

# Set/Define Server Host Limit

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\text{ SERVER HOST [LIMIT] }\left\{\begin{matrix}\textit{limit}\\ \text{NONE}\end{matrix}\right\}$$

Sets the maximum number of TCP/IP hosts the server will keep information for. Only hosts seen with the Rwho broadcasts are subject to this limit. If the new limit is less than the current limit and the host table is full, the limit will be slowly weeded down to the new value.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***

A value between 0 and 200. If number is None, no limit is set.

**Example**

```
Local> SET SERVER HOST LIMIT 6
```

# Set/Define Server Identification

{SET / DEFINE} SERVER IDENTIFICATION *IDstring*

Specifies the identification string that is broadcast with the LAT service messages. The identification string is also broadcast as the ident string for a LAT login service.

**Restrictions** This command requires privileged status.

**Parameters** ***IDstring***
Up to 40 characters of identification data

**Example**

```
Local> SET SERVER IDENTIFICATION "Googol Company Lat
        Box"
```

# Set/Define Server Inactivity Limit

{SET / DEFINE} SERVER INACTIVITY [TIMER] *limit*

Sets the period of time after which a port with Inactivity Logout enabled is considered inactive and is automatically logged out.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***
Inactivity period in minutes

**Example**

```
Local> DEFINE SERVER INACTIVITY LIMIT 20
```

**See Also** Set/Define Port Inactivity Logout, page 10-36.

# Set/Define Server Incoming

```
           LAT
           TELNET
 SET       BOTH
{     } SERVER INCOMING { NONE }
 DEFINE    PASSWORD
           NOPASSWORD
```

Allows or denies incoming LAT or Telnet connections and enforces password protection if desired. The default setting is None, Nopassword. The Show Server command shows the status of incoming connection parameters.

The status of the Incoming Telnet also controls incoming Rlogin sessions from remote hosts - the Set Server Rlogin command controls outgoing rlogin connections.

The Set Service Incoming LAT (or Both) will create a new LAT service that is used to log into the server. Use Set Server Telnet (or None) to remove the service and remove the ability to log in using LAT.

**Restrictions** This command requires privileged status.

**Example**

```
Local> SET SERVER INCOMING TELNET INCOM PASSW
(sets up password protected Telnet logins)
```

**See Also** Set/Define Server Rlogin, page 10-67, *LAT and TCP/IP Connectivity*, page 2-13.

# Set/Define Server Ipaddress

```
 SET
{      } SERVER IPADDRESS IPaddress
 DEFINE
```

Sets the server's IP network address. The IP address must be set before any TCP/IP connections (Telnet, Rlogin, Reverse Telnet) can be started. See the Glossary for a detailed description of the address. The IP address cannot be changed when there are active TCP/IP sessions in progress. A default Subnet Mask will also be set when you set the IP address; it can be overridden with the Set Server Subnet Mask command.

**Parameters** **IPaddress**
In standard, numeric format.

**Example**

```
Local> SET SERVER IPADDRESS 192.0.1.49
```

**See Also** *TCP/IP*, page 2-5, Set/Define Server Subnet, page 10-72.

# Set/Define Server Keepalive Timer

{SET / DEFINE} SERVER KEEPALIVE [TIMER] *time*

Sets the period between "hello" messages to active, but quiet, LAT nodes.

**Restrictions** This command requires privileged status.

**Parameters** ***time***
Time in seconds between "hello" messages

**Example** `Local> SET SERVER KEEPALIVE TIMER 30`

# Set/Define Server Loadhost

{SET / DEFINE} SERVER LOADHOST *IPaddress*

Specifies the host to be used for downloads from TCP/IP hosts. The host name must be a numeric IP-style address (i.e. it cannot be nameserved). The ETS requests its run-time code from this host. For LAT-loading servers, this command has no function.

**Restrictions** This command requires privileged status.

**Parameters** **IP_address**
In numeric IP format

**Example** `Local> DEFINE SERVER LOADHOST 193.23.71.49`

**See Also** Chapter 8, *Unix Host Setup*.

# Set/Define Server Lock

{SET / DEFINE} SERVER LOCK {ENABLED / DISABLED}

Controls whether or not local users are permitted to Lock their ports.

**Restrictions** This command requires privileged status.

**See Also** Lock, page 10-14.

# Set/Define Server Login Password

{SET / DEFINE} SERVER LOGIN [PASSWORD] [*passwd*]

Specifies the password that is used to log in to the server from the serial ports or the network. If the password is not given on the command line, the user will be prompted for it and it will not be displayed when typed. Users will only be required to provide this password if their ports also have Password Enabled.

**Restrictions** This command requires privileged status.

**Parameters** ***passwd***
6 or less alphabetic characters.

**Examples**

```
Local> SET SERVER LOGIN PASSWORD
Password> platyp (password will not be displayed)
Verification> platyp
Local>
```

**See Also** Set/Define Port Password, page 10-40.

# Set/Define Server Maintenance Password

{SET / DEFINE} SERVER MAINTENANCE [PASSWORD] [*passwd*]

Specifies the password which allows remote NCP connections to the server. Unlike other ETS passwords, the maintenance password is a string of a maximum of sixteen hexadecimal digits (0-9, A-F, a-f) which defaults to all zeros (off). Zero is also NCP's default.

**Restrictions** This command requires privileged status.

**Parameters** ***passwd***
16 or fewer hexadecimal digits (0-9, A-F).

**Example**

```
Local> DEFINE SERVER MAINTENANCE PASSWORD 89aacb
```

# Set/Define Server Multicast Timer

{SET / DEFINE} SERVER MULTICAST [TIMER] *timer*

For LAT circuits, specifies how many seconds apart the ETS should broadcast its service announcement messages. This parameter should not need to be changed under normal circumstances.

**Restrictions** This command requires privileged status.

**Parameters** **timer**
Interval, in seconds, between multicasts.

**Example** `Local> SET SERVER MULTICAST TIMER 40`

**See Also** *LAT and Services*, page 2-2.

# Set/Define Server Name

{SET / DEFINE} SERVER NAME *ServerName*

Specifies the name of the ETS. The name string must be in quotes if lowercase characters are used.

**Parameters** **ServerName**
16 characters or fewer, alphanumeric only.

**Example** `Local> SET SERVER NAME "LATBOX2"`

# Set/Define Server Nameserver

{SET / DEFINE} SERVER NAMESERVER *IPaddress*

Specifies the IP address of the name server (if any) for TCP/IP connections. This host will attempt to resolve text Telnet hostnames into numeric form if the local host table cannot do so.

**Restrictions** This command requires privileged status.

**Parameters** **IPaddress**
The network address of the nameserving host, in numeric IP format.

**Example** `Local> SET SERVER NAMESERVER 192.0.1.49`

**See Also** TCP/IP, page 2-5.

# Set/Define Server Netware Loadhost

{SET / DEFINE} SERVER NETWARE LOADHOST *ServerName*

The loadhost parameter is used to specify name of the Netware file server to be used for downloading new software.

**Restrictions** This command requires privileged status.

**Parameters** **ServerName**
This parameter specifies the name of the file server to be used.

**Example** `Local> SET SERVER NETWARE LOADHOST FRED`

# Set/Define Server Netware Printserver

```
 SET
{      } SERVER NETWARE  PRINTSERVER ServerName
DEFINE
```

The Printserver parameter is used to specify name of the print server VAP/NLM running on the Netware file server. This is used when setting up the ETS as a RPRINTER client.

**Restrictions** This command requires privileged status.

**Parameters** **ServerName**
This parameter specifies the name of the file server to be used.

**Example** `Local> SET SERVER NETWARE PRINTSERVER FRED`

# Set/Define Server Netware Reset

```
 SET
{      } SERVER NETWARE  RESET
DEFINE
```

This command tells the print server to scan all accessible file servers for queues that it can service and would normally be used after configuring queues on a file server using PCONSOLE.

**Restrictions** This command requires privileged status.

**Example** `Local> SET SERVER NETWARE RESET`

# Set/Define Server Node Limit

{SET / DEFINE} SERVER NODE [LIMIT] {*limit* / NONE}

Sets the maximum number of LAT service nodes the server will keep information for. If the new limit is less than the current limit and the service table is full, the limit will be slowly weeded down to the new value.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***
A value between 0 and 200. If None is specified, there will be no limit on the number of nodes saved.

**Example**

```
Local> SET SERVER NODE LIMIT 6
```

# Set/Define Server Number

{SET / DEFINE} SERVER NUMBER *ServerNumber*

Sets the number of this server. On larger local area networks, more than one server may be present and each should have a unique number. Note that this command actually does nothing and is only present for compatibility reasons.

**Restrictions** This command requires privileged status.

**Parameters** ***ServerNumber***
A value between 0 and 32767.

**Example**

```
Local> SET SERVER NUMBER 42
```

# Set/Define Server Password Limit

{SET / DEFINE} SERVER PASSWORD [LIMIT] {*limit* / NONE}

Limits the number of failures allowed when doing a Set Privileged command. After *limit* retries, the port will be logged out. The user can abort the password process by typing Ctrl-Z instead of the password.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***
A value between 0 and 100. If zero is specified, the port is never logged out for too many password failures. If None is specified, limit will be set to the default value.

**Example**

```
Local> SET SERVER PASSWORD LIMIT 10
```

# Set/Define Server Privileged Password

{SET / DEFINE} SERVER PRIVILEGED [PASSWORD] [*passwd*]

Sets the password for becoming the "super-user" of the server. If the password is not specified on the command line, the user will be prompted for it, and it will not be displayed on the screen as it is typed.

**Restrictions** This command requires privileged status.

**Parameters** ***passwd***
Six or fewer alphanumeric characters.

**Example**

```
Local> SET SERVER PRIVILEGED PASSWORD "yodel"
```

**See Also** Set Privileged, page 10-47.

# Set/Define Server Prompt

$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}$ SERVER PROMPT *PromptString*

This command allows the manager to change the prompt that users see from the default **Local_x>** string. A string up to 16 characters long can be configured, and should be enclosed in quotes.

**Restrictions** This command requires privileged status.

**Parameters>** The following parameters can be included in the prompt string:

| | |
|---|---|
| %p | substitutes the current port’s name |
| %n | substitutes the current port’s number |
| %s | substitutes the current server name |
| %D | substitutes the product name (ETS8, etc) |
| %C | substitutes the company name (Lantronix) |
| %S | substitutes the current session name |
| %P | substitutes a **>** if user is currently privileged |
| %% | substitutes a % |

**Examples** (Shown with the prompt that might result on the next line)

```
Local> SET SERVER PROMPT “Port %n:“
Port 3: SET SERVER PROMPT “%D:%s! “
ETS8:LabServ! SET SERVER PROMPT “%p%S_%n%P%% “
Port_5[NoSession]_5>% SET SERVER PROMPT “Lcl_%n>%P”
Lcl_3>>
```

# Set/Define Server Queue Limit

$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}$ SERVER QUEUE [LIMIT] $\left\{\begin{array}{c}\textit{limit}\\ \text{NONE}\end{array}\right\}$

Limits the number of queue entries for users waiting for a local service. If connect requests come in after the limit is reached, they will be rejected.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***
A value between 0 and 32. If 0 is specified, queueing is not allowed and only the user who has possession of the service may use it.

**Example**

```
Local> DEFINE SERVER QUEUE LIMIT 4
```

# Set/Define Server RARP

| |
|---|
| {SET / DEFINE} SERVER RARP {ENABLED / DISABLED} |

Enables or disables querying for a RARP host at system boot time.

**Restrictions** This command requires privileged status.

# Set/Define Server Retransmit Limit

| |
|---|
| {SET / DEFINE} SERVER RETRANSMIT [LIMIT] *RetransLimit* |

Specifies the number of retries attempted if a LAT network message receives no acknowledgment. This limit may need to be increased on especially noisy or heavily-used networks.

**Restrictions** This command requires privileged status.

**Parameters** ***RetransLimit***
A value between 4 and 100.

**Example** `Local> SET SERVER RETRANSMIT LIMIT 5`

# Set/Define Server Rlogin

| |
|---|
| {SET / DEFINE} SERVER RLOGIN {ENABLED / DISABLED} |

This command restricts the use of the RLOGIN command from the server. If Rlogins are disabled, users may not RLOGIN to remote hosts, but incoming rlogin connections are still enabled. The current status is visible with the Show Server command. Use the Set Server Incoming command to enable or disable incoming LAT and Telnet logins.

**Restrictions** This command requires privileged status.

# Set/Define Server Secondary

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\ \text{SERVER SECONDARY}\left\{\begin{matrix}\text{GATEWAY}\\ \text{LOADHOST}\\ \text{NAMESERVER}\end{matrix}\right\}\left\{\begin{matrix}\textit{IPaddress}\\ \textit{hostname}\end{matrix}\right\}$$

Specifies a backup host for the loadhost, gateway, or nameserver parameters. The backup host will be used if the primary host cannot be accessed. Note that the Set Server Secondary Loadhost command must be Defined to be useful - if it is only Set, it will be cleared at boot time. If a text hostname is specified for any of the parameters and that hostname cannot be resolved, and error message will be printed and the parameter will not be configured.

**Restrictions** This command requires privileged status.

**Parameters**

***IPaddress***
An address in numeric IP format.

***hostname***
An address resolved into numeric form.

**Examples**

```
Local> DEFINE SERVER SECONDARY LOADHOST 192.0.1.89
Local> SET SERVER SECONDARY GATEWAY 192.0.1.10
Local> SET SERVER SECONDARY NAMESERVER 192.0.1.34
```

**See Also** Set/Define Server Loadhost, page 10-59, Set/Define Server Gateway, page 10-55, Set/Define Server Nameserver, page 10-62.

# Set/Define Server Service Groups

$$\left\{\begin{matrix}\text{SET}\\ \text{DEFINE}\end{matrix}\right\}\ \text{SERVER SERVICE [GROUPS]}\ \left\{\begin{matrix}\text{ALL}\\ \mathit{GroupList}\end{matrix}\right\}\left[\begin{matrix}\text{ENABLED}\\ \text{DISABLED}\end{matrix}\right]$$

Establishes the group numbers of the services provided by this server. Groups are numbered from 0 to 255, and all groups may be selected with the All option. Specifying either Enabled or Disabled adds or removes the specified groups, respectively. If neither is specified, the group list given replaces the current one. The groups can be separated by commas for individual numbers and hyphens for ranges of numbers.

**Restrictions** This command requires privileged status.

**Parameters**

***GroupList***
Replaces the server's list of group numbers.

**All**
All group numbers (0-255) are provided or disabled by the server.

**Enabled**
The given group numbers are added to the server's list of the service groups.

**Disabled**
The given group numbers are removed from the server's list of service groups.

**Examples**

```
Local> DEFINE SERVER SERVICE GROUPS 2,5,6,8-44
Local> SET SERVER SERVICE GROUPS ALL DISABLED
```

**See Also** *LAT and Services*, page 2-2, Set/Define Port [Authorized] Groups, page 10-26.

# Set/Define Server Session Limit

| | |
|---|---|
| {SET / DEFINE} SERVER SESSION [LIMIT] {*limit* / NONE} | |

Sets the limit on active sessions per port. Each port can have an individual limit less than or equal to this limit.

**Restrictions** This command requires privileged status.

**Parameters** ***limit***
A value between 0 and 8.

**Example** `Local> SET SERVER SESSION LIMIT 2`

# Set/Define Server Slip Password (*)

| | |
|---|---|
| {SET / DEFINE} SERVER SLIP [PASSWORD] [*passwd*] | |

Configures the password required to start SLIP on a port. If the password is not provided on the command line, it will be prompted for.

**Restrictions** This command requires privileged status.

**Parameters** ***passwd***
This parameter specifies the password that a SLIP user must enter when attempting a connection.

# Set/Define Server Software

| |
|---|
| {SET / DEFINE} SERVER SOFTWARE *filename* |

Specifies the name of the download software file (if any) the server will attempt to load at boot time. For IP or LAT-loading hosts, this is the file that will be requested at boot time. Note that this parameter is only useful if it is Defined; if it is Set, it will be cleared/reset at boot time. For TFTP loading, you can also specify the complete path name of the file if the file is located in a directory other than the default. The pathname can be up to 15 characters (not counting the filename), and you must enclose the path name in quotes if there are any lower case characters.

**Restrictions** This command requires privileged status.

**Parameters** ***filename***
Load file name, 11 characters or less. The server will add the ".SYS" extension to the name.

**Examples**

```
Local> DEFINE SERVER SOFTWARE ETS
Local> DEFINE SERVER SOFTWARE "SYS:\LOGIN\PS1.SYS"
Local> DEFINE SERVER SOFTWARE "/usr/rich/tscode"
```

**See Also** Set/Define Server Loadhost, page 10-59.

# Set/Define Server Startupfile

| |
|---|
| {SET / DEFINE} SERVER STARTUPFILE *host:filename* [RETRY {*retry* / NONE}] |

Configures the startup configuration file that the ETS will attempt to download at boot time. This file contains the ETS commands that will configure the server before the users and services, etc, are started. If a retry limit of NONE is specified, the ETS will retry failed downloads forever; otherwise it will retry that number of times and then boot normally. To load from a VAX or Ultrix machine via LAT, use the **node::filename** format. To load from a TCP/IP host via TFTP, use the **host:filename** format. If a text hostname is used for TFTP, the name must be resolvable at boot time, otherwise you must use an IP address. To load from a Novell fileserver, use the **node\sys:\login\filename** format.

Both the Telnet and NCP consoles are available at the time the server attempts to download the startupfile; if there is a problem with the download, you can still log into the server and determine what went wrong.

# Set/Define Server Startupfile (cont.)

**Restrictions** This command requires privileged status.

**Parameters**

***host***
MOP node name or TCP/IP hostname or IP address.

***filename***
Startup file name, 11 characters or less.

***retry***
The number of times to retry the download attempt; None will continually attempt to download.

**Examples**

```
Local> DEFINE SERVER STARTUPFILE “bob:start” RETRY 6
Local> DEFINE SERVER START hevax::start.com
Local>DEFINESERVERSTARTUPFILEengfs\sys:\login\start.cmd”
```

# Set/Define Server Subnet Mask

{ SET / DEFINE } SERVER SUBNET [MASK] *mask*

Sets an IP subnet mask for the server. A mask will be set by default when the IP address is set, and will be the correct one for most environments. It can be overridden with this command, however. In short, the mask is applied to target IP addresses to determine whether the destination address is on the local network segment or not. If not, the server’s gateway host will be accessed to provide the connection.

**Restrictions** This command requires privileged status.

**Parameters**

***mask***
In numeric IP format.

**Example**

```
Local>  SET SERVER SUBNET MASK 255.255.192.0
```

**See Also** *TCP/IP*, page 2-5, Set/Define Server Ipaddress, page 10-58, Set/Define Server Gateway, page 10-55.

# Set/Define Server UUCP

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVER UUCP $\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$

Enables or disables the UUCP handler on the ETS. If enabled, the ETS will listen to TCP/IP port 540 and attempt to connect any logins there to a service called “UUCP” (typically a serial line with an attached modem). If this service is non-existent, the connection will be closed.

**Restrictions** This command requires privileged status.

**See Also** *UUCP Support*, page 2-12.

# Set/Define Service Appletalk

$$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}\ \text{SERVICE}\ \mathit{ServiceName}\ \text{APPLETALK}\ \left\{\begin{array}{c}\text{ENABLED}\\ \text{DISABLED}\end{array}\right\}$$

Specifies whether AppleTalk clients will be able to use the service. If enabled, the service name will be displayed as a selectable printer in Macintosh chooser screens.

**Restrictions** This command requires privileged status.

# Set/Define Service Banner

$$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}\ \text{SERVICE}\ \mathit{ServiceName}\ \text{BANNER}\ \left\{\begin{array}{c}\text{ENABLED}\\ \text{DISABLED}\end{array}\right\}$$

Specifies whether the ETS should print a banner page before starting the job. Banners should be disabled (the default) for all PostScript and plotter (binary) data.

**Restrictions** This command requires privileged status.

# Set/Define Service Binary

$$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}\ \text{SERVICE}\ \mathit{ServiceName}\ \text{BINARY}\ \left\{\begin{array}{c}\text{ENABLED}\\ \text{DISABLED}\end{array}\right\}$$

If the binary characteristic is enabled on a service, character translation (i.e. <cr> to <cr><lf> translation and tab expansion will be performed on the print data. The binary characteristic should be disabled when printing PCL data.

**Restrictions** This command requires privileged status.

# Set/Define Service Connections

$$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}\ \text{SERVICE}\ \mathit{ServiceName}\ \text{CONNECTIONS}\ \left\{\begin{array}{c}\text{ENABLED}\\ \text{DISABLED}\end{array}\right\}$$

Specifies whether the service is accepting connect requests. Connections might be disabled to reconfigure the corresponding port, or to repair the attached hardware device.

**Restrictions** This command requires privileged status.

# Set/Define Service DLC

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* DLC $\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$

Specifies that the service will be handling DLC print requests from Windows NT hosts. Note that only one service on the ETS can have DLC enabled at any time.

**Restrictions** This command requires privileged status.

# Set/Define Service Formfeed

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* FORMFEED $\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$

If enabled (the default) the ETS will append a formfeed at the end of any LPR print jobs.

# Set/Define Service Identification

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* IDENTIFICATION *IDstring*

Specifies an identification string for this service to be sent by the server in multicast messages.

**Restrictions** This command requires privileged status.

**Parameters** ***IDstring***
1 to 40 characters of identification information.

*When spaces or lower-case characters are used,* ***IDstring*** *must be enclosed by quotation marks.*

**Example**

```
Local> SET SERVICE pplab5 IDENT "Printer for Lab 5"
```

# Set/Define Service Lanmanager

$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}$ SERVICE *ServiceName LANMANAGER* $\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$

Enables or disabled LanManager (NetBios) access to the specified service.

**Restrictions** This command requires privileged status.

# Set/Define Service LAT

$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}$ SERVICE *ServiceName* LAT $\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$

Enables or disabled LAT access to the specified service.

**Restrictions** This command requires privileged status.

# Set/Define Service NetWare

$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}$ SERVICE *ServiceName* NETWARE $\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$

Enables or disables Netware access to the specified service.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Protocol Netware, page 10-49, Netstat, page 10-16.

# Set/Define Service Password

$\left\{\begin{array}{c}\text{SET}\\\text{DEFINE}\end{array}\right\}$ SERVICE *ServiceName* PASSWORD $\left\{\begin{array}{c}\text{ENABLED}\\\text{DISABLED}\end{array}\right\}$

Specifies an access password that users must enter to connect to the service. If the password is not given on the command line, the user will be prompted for it.

**Restrictions** This command requires privileged status.

**Parameters** ***passwd***
1 to 6 alphanumeric characters.

**Example**

```
Local> DEFINE SERVICE lab5 PASSWORD “this”
```

# Set/Define Service Ports

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* PORTS $\left\{\begin{matrix}\textit{PortList}\\\text{ALL}\end{matrix}\right\}$

Specifies a list of ports that will support or offer this service. If Enabled or Disabled is specified, the ports listed will be added or removed from the port list, respectively. If neither is given, the given port list given will replace the old port list. Note that ports offering a service must be in the correct access mode for connections to succeed. See Set/Define Port Access on page 10-25 for more information.

**Restrictions** This command requires privileged status.

**Parameters** ***PortList***
**All**

**Example** `Local> SET SERVICE lab5 PORTS 3,4,7-8 ENABLED`

# Set/Define Service Postscript

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* POSTSCRIPT $\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$

If enabled, the ETS will assume there is a PostScript printer attached to the service port(s) and try to ensure a job is done before starting another. It will send a Ctrl-D to the attached device before each job, and wait for the printer to confirm with a Ctrl-D back before starting the job transfer. If this is not done, slower printers may lose jobs as new jobs overwrite the still-interpreting previous job. Setting PostScript mode is strongly recommended for all PostScript queues.

**Restrictions** This command requires privileged status.

# Set/Define Service Queue

$\left\{\begin{matrix}\text{SET}\\\text{DEFINE}\end{matrix}\right\}$ SERVICE *ServiceName* QUEUE $\left\{\begin{matrix}\text{ENABLED}\\\text{DISABLED}\end{matrix}\right\}$

Specifies whether new connect requests will be queued or not. If queueing is disabled, users wishing to connect must wait until the current user disconnects before accessing the service.

**Restrictions** This command requires privileged status.

**See Also** Set/Define Server Queue Limit, page 10-66.

# Set/Define Service RTEL

```
{ SET    } SERVICE ServiceName RTEL { ENABLED  }
{ DEFINE }                          { DISABLED }
```

Enables or disabled TCP/IP access to the specified service.

**Restrictions** This command requires privileged status.

# Set/Define Service Tcpport

```
{ SET    } SERVICE ServiceName TCPPORT portnum
{ DEFINE }
```

Specifies a TCP listener socket for this port. TCP connections to this socket will be connected to the service.

**Restrictions** This command requires privileged status.

# Set/Define Service Telnetport

```
{ SET    } SERVICE ServiceName TELNETPORT portnum
{ DEFINE }
```

Specifies a TCP listener socket for this port. TCP connections to this socket will be connected to the service. Unlike the TCPport option, a Telnetport socket will do Telnet IAC negotiations on the data stream.

**Restrictions** This command requires privileged status.

# Set/Define Service Virtual

| | |
|---|---|
| SET SERVICE VIRTUAL | { ENABLED<br>DISABLED } |

This command will create a LAT "service" out of a TCP/IP hostname. When LAT users connect to this "service" they are given a Telnet session with the specified host. See the section on Protocol Conversion for details of the connection mechanism. Using the Disabled option will remove the host as a service, but will leave it in the host table. Note that if the host is not reachable or resolvable at connect time, the connect attempt will fail. The service name to use should be the hostname to connect to; an identification string showing it as a Telnet connection to the host will be added if none is specified.

**Restrictions** This command requires privileged status.

**Example**

```
Local>  SET SERVICE hydra VIRTUAL ENABLED
```

**See Also** *LAT and TCP/IP Connectivity*, page 2-13, *Environment for TCP/IP and LAT*, page 2-21.

# Set Session Delete

| | |
|---|---|
| SET SESSION | { DELETE<br>BACKSPACE } |

Allows you to change what is sent to the remote TCP/IP host when you press the Delete key. This command has no effect in local mode or on LAT sessions; nor does it have any effect if Pasthru or Passall (see below) are in effect. This command and the Set Session Newline command may be helpful if you are getting odd output from a Telnet session.

**Parameters**

**Delete**
Send a delete character (ASCII 0x7f)

**Backspace**
Send a backspace character (ASCII 0x8, or Ctrl-H)

**Example**

```
Local> SET SESSION DELETE BACKSPACE
```

# Set Session Echo

SET SESSION ECHO $\left\{ \begin{array}{c} \text{ENABLED} \\ \text{DISABLED} \end{array} \right\}$

Requests the ETS do local echoing for TCP connections. Default is Disabled, on the assumption that the remote host will provide echoing.

# Set Session Interactive

SET SESSION INTERACTIVE $\left\{ \begin{array}{c} \text{ENABLED} \\ \text{DISABLED} \end{array} \right\}$

Allows server-specific keys (i.e. Forward, Backward, and Local) and messages to be interpreted by the ETS.

**See Also** Set Session Passall, page 10-81, Set Session Pasthru, page 10-81.

# Set Session Newline

SET SESSION NEWLINE $\left\{ \begin{array}{c} \text{CR} \\ \text{LF} \\ \text{CRLF} \end{array} \right\}$

Since different hosts treat the Carriage Return and Line Feed keys differently, you may need to change what is sent to the remote service when you press the newline (usually the <Return>) key. This command has no effect in local mode or on LAT sessions; nor does it have any effect if Pasthru or Passall (see above) are in effect. This command and the Set Session Delete command may be helpful if you are getting odd output from a Telnet session.

**Parameters**

**CR**
Send carriage returns (ASCII 0xA) only

**LF**
Send linefeeds (ASCII 0xD) only

**CRLF**
Send both carriage return and linefeed

**Example**

```
SET SESSION NEWLINE CRLF
```

# Set Session Passall

```
SET SESSION PASSALL
```

Passall disables server interpretation of switch characters, messages, and XON/XOFF flow control. Use this for binary file transfers (i.e., executable files, graphics, etc.).

**See Also** Set Session Interactive, page 10-80, Set Session Pasthru, below.

# Set Session Pasthru

```
SET SESSION PASTHRU
```

Disables server interpretation of switch characters and server messages, but not XON/XOFF flow control. Used for ASCII file transfers.

**See Also** Set Session Interactive, page 10-80, Set Session Passall, above.

# Set/Define Slip (*)

$$\text{SET SLIP} \left[ \begin{array}{c} \text{BROADCAST} \left\{ \begin{array}{c} \text{ENABLED} \\ \text{DISABLED} \\ \text{DEFAULT} \end{array} \right\} \\ \text{MTU } \mathit{mtuval} \\ \text{IPADDRESS} \left\{ \begin{array}{c} \mathit{IPaddress} \\ \mathit{hostname} \end{array} \right\} \end{array} \right]$$

Set SLIP and Define SLIP do very different things, so they will be discussed separately. Set SLIP can only initiate a SLIP session on your port. The remote IP address and any options can be set on the command line or obtained from any server or port SLIP defaults. Define SLIP is used to configure default SLIP parameters for the server as well as set up permanent SLIP sessions for particular ports. There are common elements between the two commands:

- IP addresses can be specified as either numeric addresses or hostnames. Text names will be resolved when the command is entered; if they cannot be resolved, the command will fail.

# Set/Define Slip (*) (cont.)

- The Broadcast option specifies whether broadcast IP packets will be forwarded to the remote SLIP host. There are 3 choices for Broadcast - Enabled, Disabled, and Default. The Default setting takes its value from the SLIP defaults; if none is set, it defaults to Disabled.
- There are 3 options for MTU - a number, "Default" (use the configured default number, or RFC if not defined) or "RFC" (use the 1006-byte RFC value).

The Set SLIP command initiates a SLIP connection. The IP address of the remote SLIP host and the MTU and Broadcast parameters are optional; the server SLIP defaults will be used if they are not specified. An IP address must be present, either on the Set SLIP command line or in the Define SLIP tables.

Notes to keep in mind when initiating SLIP sessions:

- When SLIP interpretation begins, the Local> prompt will no longer be accessible unless the port is logged out (via the Logout command, DSRlogout, modem controls, etc).
- Once started, each SLIP connection is not affected by the Set command - you must log out of the port for the SLIP session to end, or to change connection options.
- Doing a Define SLIP Port n implicitly does a Define Port n SLIP Enabled.

**Parameters**

***mtuval***
One of (number, RFC or Default).

***broadval***
One of (Enabled, Disabled, or Default).

***host***
Either an IP address or a text host name.

**Examples**

```
Local> Set SLIP IPAD tunafish  MTU 500
Local> Set SLIP [uses all defaults]
```

# Set/Define Slip (*) (cont.)

**See Also** *SLIP*, page 2-9, Show/List Slip, page 10-92, Set/Define Server Slip, page 10-70.

Define SLIP is used to configure either SLIP defaults (to be used when a user doesn't explicitly specify options in the Set SLIP command) or permanent SLIP connections for particular ports. The MTU, Broadcast, and IPaddress options are as above.

The Force option controls whether the SLIP defaults will take precedence over options in the Set SLIP command line. If FORCE is enabled, the defaults will take priority; if disabled, the Set SLIP command line options will take precedence over the defaults. If the Autostart attribute is enabled, the port will treat all serial input as SLIP packets. It will run SLIP until the port is logged out, either via the Logout Port command or the drop of the DSR serial signal. It will then remain idle until the port is active, start SLIP interpretation, etc. If the Autostart attribute is not enabled, the Define command will be treated as specifying defaults for that particular port.

```
             [ AUTOSTART autostart   ]
             [ BROADCAST broadcast   ]
DEFINE SLIP  [ MTU mtuval            ]
             [ FORCE force           ]
             [ IPADDRESS address     ]
```

**Restrictions** This command requires privileged status.

**Parameters**

***mtuval***
One of (number, RFC or Default).

**broadval**
One of (Enabled, Disabled, or Default).

***host***
Either an IP address or a text host name.

***forceval***
One of (Enabled, Disabled, or Default).

**Examples**

```
Local> Define SLIP Port 5  IPAD 192.0.1.100  MTU 500
Local> Define SLIP DEFAULT MTU RFC BROADCAST
              ENABLED FORCE ENABLED
```

**See Also** *SLIP*, page 2-9, Show/List Slip, page 10-92, Set/Define Server Slip, page 10-70.

# Set/Define SNMP

$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}$ SNMP COMMUNITY *community* ACCESS *access*

Configures a community name and access mode for SNMP access. Each name has an access restriction associated with it; if an SNMP command comes in with an unknown name or an unauthorized command, an SNMP error reply will be sent. Community names are not case-sensitive.

**Restrictions** This command requires privileged status.

**Parameters**

***community***
A text name, up to 16 characters long.

**access**
One of Readonly, Both, or None.

**Example**

```
Local> Set SNMP SUNMAN ACCESS BOTH
```

**See Also** Show/List SNMP, page 10-92, Clear/Purge SNMP, page 10-7.

# Set/Define Telnet Hosts

$\left\{\begin{array}{c}\text{SET}\\ \text{DEFINE}\end{array}\right\}$ [TELNET] HOSTS *hostname ipaddress*

Allows you to associate a TCP/IP *hostname* with an IP address in the local host table. This will allow you to use the *hostname* for Telnet connections, even if there is no name server to resolve host names. If the given host name has already been Set or Defined, the new IP address will replace the previous value. The host name is limited to 64 alphanumeric characters, with only 16 characters between any period delimiters.

**Restrictions** This command requires privileged status.

**Parameters**

***hostname***
The hostname string you wish to define

***ip_address***
Standard, numeric IP address of the machine referred to by the hostname.

**Errors** IP addresses specified in a questionable format will be so noted.

**Example**

```
Local> Set HOST SPECTRE 192.0.1.15
```

# Set/Define Telnet Hosts (cont.)

**See Also** Clear/Purge Telnet Hosts, page 10-8, Show Telnet Hosts, page 10-93.

# Show/List Dialback (*)

$$\left\{ \begin{array}{c} \text{SHOW} \\ \text{LIST} \end{array} \right\} \text{DIALBACK}$$

Displays the currently configured dialback strings, as well as the number of connect attempts with that string and the number of connect failures.

**Restrictions** This command requires privileged status.

**See Also** Clear/Purge Dialback, page 10-4.

# Show/Monitor/List Hosts

See Show/Monitor/List Telnet Hosts, page 10-93.

# Show/Monitor/List Ipsecurity

$$\left\{ \begin{array}{c} \text{SHOW} \\ \text{MONITOR} \\ \text{LIST} \end{array} \right\} \text{IPSECURITY}$$

Displays the current or saved TCP/IP security table. Each address or address range is displayed, with the direction of concern, as well as the list of ports affected by the entry.

**See Also** Clear/Purge Ipsecurity, page 10-5, Set/Define Ipsecurity, page 10-22.

# Show/Monitor/List Logging

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\\\text{LIST}\end{array}\right\}\text{LOGGING}$$

Displays the current or saved event logging configuration.

**See Also** Set/Define Logging, page 10-23.

# Show/List Menu (*)

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\\\text{LIST}\end{array}\right\}\text{MENU}$$

Displays the current or saved Menu entries.

**See Also** Clear/Purge Menu, page 10-5, Set/Define Menu, page 10-24.

# Show/Monitor Nodes

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\end{array}\right\}\text{NODES}\left[\begin{array}{c}\textit{nodename}\\\text{ALL}\end{array}\right]\left[\begin{array}{c}\text{COUNTERS}\\\text{STATUS}\\\text{SUMMARY}\end{array}\right]$$

Used to display information about all or a particular LAT service node(s). If the server has no knowledge of the node you specify, it will tell you so. If All or * is specified, all of the known nodes on the network are listed. All is the default. The Summary option (the default) shows the node(s), the LAT protocol version, and the node's identification string. The Status option adds the node's Ethernet address and all the services the node offers, as well as their ident strings. The Counter option shows the LAT message and error counters associated with the node. Note that multicasts are not counted as messages in this field - only connection messages are counted.

**Restrictions** Monitor Nodes requires privileged status.

**Errors** Nodes cannot be Listed.

**Example** `Local> MONITOR NODE calvin COUNTERS`

# Show/Monitor/List Ports

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\\\text{LIST}\end{array}\right\}\text{PORTS}\left[\begin{array}{c}\text{ALL}\\ \text{ACCESS}\left\{\begin{array}{c}\text{LOCAL}\\\text{DYNAMIC}\\\text{REMOTE}\\\text{NONE}\end{array}\right\}\\ \textit{portnum}\end{array}\right]\left[\begin{array}{c}\text{CHARACTERISTICS}\\\text{COUNTERS}\\\text{STATUS}\\\text{SUMMARY}\end{array}\right]$$

These commands display information about the server’s ports. The current port is the default, unless another port number or All is specified; you can also get information about all the local ports having a particular Access value. The Counters option will show the port’s local and remote accesses as well as the communication errors on the device. The Status option shows the port’s connection status: the number of sessions and information about the current connection. It also shows the current flow control state (flow controlled or not) and the state of the DSR and DTR serial signals. The Summary option shows the access type, any offered services, and the login status of the port. The Characteristics option (the default) will show the port’s settings, such as baud rate, parity, preferred services, name, username, and group codes.

If the port is a virtual port, irrelevant information, such as baud rate, parity, or flow control will not be displayed. Any List on a virtual port will display the template port’s configuration.

**Restrictions** Monitor Ports requires privileged status. Secure ports cannot Show or List ports other than their own.

**Errors** Status and Counters parameters are not valid with List. Counters is not valid for virtual ports.

**Example**

```
Local> LIST Port ACCESS DYNAMIC COUNTERS
```

# Show/Monitor/List Printer

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\\\text{LIST}\end{array}\right\}\text{PRINTER}\left[\begin{array}{c}\text{CHARACTERISTICS}\\\text{COUNTERS}\\\text{STATUS}\\\text{SUMMARY}\end{array}\right]$$

These commands display information about the server's parallel printer port(s). The Counters option will show the port's local and remote accesses as well as the communication errors on the device. The Status option shows the port's connection status: the number of sessions and information about the current connection. The Summary option shows the access type, any offered services, and the login status of the port. The Characteristics option (the default) will show the port's settings, such as printer type, preferred services, name, username, and group codes.

**Restrictions** Monitor requires privileged status.

**Errors** Status and Counters parameters are not valid with List.

**Example** `Local> SHOW PRINTER STATUS`

# Show/Monitor/List Protocols

$$\left\{\begin{matrix}\text{SHOW}\\ \text{MONITOR}\\ \text{LIST}\end{matrix}\right\}\text{PROTOCOLS}\left[\begin{matrix}\text{LAT}\\ \text{APPLETALK}\left[\begin{matrix}\text{NODES}\\ \text{ZONES [LOCAL]}\end{matrix}\right]\\ \text{NETWARE [ACCESS]}\\ \text{TCP}\end{matrix}\right]$$

The Show Protocols command by itself will display a summary screen of all 4 protocols. General figures, such as packet counts and error status, will be shown. For detail on an individual protocol, the 4 subcommands below are provided. They show in much greater detail what is happening with each protocol stack. Also refer to the Netstat command, which shows currently active network socket connection, as opposed to counters.

**Restrictions** This command requires privileged status.

**Parameters** **Access**
Shows the current NetWare access list.

**Errors** Status and Counters parameters are not valid with List.

**Example** `Local> SHOW PRINTER STATUS`

**See Also** Netstat, page 10-16.

# Show/Monitor/List Queue

$$\left\{\begin{array}{c}\text{SHOW}\\ \text{MONITOR}\\ \text{LIST}\end{array}\right\}\text{QUEUE}\left\{\begin{array}{c}\text{PORT } \textit{portnum}\\ \text{NODE } \textit{nodename}\\ \text{SERVICE } \textit{servicename}\end{array}\right\}$$

Show Queue will display the entries in a particular local service's connect queue, if it exists. Particular sets of queues or entries can be selected with the parameters: Node *node_name* shows only connect requests from a specific network node; Port *port_num* shows the entries that could be serviced by the specified port; and Service *service_name* shows the entries waiting for the specified service. All (the default) can also be specified to show all entries.

**Restrictions** Monitor is a privileged command.

**Examples**

```
Local> SHOW QUEUE Port 6
Local> MONI QUEUE Service lab5
```

# Show/Monitor/List Server

$$\left\{\begin{array}{c}\text{SHOW}\\ \text{MONITOR}\\ \text{LIST}\end{array}\right\}\text{SERVER}\left[\begin{array}{c}\text{BOOTPARAMS}\\ \text{CHARACTERISTICS}\\ \text{COUNTERS}\\ \text{SUMMARY}\\ \text{STATUS}\end{array}\right]$$

This command is used to display the global attributes or counters for the server itself. It also displays the software version number and the time since the last reboot. The Counter fields are described below. The Bootparams option shows parameters related to rebooting the unit and reloading the software file. The Characteristics option (the default) displays the LAT network settings, the IP addresses, the server's enabled characteristics (Broadcast, for example), and the local limits (password, queue, session, etc). The Counters option shows the accumulated counters for the Ethernet, LAT and TCP/IP protocols. These will be reset to zero after a Zero Counters All command. The Status option displays current server use, such as active ports, services, and circuits, for example.

**Restrictions** Monitor is a privileged command.

**Errors** Only Characteristics is valid with List Server.

**Examples**

```
Local> SHOW SERVER STATUS
Local> MONITOR SERVER COUNTERS
```

# Show/Monitor/List Services

$$\left\{\begin{matrix}\text{SHOW}\\ \text{MONITOR}\\ \text{LIST}\end{matrix}\right\}\text{SERVICES}\begin{bmatrix}\text{LOCAL}\\ \textit{service}\\ \text{ALL}\end{bmatrix}\begin{bmatrix}\text{CHARACTERISTICS}\\ \text{COUNTERS}\\ \text{SUMMARY}\end{bmatrix}$$

This command is used to display the characteristics of the services on the network. Remember that this list is masked by the service groups that this port is eligible to see, i.e., users will not see services they cannot connect to. The Local option is used to display all the services provided by this server, whether they are available or not. The All option shows all the network services usable by the current port. Individual services can be specified by name. The Characteristics option shows the service's rating, group codes, and, if the service is local, the service ports and service flags (such as Queueing, Connections, and Password). The Status option shows the service name, offering node, availability status, and ident string. The Summary option (the default) shows the service, ident string, and availability.

**Restrictions** Monitor Services requires privileged status.

**Parameters**

**Local**
Displays those services local to this server

**All**
Displays all known services.

***service***
Displays any known information about this service. Wild cards are permitted.

**Examples**

```
Local> SHOW SERVICE lab5_prtr STATUS
Local> MONITOR SERVICE LOCAL SUMMARY
```

# Show/Monitor Sessions (*)

{ SHOW | MONITOR } SESSIONS [ PORT *portnum* | ALL ]

This command displays information about the current port's sessions (or all ports' sessions if the All parameter is used).

**Restrictions** Monitor Sessions is a privileged command.

**Errors** Secure users cannot specify Port or All.

**Examples**

```
Local> SHOW SESSION
Local> SHOW SESSION PORT 5
```

# Show/List Slip (*)

{ SHOW | LIST } SLIP [PORT *portnum*]

Displays the current or saved SLIP entries. The SHOW command displays currently active SLIP connections. If a port number is specified, that port's connections and SLIP counters will be displayed. The List command displays permanently-configured address/port pairs, as well as any configured SLIP default options. The Port option cannot be specified with List.

**See Also** Clear/Purge Slip, page 10-7.

# Show/List SNMP

{ SHOW | LIST } SNMP

Displays the current or saved SNMP security table entries.

**Restrictions** This command requires privileged status.

**See Also** Clear/Purge SNMP, page 10-7, Set/Define SNMP, page 10-84.

# Show/Monitor/List [Telnet] Hosts

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\\\text{LIST}\end{array}\right\}\text{[TELNET] HOSTS}\left[\begin{array}{c}\text{ALL}\\\textit{hostname}\end{array}\right]$$

Displays either the currently available TCP/IP (Telnet/rlogin) hosts (Show) or the ones that have been Defined locally in the host table (List). Hosts will be shown with the method of discovery (i.e., rwho, connection, host table, etc.) and will also be marked if they are the current nameserver and/or gateway. Specifying a particular host name will show only that host's information. Wildcards for the hostnames are allowed. The All option is the default, and it displays all known TCP/IP hosts.

**Restrictions** Monitor Hosts requires privileged status.

**Examples**

```
Local> SHOW TELNET HOSTS ALL
Local> LIST HOST MYVAXSTN
```

**See Also** Set/Define Telnet Hosts, page 10-84.

# Show/Monitor Users

$$\left\{\begin{array}{c}\text{SHOW}\\\text{MONITOR}\end{array}\right\}\text{USERS [}\textit{username}\text{]}$$

These commands display the current users logged onto the server. For each user, the ETS displays the port username and current connection information. If a *username* is given, only information for that user is shown.

**Restrictions** Monitor is a privileged command.

**Errors** List Users will cause an error.

# Source

SOURCE *host*: [:] *filename* [VERIFY]

*If **filename** contains lower-case letters, it must be enclosed in quotation marks.*

Source attempts to download a configuration file from a MOP, TFTP, or NetWare host. The file is assumed to be lines of server commands which will be executed. The Source command is most useful for trying out a configuration file before using the Set Server Startupfile command (page 10-71). To download a file from a MOP host, use the **node::filename** format; to use TFTP, use the **hostname:filename** format. For Netware, use the **hostname\sys:\login/filename** format. If the Verify option is included, each command from the downloaded file will be echoed before executing it. Note that files to be downloaded via NetWare must be in the login directory on the fileserver, due to access restrictions.

**Parameters**

**host**
Either a MOP node name, NetWare fileserver name, or a TFTP host (text name or IP address)

**filename**
The download path and filename, 22 characters maximum.

**Restrictions** This command requires privileged status.

**Examples**

```
Local>  SOURCE ALVAX::start.com veri
Local>  SOURCE "labsun:start.com"
Local>  SOURCE LABFS4\SYS:\LOGIN\ETS.COM
```

**See Also** Set/Define Server Startupfile, page 10-71.

# Stty

The Stty command is functionally identical to the Set/Show Ports commands (page 10-25). If used with no parameter, it is equivalent to Show Ports. If used with a parameter, it is treated as a Set Ports command.

**Examples**

```
Local>  STTY
Local>  STTY SPEED 9600
```

# Su

The Su command is functionally identical to the Set Privileged command (see page 10-47). It is provided for Unix compatibility.

# Telnet (*)

Telnet is a shorthand for the Connect Telnet command, and is provided for simplicity. The hostname can be either text or an IP address. You can specify a port number for Telnet connections (i.e.- *x.x.x.x:port*) if you do not want to use the default Telnet *port_number*. This method is only recommended for hosts that cannot use the normal Telnet port. If a preferred Telnet service has been defined, the *host_name* parameter is optional. An environment string can also be added to the command - see the Concepts chapter for details.

**Examples**

```
Local> TELNET 192.0.1.252
Local> TELNET 192.0.1.252:2812 (connects to port 2812)
Local> TELNET dopus.csedu.uvts.edu
```

**See Also**

Connect, page 10-9, Set/Define Port Preferred, page 10-41, *Telnet*, page 2-5, and Chapter 8, *Unix Host Setup*.

# Test Loop

$$\text{TEST LOOP } \mathit{address} \left[ \text{HELP} \left\{ \begin{matrix} \text{RECEIVE} \\ \text{TRANSMIT} \\ \text{FULL} \end{matrix} \right\} \text{ASSISTANT } \mathit{address} \right]$$

This command is used to test the network connections to other hosts. A simple loopback service verifies that the remote node is receiving the server's transmissions. A second, "helper" node can be specified, which will forward one or both of the transmissions (outbound or incoming). The nodes can be specified as text names, or as Ethernet addresses in the form xx-xx-xx-xx-xx-xx, where each xx is a 2 digit hex number representing one of the 6 bytes of the node's hardware address. If text names are used, only LAT service names can be resolved to hardware addresses.

**Restrictions**

This command requires privileged status.

**Examples**

```
Local> TEST LOOP 45-a2-ed-48-12-3c
Local> TEST LOOP 12-68-df-ea-38-c5  HELP
             REC ASSI 78-23-ad-2c-11-4e
Local> TEST LOOP LABVAX
```

# Test Port

TEST PORT [*portnum*] [POSTSCRIPT] [ WIDTH *width* | COUNT *count* ]

Tests a serial port's connection. A continuous stream of ASCII alphabetic characters (each a line of width *w*) is sent from the port until the number of lines specified by Count is reached. You can stop the test by pressing any key. Virtual and multisession-enabled ports can only be tested by the user on that port.

Specifying the Postscript option sends a PostScript test page to the port instead of ASCII data. The Count parameter controls the number of pages to print in this case, and the Width parameter is ignored.

**Restrictions** Non-privileged users may only test their own port.

**Examples**

```
Local> TEST Port
Local> TEST Port 4 WIDTH 45 COUNT 5
```

# Test Service

TEST SERVICE *Servicename* [POSTSCRIPT] [ DESTINATION *port* | NODE *nodename* | COUNT *count* ]

Tests a connection to a service, with an optional Destination port and Node. A continuous stream of ASCII alphabetic characters (each a line of width *w*) is sent to the service and (hopefully) echoed back until the number of lines specified by Count is reached. You can stop the test by pressing any key. The ETS will show the number of packets sent and lost. Testing a service local to this ETS will fail, as will testing either the Login Service (see Set/Define Server Incoming, page 10-58) or a proxy Telnet service (see Set/Define Service Virtual, page 10-79). Use Test Port if you are unsure of the connection to a local service.

Specifying the Postscript option will perform a bidirectional data test of the interface between the ETS and the printer. This test will not generate any paper output. It should echo the postscript data back to the ETS and will issue a status message on the result of the test.

**Restrictions** Non-privileged users may only test their own port.

**Example**

```
Local> TEST Service ALEX COUNT 1000 WIDTH 80
```

# Unattach Port (*)

| |
|---|
| UNATTACH PORT *portnum* |

Detaches a previously Attached port. Use Disconnect to disconnect your own port. Use Logout Port to disconnect all sessions and to logout a port.

**Restrictions** This command requires privileged status.

**Example** `Local> UNATTACH PORT 5`

# Unlock Port (*)

| |
|---|
| UNLOCK PORT*portnum* |

Unlocks a locked port, necessary if the user has locked the port and forgotten the password. The command does nothing if the port is already unlocked.

**Restrictions** This command requires privileged status.

**Example** `Local> UNLOCK PORT 6`

# Who

The Who command is functionally identical to the Show Users command, discussed on page 10-93. It is provided for Unix compatibility.

# Zero Counters

ZERO COUNTERS [ ALL | NODE *nodename* | PORT *portnum* ]

This command is used to reset the counters for errors and other network and server events. The counters zeroed can be All (Ethernet, LAT, TCP/IP, SLIP, and all serial ports) Node (counters for a remote LAT node) or Port (for events associated with a particular serial port, including SLIP events).

**Restrictions** To zero some other port (or All or Node) is a privileged command.

**Examples**

```
Local> ZERO COUNTER NODE vax5
Local> ZERO COUNTERS Port 6
```

# Technical Support

If you are experiencing problems with the ETS or have suggestions for improving the product, please contact Lantronix Technical Support at (800) 422-7044 or (714) 453-3990. We are also reachable via Internet email at **support@lantronix.com**.

If you are submitting a problem, please provide the following information:

- Your name, company name, address, and phone number
- Product name
- Unit serial number
- Software version (available by issuing the **Show Server** command)
- Network configuration including the output from a Netstat command
- Description of the problem
- Provide a Debug report (stack dump) if applicable
- Product status when the problem occurred. Please try to include information on user and network activity at the time.
- If the problem is with a particular port, please have the results of a Show Port Counters and Show Port Characteristics for that port.

Enjoy your ETS, and we thank you for your patronage!

LANTRONIX
15353 Barranca Parkway
Irvine, California 92718 USA
714-453-3990 • FAX 714-453-3995 • Toll Free 800-422-7044

Email: **support@lantronix.com**

# Technical Support

# Glossary

**AppleTalk**

AppleTalk was developed by Apple Computer to allow networking between Macintoshes. All Macintosh computers have a LocalTalk port, running AppleTalk over a 230kbit serial line. AppleTalk also runs over Ethernet (EtherTalk, which the ETS units support) and Token Ring (TokenTalk) network media. Due to dynamic addressing and powerful name directory services, installing an AppleTalk device is generally as simple as attaching it and turning it on.

**ARP**

Address Resolution Protocol. A network protocol that allows hosts to discover a node's **hardware address** from its **IP address**. ARP requests are generally sent as broadcast to all nodes, and the node whose IP address matches that in the request replies. The arp command on Unix hosts shows the current IP address to hardware address mapping; the arp command on the ETS shows the current IP host table.

**AUI**

Attachment Unit Interface. A 15-pin shielded, twisted pair Ethernet cable used (optionally) to connect between the ETS and a **MAU**.

**Autobaud**

Autobauding occurs when the server tries to determine a terminal device's speed (baud rate) and parity by the characters that are being sent at login time. Ideally, the user could know nothing about setting up his or her terminal, would turn it on and press the Return key, and the server will configure itself automatically. Autobaud will not work for all parity and baud rate combinations - it will only function if the port is using either 10-bit characters with no parity, or 10-bit characters with even parity. It also will not work for device speeds below 1200 baud. Depending on the speed of the terminal device, it may take 2 to 4 presses of the <Return> key at login time.

**Backbone**

Any communications network has a main "cable," and devices can be thought of as attaching to this main conductor. ("Cable" is misleading, because networks can be over microwave or fiber optic carriers, as well.) For example, in a university setting, the backbone might be the cable that runs to all the buildings that need network access. Local sub-networks and devices are attached off the main backbone.

**BOOTP**

BOOTP is a TCP/IP network protocol that lets “dumb” network nodes request configuration information from a BOOTP “server” node. At boot time, the dumb node sends a broadcast message requesting information and waits for a reply. The BOOTP host, if configured, provides the dumb node with an IP address, the IP address of a load host (usually itself) and the name of the download file to use. The dumb node needs only know its own hardware address, as this is what the BOOTP server uses to decide whether it can help the node and what information to provide. BOOTP implementations usually have a configuration file (such as **/usr/etc/bootptab**) that lists the hardware and IP addresses of BOOTP clients, as well as the download file they should use. Most TCP/IP hosts can act as servers if BOOTP replies have been enabled. See your local documentation (man pages, etc) for specific information.

**Console**

The terminal used to configure the ETS at boot time. When the server first powers up and cannot access a load file, a console is enabled (on Port 1) to let the server manager set up the server. This console port is the only port that is operational at boot time. After the server is running normally, the server will send diagnostic and error messages to this port. See the Set Server Console command for more information.

**Dialback**

A security feature that ensures people do not log into usernames or ETS ports that they shouldn’t have access to. When dialback is enabled on a port, the user is prompted for a username. If that username is not in the dialback table, the port is logged out and ETS access is denied. If it is in the table, the port is logged out and the ETS sends a command to the serial port (and presumably the attached modem) to try to reconnect to the user. The command is commonly a modem command to dial a particular phone number, thus verifying that the user is accessing the server from the proper place.

**Domain Name**

A domain name is a text name appended to a host name to form a unique host name across internets. Very loosely, it is a text name corresponding to the network portion of an IP address. For example, IP address 192.0.1.150 might correspond to host “alex.ctcorp.com”; if the hostname is “alex”, the domain name is “ctcorp.com”. The domain name on the ETS is used for **nameserving**, when a host might be known only by its full name (including domain name).

**Download**

The transfer of a file or information from one network **node** to another. Generally it refers to transferring a file from a “big” node, such as a computer, to a “small” node, such as a terminal server or printer. Because the server code is downloaded (as opposed to having the operating software permanently in the device’s ROM), the code can be upgraded easily. Unless you have the **ROM** version of the ETS, the server will have to download its executable code from a **host** before it can operate normally.

**Flash ROM**

See **ROM**.

**Gateway**

A TCP/IP gateway machine is a host that can access two or more different Ethernet networks. As such, it can forward messages across networks that other hosts would not be able to access. The host generally has multiple IP addresses (one for each network). For example, a gateway might “live” at addresses 192.0.1.8 and 193.0.1.8. Hosts wishing to forward messages address the messages to the gateway, which will then pass them on to the other network. The ETS supports this with the Set Server Gateway command. The IP address of the gateway host on this server’s network is specified, and messages for hosts not on this network will be forwarded. For example, if an ETS at address 192.0.1.33 wished to talk to a host at address 196.0.1.58, he would have to go through a gateway machine on his network. The gateway would forward messages between networks 192.0.1 and 196.0.1. See the **subnet mask** entry below for information on how these networks work.

**Hardware Address**

See **Network Address**.

**Host**

Host is generally used in this manual to mean a **node** on a network that can be used interactively, i.e. logged into. This generally refers to interactive computers. In the ETS’ command set, host is used to refer to a TCP/IP node, rather than TCP/IP and LAT nodes. For example, the Show Hosts command displays only the known TCP/IP nodes.

**Host Table**

A host table is a list of TCP/IP **hosts** on the network along with their **IP addresses**. Having this list locally means users wishing to connect to these hosts can use text names (rather numeric IP addresses) without using a **nameserver**. See the Set Telnet Host and Show Hosts commands, as well as the “Host Table” section of the Concepts chapter for more information.

**ICMP**

ICMP stands for the Internet Control Message Protocol. ICMP messages are sent by TCP/IP **nodes** in response to errors in TCP/IP messages or queries from other nodes. They are sent, for example, when a node sends a packet to an incorrect gateway host, or if a network packet expires. ICMP messages may also be sent as broadcasts, not just sent to a specific host. The ETS counters for ICMP messages may be helpful for detecting network problems.

**IP Address**

See **Network Address**.

**LAT**

Local Area Transport. LAT is a Digital Equipment Corp. proprietary network communication **protocol**. The protocol is based on the idea of a relatively small, known number of hosts on a local network sending small network packets at regular intervals. LAT will not work on a wide area network scale, as **TCP/IP** does, as it cannot distinguish large numbers of **nodes**. For local networks, however, LAT is usually faster and less prone to pauses than TCP/IP. It also allows remote connections to **services** and other network devices, of which TCP/IP has no understanding. The fact that the ETS supports both LAT and TCP/IP protocols transparently means that you are not bound by the limitations of either protocol.

**Login Service**

If the ETS is configured to allow users to log in from the network (as opposed to logging in from the serial ports) it creates what is called a login **service**. The ETS advertises a new service with the same name as the server. **LAT** users who connect to the "service" will not be connected to a physical port or device; rather, they will be given a "virtual port" on the ETS. They obtain a "Local>" prompt and can issue commands as if they were physically connected to the ETS. When they log off the ETS, their "port" disappears. Note that each login to the ETS consumes a **virtual port**.

**MAU**

Medium Attachment Unit. This is a small device used as a transceiver between a **Thinwire** network cable and an **AUI** cable or a **Thickwire** cable and an **AUI** cable. It consists of the physical adapter as well as the circuitry needed to convert signals from one medium to the other.

**MMJ**

Modified Modular Jack. These are the 7-pin connectors that the ETS uses to connect serial terminal lines to terminal devices. They are much smaller and easier to manage than the traditional RS-232 25-pin connectors. Many non-terminal devices, such as printers and modems, still use RS-232 connectors, so adapters are frequently needed. See Appendix A for pinout information. MMJ jacks can be distinguished from the similar RJ-12 jacks by the side locking tab, rather than a center-mounted one.

**MOP**

Maintenance Operations Protocol. MOP is a DEC **protocol** for Ethernet network traffic. The protocol is used for remote communications between **hosts** and devices on the network. Terminal servers use this protocol to download code from a host quickly and easily. For networks where LAT and MOP are the dominant network services, MOP is the simplest way to boot the server. At boot time, the server broadcasts a request on the network for a load file, and a MOP host will respond and send the file. MOP is also used to signal the server of an **NCP** request or connection from another host, and is the protocol that TSM is based on.

**MTU**

Maximum Transmission Unit. The MTU of a link is the maximum packet size, in bytes, that can be transmitted across the link. For Ethernet, this is 1536 bytes. For SLIP lines, it can be variable, based on each host's ability to receive and reassemble packets. The RFC for SLIP suggests an MTU of 1006 bytes, but this is not a requirement.

**Multicast**

A multicast is a message that is sent out to multiple devices on the network by a **host**. Multicasts are generally sent at specified intervals to avoid cluttering the network, and in the case of LAT, contain the name of the host sending them as well as information about what LAT **services** that host provides.

**Multisession**

Multisession is a parameter that allows multiple sessions to run simultaneously over a single serial cable. A multi-session-capable terminal is necessary to use multisessions on the server. The terminal generally provides a window for each session, giving the impression of multiple physical ports to the ETS. Multisession is used to allow terminals to provide two session "windows" that can display session output simultaneously. It allows only two sessions to be active at once, however. Multisession users on the ETS have access to TCP/IP hosts as well as LAT services.

**Name Server**

When starting Telnet and rlogin connections, the host to connect to can be specified one of two ways - either as a text name, such as "alex.weasel.citco.com," or as a numeric **IP Address** (such as 195.22.89.172). The latter form can be routed easily, since hosts on the network know how to map numeric addresses to exact hosts. They have more trouble with names, however, and this is where name servers come in. Some host(s) on the network are designated as name servers, to translate (or *resolve*) text-style names into numeric addresses. No Telnet connect request can be attempted until a numeric address is known for a host, so if there is no accessible name server (or the ETS has not been told of its identity with the Set Server Nameserver command), numeric addresses must be used.

**NCP**

Network Control Program. NCP is a program run on VMS machines to configure local network hardware and remote network devices. In the case of the ETS, NCP can be used to remotely log into and reboot the server or cause it to "dump" its memory to a host for analysis. It is can also be used to access the ETS' console port from a host VMS system.

**NetWare**

A Novell-developed Network Operating System (NOS). Provides file and printer sharing among networks of Personal Computers (PC's). Each NetWare network must have at least one fileserver, and access to other resources is dependent on connecting to and logging into the fileserver. The fileserver(s) control user logins and access to other network clients, such as user PC's, print servers (such as the ETS), modem/fax servers, disk/file servers, etc.

**Network Address**

Every **node** on a network has one or more addresses associated with it. Every node has what is called a "hardware address" that is unique across every network everywhere, at any time. If you know a node's hardware address, you should be able to identify the exact piece of equipment it goes with. Hardware addresses are generally set up by the company that manufactured the equipment and should never change. The hardware address is usually specified as a list of six hexadecimal numbers separated by dashes, such as "ae-34-2c-1d-69-f1." The hardware address for your ETS is shown on a label on the rear of the unit.

**Network Address (cont.)**

In the case of **TCP/IP** networks, each node also has a software or "IP" address. This is configurable by the managers of the nodes. The software address is usually specified as four decimal numbers separated by periods (for example, "197.49.155.247"). In this case, each number must be between 0 and 255, and each corresponds to a different network or sub-network. Depending on how many other nodes and networks a node can "see" on its network, addresses are either assigned to nodes (in the case of large, cross-country networks) or chosen randomly (for small networks that do not connect to the outside world). Each software address should be unique across all the networks it can access.

**Node**

A node is any intelligent device connected to the network. This includes terminal servers, host computers, and any other devices (such as printers and terminals) that are directly connected to the network. A node can be thought of as any device that has a "hardware address" (see **network address**, above). A "service node" is a node on the network that provides a service other users can connect to (a printer, for example). A terminal server that allows only local logins, on the other hand, is not a service node, as remote (network) users are not allowed to connect to it.

**PostScript**

A printer/display protocol developed by Adobe Corp. PostScript is an actual printing and programming language to display text and graphics. Unlike line/ASCII printers, which print character input verbatim, PostScript printers accept an entire PostScript page, and then interpret and print it. The programming aspects of PostScript can be used to define shapes and routines that will be used on successive pages, change fonts and text orientation, and print fine bitmap images within blocks of text.

**Protocol**

Any standard method of communicating over a network. There are protocols for placing actual bits onto the network cable; other protocols are layered on top from there. **LAT** is a protocol for network access, while both **TCP** and **IP** are also protocols (TCP runs on top of the IP layer/protocol.)

**Proxy Service**

A proxy **service** is created when the ETS is configured to offer a Telnet host as a LAT service. With the Set Service Virtual command, the ETS can be set to offer Telnet hosts as services to which LAT users can connect. When they connect, the ETS translates between the LAT and Telnet sessions. When the user logs out of the remote Telnet host, he is also disconnected from the ETS. Note that a user connecting "through" the ETS in this way consumes a **virtual port** while connected.

**RARP**

RARP is a protocol that allows a node to broadcast a message asking for an IP address given its hardware address. If a RARP server has been configured to reply to this node, it will tell the node what its IP address should be. Note that RARP provides less information than BOOTP, but is more widely implemented.

**Repeater**

A repeater is a network device that repeats signals from one cable onto one or more other cables, while restoring signal timing and waveforms. Repeaters are the most common way to connect local networks together, and can provide either **Thinwire** or **Thickwire** connections. They are commonly used to create larger local networks up to a certain limit based on the number of repeaters and the length of the cables.

**RFC**

Request For Comments--a standard document describing protocols, systems, or procedures used by the Internet community. For example, the IP network protocol is detailed in an RFC (RFC 791), as are SNMP, TCP, Finger, BOOTP, and the Domain name system. Information on obtaining RFC's is available from NIC@NIC.DDN.MIL.

**Rlogin**

Rlogin is an application that provides a terminal interface between (usually UNIX) hosts using the **TCP/IP** network protocol. Unlike **Telnet**, Rlogin assumes the remote host is (or behaves like) a UNIX machine. Rlogin can also be configured to disable login password checking, so should be used with care. See your host's documentation (especially regarding the *.rhost* and *hosts.equiv* files) for more information.

**ROM**

Read-Only memory, or a memory device that retains its information even when power to it is removed. A ROM version of the ETS does not need to **download** code, since it carries the entire executable code in ROM and thus never needs to reload it. Frequently the ROM is provided as "flash ROM", which can be reprogrammed if the software needs updating.

**Router**

On large networks, some hosts are connected to more than one network and are designated routers - they route messages on the net to the correct "place." They may send an incoming message to an adjacent network if they do not know the destination host by name, or they may send it right to the intended destination. On cross-country networks, for example, a message may go through several routers before arriving at its final destination.

**rwho**

Rwho is a UNIX feature that enables network **hosts** to know what users and systems are on the network without actively connecting to them. If rwho is enabled on a host, it both sends and receives network broadcasts containing this information. The packets generally contain the host's name, IP address, and the number and names of the users on the system. The ETS listens for these broadcasts and adds the hosts to the local **host table**, eliminating the need to **nameserve** for their names. Since the broadcasts are periodic (30 seconds to 2 minutes is typical) hosts that are seen with rwho and then removed from the ETS' host table will reappear later. Rwho usually has to be enabled explicitly on the host system, and may not be enabled on large network environments where the extra network traffic is unacceptable. In this case hosts can be added to the ETS' host table by hand.

**Service**

A service is any device on a network that can be connected to and accessed, such as a printer, modem, or a remote computer. Network users can generally see the services available on the network because the **nodes** that provide these services "advertise" them to the world. In the case of LAT, each service node sends out occasional network messages (or **multicasts**) describing what services it is providing and which users are connected to them. Note that the concept of services is one specific to **LAT** and local area networks. TCP/IP (and other wide area) networks have no such facility. The service will occasionally be used to refer to anything that can be connected, whether LAT or not.

**Session**

A session is any connection to a network **service**. The network service can be an interactive login to a LAT or TCP/IP host, a connection to a modem on the ETS or another server, or something else. Users on the ETS can have more than one session active at one time, allowing them to quickly switch between hosts or connections. The sessions are kept in a list, and can be switched with the Forward and Backward commands. They can be listed with the Show Session command. There is an absolute limit of eight sessions per port on the ETS; this limit can be set lower with the Set Server Session and Set Port Session commands.

**SLIP**

Serial Line Internet Protocol, SLIP is a protocol for running TCP/IP over serial lines. This allows hosts to dial into the ETS via modems, for example, and run TCP/IP connections to the ETS' attached Ethernet network. The ETS in this case routes packets between the serial lines and the ethernet, making connections to all the hosts transparent to the user. Since the serial line is actually carrying network packets and not plain keyboard data, there is no way to return to the ETS' local prompt without closing the SLIP session and re-logging into the server.

**SNMP**

Simple Network Management Protocol. SNMP allows a TCP/IP host running an SNMP application to query other nodes for network-related statistics and error conditions. The other hosts, which provide SNMP *agents*, respond to these queries and allow a single host to gather network statistics from many other network nodes. The terminal server provides this SNMP agent only - it cannot generate queries to other hosts. It only responds to them.

**Subnet mask**

The ETS supports Telnet connections across networks through the use of **gateways**. A gateway host will forward messages across network boundaries. At connection time, however, the ETS must decide whether the remote machine is on the local network (and can be accessed directly) or whether it is on a remote network (and must be accessed through a gateway). It does this with the subnet mask set up for the server. The ETS uses the mask as a filter; if the ETS' IP address and the remote IP address appear the same after the filter, the remote host is assumed to be on the same local network. Otherwise, the gateway is used. The mask itself is a list of bits that should be enabled in the result - a 1 in the mask means to let that bit in the IP address through, and 0 means do not.

**Subnet mask (cont.)**

For example, address 192.1.2.22 with mask 255.255.0.0 becomes 192.1.0.0. For network purposes, host 192.1.5.12 is on the same network, based on the mask specified. In this case, a gateway would not be accessed. A host at 192.8.12.34 would be considered on a different network, however, since the network mask comes out to be 192.8.0.0, and this does not match the previous two masks. In this second case, the gateway host would be used - if it had not been defined, the connect attempt would fail. If the subnet mask is not set explicitly (with the Set Serve Subnet command), the ETS will assume a mask based on the ETS' IP address (and thus the apparent network type). This mask will be 255.255.255.0 for most (19x.x.x.x and 2xx.x.x.x) IP addresses.

**Tap**

A tap usually refers to a point on the network **backbone** where other devices can be attached. If the network is thought of as a freeway, the taps on the network are the on- and off ramps to the freeway (like freeway access ramps, taps are usually where network problems occur). Taps may be of several types, including simple "T" connectors or "Vampire" taps that attach directly to a network cable.

**TCP/IP**

TCP and IP are the standard network **protocols** in UNIX environments. They are almost always implemented and used together and called TCP/IP. TCP/IP is an extremely flexible protocol, allowing reliable access to over four billion possible nodes anywhere in the world. It also allows many protocols to run on top of it, notably **Telnet, rlogin,** and **TFTP**. TCP/IP support may be integral to an operating system, as in UNIX, or it can be a separate product added later as on VMS machines.

**Telnet**

Telnet is an application that provides a terminal interface between hosts using the **TCP/IP** network protocol. It has been standardized so that telnetting to any host should give you an interactive terminal session, regardless of the remote host type or operating system. Note that this is very different from the **LAT** software, which allows only local network access to LAT hosts only.

**Telnet Proxy Service**

See **Proxy Service.**

**10BASE2**

**Thinwire** network cable.

**10BASE5**

**Thickwire** network cable.

**10BASE-T**

Ethernet running across Unshielded Twisted Pair (UTP) cable. Note that 10BASE-T is a point-to-point network media, with one end of the cable typically going to a repeater/hub and the other to the network device.

**TFTP**

Trivial File Transfer Protocol. On computers that run the **TCP/IP** networking software, TFTP is used to quickly send files across the network with fewer security features than FTP. TFTP is used by the ETS to download a boot file in the UNIX environment.

**Thickwire**

Ethernet cables generally come in two varieties - thickwire and **thinwire**. Thickwire network cable is 1/2" diameter coax cable. It is generally found on larger networks where a cable may travel for long distances and usually connects multi-user computers to the network. It is harder to work with than thinwire cable, but offers better noise and error protection and can be run much farther (up to 500m/1500 ft without repeaters) than **thinwire**. Connections to thickwire networks are usually made with 15-pin connectors.

**Thinwire**

Thin, co-axial cable similar to that used for television/video hookups and typically used with BNC-type connectors. Thinwire cable is much easier to route and work with than **thickwire**, but it should not be run more than 185 meters (~600 feet) without using a repeater to reinforce the signal. Thinwire connectors are usually seen on terminal servers, personal workstations, networked printers, and in networks where the **nodes** are all relatively close to each other. Thinwire is also occasionally referred to as "Cheapernet" due to its lower cost.

**Transceiver**

The actual device that interfaces between the network and the local **node**. When talking about networks, the term transceiver generally refers to any connector that actively converts signals between the network and the local node. An example of a transceiver is a **MAU**.

**TSM**

The Terminal Server Manager (TSM) program is a software package that allows terminal servers on a network to be remotely managed from another node. It is supported on VMS systems running the LAT protocol (and is incompatible with TCP/IP-only networks).

**Virtual Port**

When users log into the ETS through the network (or using a new multisession), they are not connected to any physical port, and thus have no default setup information for the port. They instead are given a partially-configured "virtual" port that can be set but not defined. When the user logs out or the ETS is rebooted, the virtual ports disappear. Rather than force each user to configure his port every time he logs in, a "template" port is also defined. Setup done to this port (port 0) will be applied to each new virtual port on the ETS. For example, if port 0 has Broadcast enabled, all the incoming logins will also have it enabled (unless they explicitly use the Set Port command to disable it after they log in). Other than this feature, the virtual ports behave much like the physical ports. They can be logged out with the Logout Port command, broadcasted to, and displayed as real ports. Due to system resource limitations, there is a limit of 16 virtual ports on the ETS.

# Index

## A



## B



## C



## D

# E



# F



# G



# H



# I



# J



# K



# L

# L (cont.)



# M



# N



# O



# P

# P (cont.)



# Q



# R



# S

# T



# U



# V



# W



# X



# Z