# TCP/IP Release Document

Software Version 3.0

Part No. 005498

Revision 05

This document describes TCP/IP Software Version 3.0.

The release notes for standard DOMAIN Software and other optional products are documented in other sets of release notes and are located in the system /doc directory.

APOLLO COMPUTER INC.
330 Billerica Road
Chelmsford, Massachusetts 01824

Print Date: January, 1987.

This document was formatted using the FMT tool distributed with the DOMAIN computer system.

Reader Notice

This document resides on line in the /doc directory. To print a copy of this document, use the **prf** command with the -npag and -pr options.

$ **prf <file-pathname> -pr <printer_name> -npag**

Contents

# CHAPTER 1

# OVERVIEW OF TCP/IP SOFTWARE VERSION 3.0

The DOMAIN Transmission Control Protocol/Internet Protocol (TCP/IP) product provides file transfer and remote log-in capabilities between DOMAIN nodes and other systems via an ETHERNET gateway.

TCP/IP Version 3.0 contains the following:

- A new feature, the subnet utility
- Changes to the **tcpstat** command
- Changes to running **tcp_server** with the -debug option
- Changes to TCP/IP installation procedure: TCP/IP No Longer Contains Driver Software
- Larger UDP packet size
- New DOMAIN **telnet debug** command
- Additional DOMAIN **ftp** commands
- Corrected bugs

## 1.1 THE SUBNET UTILITY

TCP and IP are protocols defined by the Defense Advanced Research Projects Agency (DARPA) to permit communication between networks using different protocols and transmission media. DARPA refers to this overall network of different networks as an Internet. One of the most well-known among DARPA Internets is the ARPANET, a communications network that spans the country.

According to the traditional design of the DARPA protocols, any communications network can communicate with a DARPA Internet as long as they supply a unique Internet address for each host within the network. (The DARPA Internet manages these addresses in network-wide routing tables.) This way, any host within a network could access the DARPA network.

Since then, many individual network administrators have created their own internets to address several communication needs. They use internets to:

- Separate sprawling communications networks into several manageable networks
- Connect geographically-separate Local Area Networks (LANs) with a high-speed point-to-point link
- Combine different types of LANs such as ETHERNET LANs and DOMAIN rings
- Ease network congestion by putting heavily-trafficked hosts on separate cables

The traditional DARPA Internet model does not support the concept of individual network users creating their own internets. That is, even though a communications network may belong to a larger internet, each network within an internet needs a unique network number to remain on a DARPA Internet. So, to communicate between two internets on a DARPA Internet, the hosts must know the network topology of the other.

Consider, for example, two hosts on the ARPANET -- one at the University of Southern California (USC) and the other at Massachusetts Institute of Technology (MIT). To send a message from the USC host to the MIT host, the USC host must specify the appropriate network within the internet at MIT.

Also, any changes to a network within an internet affect DARPA Internet routing tables. So, any time network administrators change networks within their internets, they must update the DARPA routing tables.

The subnet utility provides more flexibility in the network structure by allowing network administrators to subdivide their network without affecting the DARPA Internet. This way, administrators can keep their network activities separate from the entire DARPA Internet.

Referring to our example, the USC host can now send a message to host at MIT by specifying a network and host number. The network number represents the entire MIT internet. When the message reaches the MIT gateway, the gateway checks whether subnets are implemented, and if so, relays the message to the appropriate network within the MIT internet.

To implement a subnet utility, you don't have to use a different DARPA Internet addressing mechanism. You simply specify a new interpretation of the current Internet address by supplying a subnet mask.

Currently, you supply a 32-bit Internet address that identifies each host (workstation) on your network. The subnet utility allows you to specify a subnet address for each LAN within your network.

For each host on your network, you specify a unique 32-bit Internet address. DARPA defines three types of Internet address: A, B, or C. You can distinguish which type of address is in use by the size of each field. That is,

- Type A addresses have a 7-bit network number, a 24-bit host number and the value of the most significant (leftmost) bit is 0.
- Type B addresses have a 14-bit network number, a 16-bit host number and the value of the two most significant (leftmost) bits are 10.
- Type C addresses have a 21-bit network number, an 8-bit host number and the value of the three most significant (leftmost) bits are 110.

Figure 1 shows how a 32-bit Internet address is divided into network and host numbers. The M refers to the most significant bit field.

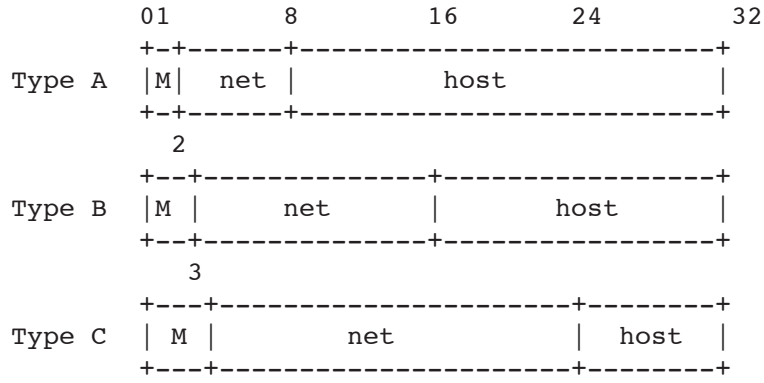



Figure 1. Type A, B, and C Internet Addresses

When using the subnet utility, you can further subdivide the Internet address into network, subnet and host field. Note that the size of the network address remains the same, the host field is divided into subnet and host fields.

Figure 2 shows some possible ways you can subdivide an Internet address into network, subnet and host numbers.

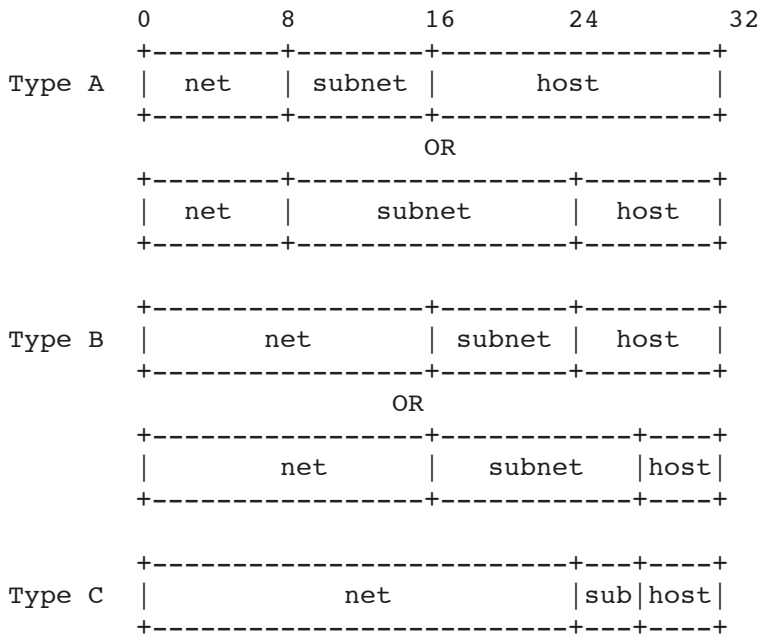



Figure 2. Internet Addresses with Subnet Fields

When determining your Internet addresses, you don't need to remember the size of each field. Instead, you can simply be sure to specify a number within the given range. The range of an the Internet address is represented in decimal number values. That is, the 4-byte Internet address is represented by four decimal numbers within the range of 0 and 255.

For example, Type C addresses have a one-byte host address, so you can choose any number between 1 and 254. (DARPA Internet reserves 0 and 255.) The network address is 3 bytes long and you can choose any number between 192.0.1 through 223.255.254. This number starts after 192 because the first three bits (0 through 192 in decimal) are reserved to signify the Type C address.

Table 1 summarizes the range of values you can specify for Type A, Band C addresses.

**TABLE 1. Range of Network and Host Values for Type A, B, and C Addresses**

| Type | Size in Bytes | | Range of Values | |
|---|---|---|---|---|
| | Network | Host | Network | Host |
| A | 1 | 3 | 1 - 126 | 0.0.1 - 255.255.254 |
| B | 2 | 2 | 128.1 - 191.254 | 0.1 - 255.254 |
| C | 3 | 1 | 192.0.1 - 223.255.254 | 1 - 254 |

To create a subnet, you subdivide the host portion of your Internet address. Table 2 lists the range of subnet and host values for each type. Note that since Type C host numbers are only 8 bits long, you're limited to 15 subnets and 14 hosts. For this reason, most users implement subnets with Type A or B addresses.

**TABLE 2. Range of Subnet and Host Values for Type A, B, and C Addresses**

| Type | Size in Bits | | Range of Values | |
|---|---|---|---|---|
| | Subnet | Host | Subnet | Host |
| A | 16 | 8 | 0.1-255.255 | 1 - 254 |
| A | 8 | 16 | 1 - 255 | 0.1 - 255.254 |
| B | 8 | 8 | 1 - 255 | 1 - 254 |
| C | 4 | 4 | 1 - 15 | 1 - 14 |

As stated previously, when implementing subnets you are merely changing the interpretation of your Internet address by supplying a bit mask or subnet mask. The mask identifies which bits of the Internet address correspond to a subnet number, and which bits correspond to the host number.

To supply the mask, you edit your Internet networks file (/sys/node data/[.node_id]/networks). You must supply the following:

- A semicolon to separate the mask information from the physical interface information
- The word, mask
- Your Internet address with network and subnet fields denoted by one's (255), and host field denoted by zero's (0)

For example, the following is a network entry without a subnet mask.

NOTE: Previously, TCP/IP releases referred to the ETHERNET interface as il. This has been changed to eth. Version 3.0 will accept either eth or il, however future revisions will accept only eth.

This example indicates that you have a Type A internet address, on an ETHERNET (eth) interface. We know this is a Type A address because the first number is within the range of 1 and 126.

```
10.9.9.7 on eth0
```

The following is a network entry with a subnet mask. Given that this is a Type A address, we know that the first field is the network number. The next field is the subnet number because it is all one's, and the host number corresponds to the last two bytes, as indicated by zeros.

```
10.9.9.7 on eth0;  mask 255.255.0.0
```

The following is a two-byte subnet mask for a Type A address:

```
10.9.9.7 on eth0;  mask 255.255.255.0
```

The following is a one-byte subnet mask for a Type B address where the first two bytes indicate the network number, the third byte is the subnet number, and the fourth byte is the host number.

```
129.9.9.9 on eth0;  mask 255.255.255.0
```

The following is a Type C address with a 4-bit subnet and 4-bit host field.

```
195.9.9.7 on eth0;  mask 255.255.255.240
```

For more information on specifying Internet addresses and editing the networks file, see the Configuring and Managing TCP/IP manual (008543).

## 1.2 CHANGES TO **tcpstat**

The TCP/IP **tcpstat** command with the -i option, which reports information about physical interfaces changed with this release. The command now displays an additional field, MASK, to report subnet masks. It also changed the way it reports status in the STAT field. We describe these changes in the sections below.

### 1.2.1 **tcpstat** TO REPORT SUBNETS

You can determine whether your network has a subnet utility by typing the **tcpstat** -i command. This command now displays a field containing the subnets mask for each physical network interface. This network mask corresponds to the information provided in the networks file.

Figure 3 is a sample report. (For printing purposes, we split this report in two.) It indicates that the first item is a DOMAIN ring interface (dr0) with a Type C Internet address and no subnets. The second item is an ETHERNET interface (eth0) with a Type A Internet address and no subnets. The third item is an ETHERNET interface (eth0) with a Type A Internet address, a one-byte subnet field, and a two-byte host field.

```
$ tcpstat -i

UNIT ADDRESS      STAT  IPKTS OPKTS RSTS FLSH
dr0  192.9.10.31  AI    1703  1460  7     0
eth0 127.0.0.1    AI    0     0     0     0
eth0  10.9.9.7    AI    0     0     0     0

OERR IERR COLL MASK
7    0    0    255.255.255.0
0    0    0    255.0.0.0
0    0    0    255.255.0.0

$
```

Figure 3. Generating a Status Report with **tcpstat -i**

Note that you cannot get this information about subnet masks with DOMAIN/IX BSD4.2 TCP/IP. The corresponding command, **netstat**, does not report information on the subnet utility.

### 1.2.2 **tcpstat** CHANGE TO STAT FIELD

The **tcpstat -i** status report, which reports the status flags for each physical interface, has been improved with this release. It reports more status codes, with mnemonics rather than bits. Table 3 lists the status codes and what they mean. Note that AI indicates that the interface is healthy. E indicates the interface is working, though some errors have occurred.

**TABLE 3. tcpstat Status Codes**

| Status Code | Meaning | Status Code | Meaning |
|---|---|---|---|
| A | available | G | global |
| C | initialization completing | I | initialized |
| D | disabled | M | subnets in use |
| E | error | W | waiting for initialization |
| F | flushing | | |

## 1.3 CHANGES TO RUNNING **tcp_server** with the DEBUG OPTION

You can run the **tcp_server** in a window with the debug option to troubleshoot TCP/IP. Prior to this revision, the debug mode displayed all its activity in the window. With Version 3.0, you can control what types of debug information **tcp_server** displays during the debug session by specifying values on the command line. This allows you to suppress debug information about activities that you don't need to see.

To get available debug information, run **tcp_server** in a window with the -debug option on the command line as follows:

```
$ /sys/tcp/tcp_server -debug [hexadecimal value]
```

The hexadecimal value you specify corresponds to a 16-bit mask. If the bit is set, the corresponding information will be displayed. The bits are defined as follows:

| Bit | Debug Information |
|---|---|
| 0001 | General information |
| 0002 | IP level information |
| 0004 | ARP information |
| 0008 | TCP information |
| 0010 | Data in TCP packets |
| 0020 | UDP information |
| 0200 | Broadcasts |
| 1000 | TCP Finite State Machine information |
| 2000 | Device level information |
| 4000 | Additional detail at any level |

If you specify the -debug option without any hexadecimal values, you'll get general information. To get additional information, you can specify bit values corresponding to other debug information.

For example, to specify TCP (0008) and IP (0002) information, you add the bits 0002 and 0008 to specify the following command line:

```
$ /sys/tcp/tcp_server -debug 000a
```

To specify TCP, IP, and device level (2000) information, you add the bits 0002, 0008, and 2000. So you specify the following command line:

```
$ /sys/tcp/tcp_server -debug 200a
```

Controlling which information gets displayed is often quite helpful during troubleshooting. For example, you might want to get all the available debug information except for broadcast information. (You might want to suppress broadcast information when **routed** or **rip_server** are running on the local network because they generate many broadcasts.) To get all the available debug information except broadcast information, supply the following bitmask on the command line:

```
$ /sys/tcp/tcp_server -debug f0ff
```

## 1.4 CHANGE TO TCP/IP INSTALLATION: TCP/IP NO LONGER CONTAINS DRIVER

Prior to this release, DOMAIN TCP/IP provided driver software that enabled TCP/IP to run over ETHERNET gateways. This version of TCP/IP does not provide driver software. Instead, you must first install the appropriate driver before installing or updating TCP/IP. You can refer to the following release documents for installing these drivers:

- EtherController-AT Release Document (009742) for installing an ETHERNET controller in DN3000 workstations and servers.

- EtherController-MB Release Document (009743) for installing an ETHERNET controller in workstations and servers that contain the MULTIBUS peripheral expansion cage.

  NOTE: Previously TCP/IP releases referred to the ETHERNET interface as il. This has been changed to eth. Version 3.0 will accept either eth or il, however future revisions will accept only eth.

## 1.5 LARGER UDP PACKET SIZE

TCP/IP Version 2.1 supported a maximum User Datagram Protocol (UDP) packet size of 1024 bytes. Version 3.0 supports a UDP size of up to 9132 bytes.

## 1.6 NEW telnet DEBUG COMMAND

You can now run **telnet** in debug mode by specifying the -debug option when you invoke **telnet** as follows:

```
$ telnet -debug
```

When you run **telnet** in debug mode, **telnet** generates a packet trace for the connection.

## 1.7 ADDITIONAL **ftp** COMMANDS

The DOMAIN File Transfer Program (**ftp**), which allows you to transfer files between your DOMAIN network and a remote host, has added new commands and new functions to existing commands. Table 4 lists the additions and changes to **ftp** commands. The following sections describe these commands in more detail.

**TABLE 4. Additional ftp Commands**

| Command | Description | Command | Description |
|---|---|---|---|
| mget/mrecv/mretrieve | Copy remote files to local host | mkdir | Make directory on remote host |
| mput/msend/mstore | Copy local files to remote host | pwd | Display remote working directory |
| put | Same as **ftp store** and **send** command | rmdir | Delete directory on remote host |
| recv | Same as **ftp get, retrieve** command | wd/cd | List local working directory |
| mdelete | Delete multiple files on remote host | debug | Toggle debugging mode |
| cup | Move to remote parent directory | remotehelp | Display remote host available commands |
| ld | Display local working directory | verbose | Toggle verbose mode |
| acct | Same as **ftp account** command. | command/cmd | Create a Shell process on the remote host. |
| trace | Toggle trace mode to generate packet traces. | | |

### 1.7.1 THE mget, mrecv, mretrieve and COMMANDS

---

mget/mrecv/mretrieve -- Copy remote files to local host

---

**FORMAT**

**mget [OR mrecv OR mretrieve] remote_files -option**

With the **mget** command (and its synonyms **mrecv** and **mretrieve**), you can use wildcards when copying remote files to a local host. You can copy the files using one of the following options on the command line:

| Option | Action |
|---|---|
| **-q** | Query for pathname of destination |
| **-s** | Destination pathname same as source pathname |
| **-w** | Source file copied to current working directory |

Note that you cannot copy nil files (for example, directories).

**ARGUMENTS**

**remote_files**

Specify the names of the files to be copied to the local host. If you omit the argument on the command line, the command will prompt you for it.

**EXAMPLE**

```
> mretrieve /src/a_* -w

  getting list of names from server....
  200 Host 192.9.10.96, port 2263 125 List started ok
  250 List completed
  retrieve "/src/a_dest" as "a_dest"
  200 Host 192.9.10.96, port 2366
  150 Retrieval of "/src/a_dest" started
  226 Transfer of "/src/a_dest" completed
  retrieve "/src/a_dst2" as "a_dst2"
  200 Host 192.9.10.96, port 2889
  150 Retrieval of "/src/a_dst2" started
  226 Transfer of "/src/a_dst2" completed

> ld

  Directory "/yoyo":
  a_dest a_dst2
```

```
  2 entries
```

If you don't supply the remote files, the command will prompt for them.

```
> mretrieve

  remote file-group descriptor: /src/a_*
  flags <-q/-s/-w>: -w
  .
  .
  .
```

### 1.7.2 THE mput, msend, and mstore COMMANDS

```
-------------------------------------------------------------------
mput/msend/mstore -- Copy local files to a remote host
-------------------------------------------------------------------
```

**FORMAT**

**mput [OR msend OR mstore] local_files -option**

With the **mput** command (and its synonyms **msend** and **mstore**), you can use wildcards when copying local files to a remote host. You can copy the files using one of the following options on the command line:

| Option | Action |
|---|---|
| **-q** | Query for pathname of destination |
| **-s** | Destination pathname same as source pathname |
| **-w** | Source file copied to current working directory |

Note that you cannot copy nil files (for example, directories).

**ARGUMENTS**

**local_files**

Specifies the names of the files to be copied to the local host. If you omit the argument on the command line, the command will prompt you for it.

**EXAMPLE**

```
> mput /y2/*dst* -s

  200 OK
  200 Host 192.9.10.96, port 3094
  125 Storing "a_dst2"
```

```
  226 File transfer completed 200 OK
  200 Host 192.9.10.96, port 3430
  125 Storing "a_dst2_a"
  226 File transfer completed >
```

### 1.7.3 THE put COMMAND

The **ftp put** command is the same as the **ftp** commands **store** and **send.** The command allows you to send a local file to a remote host. For a complete description, see the section on **store** and **send** in the manual, Using telnet and ftp.

### 1.7.4 THE recv COMMAND

The **ftp recv** command is the same as the **ftp** commands **get** and **retrieve.** The command allows you to transfer a remote file to a local host. For a complete description, see the section on **retrieve** and **get** in the manual, Using telnet and ftp.

### 1.7.5 THE mdelete COMMAND

---

mdelete -- Delete multiple files on the remote host

---

**FORMAT**

**mdelete remote_files**

Use the **mdelete** command to delete several files using wildcards. The command queries you before it deletes a file. Supply the following responses to each query:

| Response | Action |
|---|---|
| **y** | Delete the file |
| **n** | Save the file |
| **q** | Stop **mdelete**, return to **ftp** '>' prompt |
| **g** | Delete files without further queries |

**ARGUMENTS**

**remote_files**

Specify the files you want to delete using DOMAIN wildcards.

**EXAMPLE**

> **mdelete /yoyo/a_***

```
  getting list of names from server....
  200 Host 192.9.10.96, port 2519
  125 List started ok
  250 List completed
  Delete file '/yoyo/a_dest' <y/n/g/q>? y
  250 "/yoyo/a!dest" deleted
  Delete file '/yoyo/a_dst2' <y/n/g/q>? n

>
```

### 1.7.6 THE cup COMMAND

---

cup -- Move to the parent directory on the remote host

---

**FORMAT**

**cup**

The **cup** command changes the working directory on the remote host to the parent directory.

### 1.7.7 THE ld COMMAND

---

ld -- Displays the local working directory

---

The **ld** command allows you to display the working directory on the local directory.

**FORMAT**

**ld**

### 1.7.8 THE mkdir COMMAND

---

mkdir -- Make a directory on the remote host

---

**FORMAT**

**mkdir pathname**

The **mkdir** command creates a new directory named **pathname** on the remote host.

**ARGUMENTS**

**pathname**

Specify the name of the directory you are creating.

### 1.7.9 THE pwd COMMAND

---

pwd -- Display the current remote working directory

---

**FORMAT**

**pwd**

The **pwd** command displays the name of the current working directory on the remote host.

### 1.7.10 THE rmdir COMMAND

---

rmdir -- Delete a directory on the remote host

---

**FORMAT**

**rmdir pathname**

The **rmdir** command deletes the directory named **pathname** on the remote host. The directory must be empty; you must have previously removed all the files with the **delete** or **mdelete** command before you can remove the directory.

**ARGUMENTS**

**pathname**

Specify the name of the directory you want to delete.

### 1.7.11 THE wd/cd COMMAND

---

wd/cd -- Changes the local working directory

---

**FORMAT**

**wd [OR cd] pathname**

The **wd** or **cd** command changes the working directory on the local host to the directory specified by pathname. If you don't specify a **pathname**, **wd** or **cd** displays the local working directory.

### 1.7.12 THE acct COMMAND

The **ftp acct** command is the same as the **ftp account** command. This command allows you to send your account number to the remote host. For a complete description, see the section on **account** in the manual, Using telnet and ftp.

### 1.7.13 THE command/cmd COMMAND

---

command/cmd -- Changes the local working directory

---

**FORMAT**

**command [OR cmd] [shell_command [command_args]]**

The **command** or **cmd** command invokes a DOMAIN Shell process and passes commands to this process rather than the executing process.

If you enter a **shell_command** immediately after **command,** the Shell process exists after processing the command. If you enter the **command** without specifying a **shell_command,** the DOMAIN Shell prompt ($) appears, and you can enter any number of Shell commands. You exit the Shell process by entering a EOF or CTRL/Z command.

Note that connections to the remote host remain open while running the Shell process.

**ARGUMENTS**

**shell_command**

Enter the DOMAIN Shell command that you want to run on the local machine. Default if omitted: DOMAIN Shell prompt appears and the Shell will process commands up to an EOF or CTRL/Z.

**command_args**

Specify any arguments required to run **shell_command.**

### 1.7.14 THE debug COMMAND

---

debug -- Toggle debugging mode

---

**FORMAT**

**debug**

The **debug** command turns debugging on and off. When debugging is on, **ftp** prints each command that it transmits to the remote **ftp** server, preceded by the string "**-->**". By default, debug mode is off.

### 1.7.15 THE remotehelp COMMAND

---

remotehelp -- Display available commands on remote host

---

**FORMAT**

**remotehelp [command]**

Use this command to see a description of the **ftp** commands available on the remote host. If you want information about a specific command, type the name of the command on the command line. You will receive information about that command, if it's available on the remote host.

**ARGUMENTS**

**command**

Specify the **ftp** command you want information about. If you omit this, **remotehelp** displays a list of all available **ftp** commands on the remote host.

### 1.7.16 THE trace COMMAND

---
trace -- Toggle trace mode
---

**FORMAT**

**trace**

The **trace** command turns tracing on and off. When tracing is on, **ftp** generates a packet trace for the connection. By default, trace mode is off.

### 1.7.17 THE verbose COMMAND

---
verbose -- Toggle verbose mode
---

**FORMAT**

**verbose**

When you run in **verbose** mode, the remote **ftp** server displays all its responses on the local host. It also displays statistics regarding the efficiency of data transfer. By default, **verbose** is on.

### 1.7.18 CHANGE TO THE list and nlst COMMANDS

You can now direct a listing to standard output with the commands **list** and **nlst** by supplying a hyphen ('-') rather than a pathname as a second argument. **list** displays information about a remote file (specified in the third argument), and **nlst** displays information about a list of remote filenames.

For more information, see the manual, Using telnet and ftp.

# CHAPTER 2

# SOFTWARE INSTALLATION PROCEDURES

## INSTALLATION PROCEDURE

This chapter describes how to install TCP/IP Version 3.0. You can add this software to a user node (one equipped with monitor and keyboard) or a DOMAIN server processor (DSP) that is running SR9.5 of the AEGIS or DOMAIN/IX operating system. If the user node or DSP is not running SR9.5 or a more recent version, follow the appropriate software update procedures as described in Installing DOMAIN Software (008860) or in the appropriate release notes.

Prior to this release, DOMAIN TCP/IP provided driver software that enabled TCP/IP to run over ETHERNET gateways. This TCP/IP does not provide driver software. Instead, you must first install the appropriate driver before installing or updating TCP/IP. You can refer to the following release documents for installing these drivers:

- EtherController-AT Release Document (009742) for installing an ETHERNET controller in DN3000 workstations and servers.
- EtherController-MB Release Document (009743) for installing an ETHERNET controller in workstations and servers that contain the MULTIBUS peripheral expansion cage.

If you have both the DOMAIN/IX and TCP/IP products, you must install DOMAIN/IX Version 9.5 before you install TCP/IP. See the Release Document for DOMAIN/IX Version 9.5 for instructions on how to install this software.

NOTE: The user node or DSP must have a minimum of 800 blocks of available disk space for a successful installation of this software.

NOTE: If you're running TCP/IP as a gateway on a DOMAIN node containing the EtherBridge product, you must update all the nodes running TCP/IP on the network to Version 3.0 if you want the nodes to communicate through the gateway. For details on using EtherBridge and TCP/IP, see the EtherBridge Version 1.3 Release Notes.

NOTE: TCP/IP Version 3.0 is not compatible with SR9.2 or lower versions software. Therefore, nodes running SR9.2 and lower software cannot use the gateway. For details on compatibility issues with SR9.5, refer to the Release Notes document with the standard DOMAIN or DOMAIN/IX SR9.5 software.

## 2.1 Conventions and Terminology

Before you start, make sure you understand these terms and conventions:

Work Node — The user node at which you perform the installation procedure.

Target — The directory into which you're installing software. The target can be a node entry directory (for example, //TARGET) or any subdirectory (for example, //TARGET/PRODUCT). If the target is on a user node, then the work node and the user node can be the same node.

NOTE: When you are installing software to update a diskless node, the target is the node entry directory of the partner node.

Secure network — A network that uses a registry of user accounts and access control lists (ACLs) to control log-in privileges and access to files and directories. Note that an open network does not use a registry or ACLs.

Source area — An on-line master area of DOMAIN software. An administrator installs software from the distribution media into the source area and users install software from the source area over the network. The source area can be a node's entry directory or any subdirectory.

Source media — The media (floppy disks, magnetic tape, cartridge tape, or another node in the network) that contains the software.

< > — Angle brackets ( < > ) enclose the name of a key on the keyboard.

## 2.2 Planning the Installation

There is one installation procedure. You can use the procedure in one of three modes: ADMINISTRATIVE mode, USER mode, or SPECIAL-CASE mode.

ADMINISTRATIVE mode creates a source area by copying the INSTALL program and the new software from the distribution media to the target. Use the

administrative mode to provide a source area for this release. See Section 2.3 for more information about administrative-mode installations.

USER mode involves copying your new software from a source area onto another node in the network; it's the simplest and most commonly used mode. You can install in user mode only AFTER an administrative-mode installation has initialized the source area with the INIT_SOURCE program.

Two default conditions apply to a user-mode installation. The defaults are:

- The INSTALL program automatically copies the new software over the network from the initialized source area, instead of asking you to specify the source area.

- The INSTALL program uses the SID "user.sys_admin" during the installation, rather than your own login SID.

To install in user mode, get the source area's pathname from your system administrator, then go on to Section 2.4.2.

SPECIAL-CASE mode involves special cases in which you need to override the user-mode defaults. The special cases are:

- You want to install software from an initialized source area on the network, but your own login SID gives you more rights to a target's protected directories than the default SID "user.sys_admin"

- You want to install software from a source other than an initialized source area (for example, source media)

- You want to install additional software in a source area that was initialized during a previous administrative-mode installation

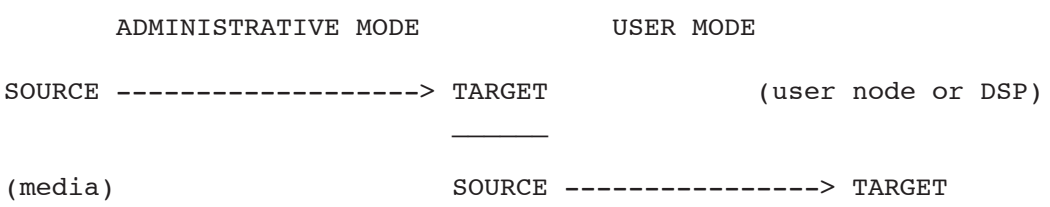
The installation procedure enters special-case mode when you invoke the INSTALL program with its -my_sid option. Specifying this option overrides the user-mode defaults, which means that the INSTALL program (1) uses your own login SID instead of "user.sys_admin" and (2) prompts for source media rather than automatically copying the software over the network from an initialized source area. In all other respects, special-case mode behaves like one of the other two modes of installation (your choice of source determines which one).

If you want to install software from an initialized source area on the network while using your own login SID, follow the directions for user-mode installations in Section 2.4.2. These directions include provisions for installing software in special-case mode.

If you want to install software from a source other than an initialized source area, or you want to install additional software in a previously initialized source area, follow the directions for an administrative-mode installation in Section 2.4.1. These directions also include provisions for installing software in special-case mode.

## 2.3 More Information About Administrative-mode Installations

The target of an administrative-mode installation generally serves as the source for subsequent user-mode installations (the administrative-mode target pathname is therefore the same as the user-mode source pathname). User-mode installations use both the INSTALL program and the software stored in the source area.




Your choice of target for an administrative-mode installation depends on whether you want the target node to RUN the software as well as act as a SOURCE for the software. If you also want the node to run the software, make the target the node's entry directory (for example, //node). If you just want the node to contain the software, you should make the target a subdirectory (for example, //node/product/source_area). In either case, users should then use the target of your administrative-mode installation as their source area.

You can install different optional software products into the same source area or into separate source areas. Whichever route you take, you can then selectively install optional products on user nodes or DSPs from the source area(s).

If you have a secure network, you must have system administrator rights to install in administrative mode. Also, during the procedure you must initialize the source area by running the INIT_SOURCE program. This program marks the installation program in the source area with special privileges for subsequent user-mode installations, such as use of the SID "user.sys_admin" during installation. The INSTALL program can then install software in protected system directories, even though the user running the program does not have rights to modify these directories. In open networks, you create a source area but you don't run INIT_SOURCE, since all users have rights to modify their system directories.

## 2.4 Installation Procedures

The following sections describe the administrative mode and the user mode of installation. To install software in special-case mode, consult Section 3.0 to determine which set of instructions you should follow.

### 2.4.1 Administrative Mode

NOTE: You can enter a CTRL/Q sequence at any prompt in the INSTALL program to abort the installation and return to the Shell.

1. If you intend to create a source area for future installations, log on to a work node using a system administrator account (for example, my_name.sys_admin.%.%). Otherwise, log on using your own account (for example, my_name.%.%.%) .

2. Set your working directory to the installation target. This target will become the source area for user installations. It can be a node entry directory (like //node) or it can be any subdirectory created prior to the installation (like //node/product/source_area) For example:

   $ **wd //node**

3. Insert the source media into the drive and enter the RBAK command shown below. If you are using a tape cartridge, use the CT option shown in the example. If you are using a magnetic tape, use the M0 (Mzero) option. If you are using a floppy disk, use the F0 (Fzero) option.

   NOTE: TCP/IP comes on multiple floppy disks. Insert the floppy disk with the numeral 1 at the end of its label (for example, "FLP8_TCP_3.0_1"). You will perform this step only with the first floppy disk.

   All of the RBAK commands shown below create an INSTALL directory on the target and write the installation software to the directory. When entering the RBAK command, use lower-case characters to ensure visibility of the install directory in case-sensitive environments. Note that you can leave the source media in the drive for use in a later step; if you remove the source media after executing the RBAK command, the INSTALL program will later prompt you to re-insert the media.

   **rbak -dev ct -f 1 install -as install -l -ms -force -sacl -du**

   **rbak -dev m0 -f 1 install -as install -l -ms -force -sacl -du**

   **rbak -dev f0 -f 1 install -as install -l -ms -force -sacl -du**

4. Set your working directory to the INSTALL directory on the target. For example:

   $ **wd //node/install**

5. Execute the INSTALL program and follow the prompts. If you are installing software in special-case mode, use the -my_sid option.

   For ADMINISTRATIVE MODE, type:

   $ **install**

   For SPECIAL-CASE MODE, type:

   $ **install -my_sid**

6. The program may prompt you to enter an installation type, based on what products already exist in the source area. If it does, answer OPT and proceed. For example:

```
**************************************
* SOFTWARE INSTALLATION -- Version n.n *
**************************************


Software installation TYPES are:

STD      --    Install standard software
RESTART  --    Restart the software installation
ACL      --    Set ACLs for existing software
CLEANUP  --    Run the Cleanup Procedure for ADD MODE installations
OPT      --    Install optional software (e.g., Pascal, FORTRAN)

Please enter installation TYPE: OPT
```

7. When the program displays the names of one or more optional products, enter the name of the optional product that you want to install. For example, to install TCP/IP, type “TCP”, as shown in the sample menu below.

```
Name     Description                    Disk Blocks Needed
                                       (Adding New Software)

TCP      TCP/IP                           800

OTHER    If the optional product that you would like to install is not
         listed above, choose OTHER.
         *Note: When you choose OTHER, you are asked a few questions
         then shown a display of Apollo's optional products. Check
         with your system administrator to determine which products
         your site has purchased and in which directory these products
         have been installed.

Enter the name of a single product you would like to install: TCP
```

8. When prompted for the name of the target, enter the appropriate pathname (that is, the node entry directory or subdirectory that you specified in Step 2). Note that you must install TCP in the node entry directory. For example:

```
The TARGET is the node or subdirectory on which you are installing
software. (e.g., '//my_node' or '//my_node/subdirectory')
Enter Target: //node
```

9. Indicate whether you are installing software on a gateway to another network. Answer YES if you are installing software on any of the following:

    - On a gateway to another network
    - On a bridge to a DOMAIN network
    - In a source area that will contain all TCP/IP software

    ```
    Is the installation to //TARGET a GATEWAY install?
    Enter YES or Y or NO or N :
    ```

10. Indicate whether you are installing software for a diskless node.

    ```
    Is the installation to //TARGET a DISKLESS install?
    Enter YES or Y or NO or N :
    ```

    If you specify YES, the procedure asks you for the ID of the diskless node. Enter the hexadecimal node ID.

    ```
    Please enter the NODE ID of the TARGET DISKLESS NODE that the tcp
    software will be installed for. (e.g., 260d):
    Enter hex node id:
    ```

11. Indicate whether this is a TCP/IP administrative node that will maintain a copy of the host mapping files. If so, answer YES. Also answer YES if you are creating a source area that will contain all TCP/IP software.

    ```
    Is the installation to //TARGET an TCP/IP ADMINISTRATOR install?
    Enter YES or Y or NO or N :
    ```

12. The following question appears if you did NOT specify a GATEWAY installation in step 9. Indicate whether this node is a client, that is a TCP/IP host.

    ```
    Is the installation to //TARGET a CLIENT install?
    Enter YES or Y or NO or N :
    ```

13. If you did NOT specify an administrative installation in step 11, the INSTALL program asks for the TCP/IP administrative node's name.

```
Please enter the name of the TCP/IP ADMINISTRATOR node on which
the sys/tcp/hostmap DIRECTORY resides (e.g. //SERVER)

Enter node name or type CTRL/Q to quit:
```

14. The INSTALL program prompts for the source media. Enter your choice.

```
Source MEDIA is one of:
     CTAPE  --  Cartridge Tape
     MTAPE  --  Magnetic Tape
     FLOPPY --  8" or 5 1/4" Floppies
     NET    --  An area on the network with valid Software
Enter Source Media:
```

15. The INSTALL program may ask you to insert the media into the drive. Insert the media and press <RETURN>.

16. The INSTALL program installs the software, listing each file it copies from the source media. Since the software resides on multiple floppy disks, the program prompts you to mount (that is, insert) the next disk and to press <RETURN> to continue.

    When the INSTALL program finishes installing the software, it displays the following menu:

```
Options:

     RERUN    --  There were errors in the transcript
                  pad and you wish to rerun the installation.

     FINISH   --  The installation ran to completion error free.
                  There is no additional optional software you
                  wish to install.

     CONTINUE -- Install additional optional software.
```

If you encountered any errors during the installation, correct the problem(s) and select RERUN. To locate error messages issued during installation, search backwards for the characters @? (an at sign followed by a question mark) in the installation's transcript pad.

If there were no errors, choose CONTINUE or FINISH. Selecting CONTINUE brings you back to the beginning of the INSTALL program; selecting FINISH terminates the program. If you were installing software from magnetic tape, cartridge tape, or floppy disks, you can now remove the media from the drive.

17. If you have a secure network and you want the target of your installation to be used as a source area for future installations, run the INIT_SOURCE program (also run the program if you are adding software to a previously initialized source area). You must be logged in as a system administrator to perform this step.

    Invoke INIT_SOURCE at the Shell prompt. When prompted, enter the pathname of the new source area (which is currently the target of your administrative-mode installation). Here is an example:

    ```
    $ init source

    Please enter the name of the SOURCE AREA
    for your network (e.g., '//NODE/SOURCE AREA'):
    //node

    The source area for your network
    has been set to: //node
    ```

18. Perform this step only after completing an error-free installation and selecting FINISH.

    a. Use the Display Manager SHUT command to shut down the target node.

    ```
    <CMD> SHUT <RETURN>
    ```

    b. After the SUCCESSFUL SHUTDOWN message and the > prompt appear, reboot the node by typing the following at the prompt:

    ```
    > RE <RETURN>
    > <RETURN>
    MD REV xx/xx/xx
    > EX AEGIS <RETURN>
    ```

This is the end of the administrative-mode installation procedure. Refer to Configuring and Managing TCP/IP for information about configuring your system.

### 2.4.2 User Mode

NOTE: You can enter a CTRL/Q at any prompt in the INSTALL program to abort the installation and return to the Shell.

Follow this procedure if you are installing TCP/IP software on a node from the network.

1. Log on to a work node using your own account (for example, my_name.%.%.%).

2. Set your working directory to the INSTALL directory in the source area (if necessary, ask your system administrator for the pathname). For example:

   $ **wd //node/install**

3. Execute the INSTALL program and follow the prompts. If you are installing software in special-case mode, use the -my_sid option.

   For USER MODE, type:

   $ **install**

   For SPECIAL-CASE MODE, type:

   $ **install -my_sid**

4. The program may prompt you to enter an installation type, based on what products already exist in the source area. If it does, answer OPT and proceed. For example:

```
**************************************
* SOFTWARE INSTALLATION -- Version n.n *
**************************************


Software installation TYPES are:

STD      --    Install standard software
RESTART  --    Restart the software installation
ACL      --    Set ACLs for existing software
CLEANUP  --    Run the Cleanup Procedure for ADD MODE installations
OPT      --    Install optional software (e.g., Pascal, FORTRAN)

Please enter installation TYPE: OPT
```

5. When the program displays the names of one or more optional products, enter the name of the optional product that you want to install. For example, to install TCP/IP, type "TCP", as shown in the sample menu below.

```
Name     Description                    Disk Blocks Needed
                                       (Adding New Software)

TCP      TCP/IP                           800

OTHER    If the optional product that you would like to install is not
         listed above, choose OTHER.
         *Note: When you choose OTHER, you are asked a few questions
         then shown a display of Apollo's optional products. Check
         with your system administrator to determine which products
         your site has purchased and in which directory these products
         have been installed.

Enter the name of a single product you would like to install: TCP
```

6. When prompted for the name of the target, enter the appropriate pathname (that is, the node entry directory or subdirectory that you specified in Step 2). Note that you must install TCP in the node entry directory. For example:

```
The TARGET is the node or subdirectory on which you are installing
software. (e.g., '//my_node' or '//my_node/subdirectory')
Enter Target: //node
```

7. Indicate whether you are installing software on a gateway to another network. Answer YES if you are installing software on any of the following:

```
Is the installation to //TARGET a GATEWAY install?
Enter YES or Y or NO or N :
```

8. Indicate whether you are installing software for a diskless node.

```
Is the installation to //TARGET a DISKLESS install?
Enter YES or Y or NO or N :
```

If you specify YES, the procedure asks you for the ID of the diskless node. Enter the hexadecimal node ID.

```
Please enter the NODE ID of the TARGET DISKLESS NODE that the tcp
software will be installed for. (e.g., 260d):
Enter hex node id:
```

9. Indicate whether this is a TCP/IP administrative node that will maintain a copy of the host mapping files. If so, answer YES. Also answer YES if you are creating a source area that will contain all TCP/IP software.

```
Is the installation to //TARGET an TCP/IP ADMINISTRATOR install?
Enter YES or Y or NO or N :
```

10. The following question appears if you did NOT specify a GATEWAY installation in step 9. Indicate whether this node is a client, that is a TCP/IP host.

```
Is the installation to //TARGET a CLIENT install?
Enter YES or Y or NO or N :
```

12. If you did NOT specify an administrative installation, the INSTALL program asks for the TCP/IP administrative node's name.

```
Please enter the name of the TCP/IP ADMINISTRATOR node on which
the sys/tcp/hostmap DIRECTORY resides (e.g. //SERVER)

Enter node name or type CTRL/Q to quit:
```

12. The INSTALL program may prompt for the source area. If so, enter the pathname (if you don't know it, ask your system administrator) . For example:

    The SOURCE AREA is the node or subdirectory from which you are copying software. (e.g., '//node' or '//node/subdirectory')
    Enter Source Area: **//node**

13. The INSTALL program installs the software, listing the name of each file it copies from the source area. Upon completion, the INSTALL program displays the following menu:

    Options:

    RERUN -- There were errors in the transcript pad and you wish to rerun the installation.

    FINISH -- The installation ran to completion error free. There is no additional optional software you wish to install.

    CONTINUE -- Install additional optional software.

    If you encountered any errors during the installation, correct the problem(s) and select RERUN (if necessary, consult your system administrator for assistance). To locate error messages issued during installation, search backwards for the characters @? (an at sign followed by a question mark) in the installation's transcript pad.

    If there were no errors, choose CONTINUE or FINISH. Selecting CONTINUE brings you back to the beginning of the INSTALL program; selecting FINISH terminates the program.

14. Perform this step only after completing an error-free installation and selecting FINISH.

    a. Use the Display Manager SHUT command to shut down the target node.

        <CMD> **SHUT** <RETURN>

b. After the SUCCESSFUL SHUTDOWN message and the> prompt appear, reboot the node by typing the following at the prompt:

```
> RE <RETURN>
> <RETURN>
MD REV xx/xx/xx
> EX AEGIS <RETURN>
```

This is the end of the user-mode installation procedure. Refer to the Configuring and Managing TCP/IP manual for information about configuring your system.

## 2.5 TCP/IP Files

The following sections list the files and links that are installed during a Client, Server, and Gateway installation.

### 2.5.1 Client Files

The following files are installed on a client node:

com/ftp

com/host

com/tcpstat

com/telnet

doc/tcp.release_notes

sys/tcp/ftp_server

sys/tcp/makegate

sys/tcp/networks_template

sys/tcp/setroute

sys/tcp/tcpinit

sys/tcp/tcpreset

sys/tcp/tcp_server

sys/tcp/telnet_server

sys/tcp/thishost_template

systest/ssr_util/dtcb

systest/ssr_util/mbd

systest/ssr_util/sodebug

systest/ssr_util/trpt

### 2.5.2 Gateway Files

The following files are installed on a gateway node in addition to all client node files:

sys/tcp/maphost

sys/tcp/rip_server

### 2.5.3 Administrative Files

The following files are installed on the administrative node:

sys/tcp/hostmap/hashnic

sys/tcp/hostmap/hosts.txt_template

sys/tcp/hostmap/local.txt_template

sys/tcp/hostmap/makehdb

sys/tcp/hostmap/makehost.sh

sys/tcp/hostmap/ndb_format

sys/tcp/hostmap/sortnic

### 2.5.4 Links

The installation procedure creates the following links on client and gateway nodes. //ADMIN indicates the administrative node that you specify during the installation procedure. TARGET is the target area where the software is being installed.

| FROM | TO |
|---|---|
| TARGET/com/net | TARGET/com/hosts |
| TARGET/sys/tcp/gateways | //ADMIN/sys/tcp/gateways |
| TARGET/sys/tcp/hostmap | //ADMIN/sys/tcp/hostmap |
| TARGET/sys/tcp/hosts.hst | //ADMIN/sys/tcp/hosts.hst |
| TARGET/sys/tcp/networks | `node_data/networks |
| TARGET/syS/tcp/thishost | `node_data/thishost |

# CHAPTER 3

# CHANGES IN DOCUMENTATION

The TCP/IP document set consists of the following manuals:

- Configuring and Managing TCP/IP (008543)
- Using telnet and ftp (008667)

The new features in TCP/IP Version 3.0 have been added to the revised manual, Configuring and Managing TCP/IP (008543). This manual was previously called Managing TCP/IP-Based Communications Products.

Configuring and Managing TCP/IP describes how to configure, manage, and troubleshoot DOMAIN and DOMAIN/IX BSD4.2 TCP/IP.

Using telnet and ftp describes how to use two common TCP/IP utilities: the TELNET remote terminal emulator and the FTP file transfer program. This book describes both the DOMAIN and DOMAIN/IX versions of these utilities.

In addition, the manual, System Administration for DOMAIN/IX BSD4.2 (009355), contains a subset of information contained in the Configuring and Managing TCP/IP that is relevant to BSD4.2 TCP/IP.

# CHAPTER 4

# BUGS AND BUG FIXES

## 4.1 BUGS IN RELEASE 3.0

This section documents known bugs in the TCP/IP documentation, software installation procedures, and software.

### 4.1.1 BUGS IN DOCUMENTATION

TCP/IP Version 3.0 comes with a revised manual of Configuring and Managing TCP/IP. Currently, there are no known bugs reported for the documentation.

### 4.1.2 BUGS IN TCP/IP SOFTWARE

The following bugs currently exist in the TCP/IP software:

- If you send two **telnet ip** (**^ip**) commands in a row, with no intervening input, the second one is received by the foreign host as the character 't'.
- The **telnet ^S** sequence and **^Q** sequence do not work well because DOMAIN TCP/IP allows the remote system to transmit up to 8K bytes of data at a time for performance reasons.

## 4.2 RESTRICTIONS IN TCP/IP SOFTWARE

The TCP/IP **ftp** and **telnet** commands are case-sensitive. You must use lowercase letters for these commands.

TCP/IP transmits broadcasts using a host address 0. It does, however, recognize broadcasts from other hosts using broadcast addresses of 0 or -1.

Prior to this release, DOMAIN TCP/IP provided driver software that enabled TCP/IP to run over ETHERNET gateways. This TCP/IP does not provide driver software. Instead, you must first install the appropriate driver before installing or updating TCP/IP. You can refer to the following release documents for installing these drivers:

- EtherController-AT Release Document (009742) for installing an ETHERNET controller in DN3000 workstations and servers.

- EtherController-MB Release Document (009743) for installing an ETHERNET controller in workstations and servers that contain the MULTIBUS peripheral expansion cage.

## 4.3 BUG FIXES SINCE RELEASE 2.1

The following TCP/IP bugs have been corrected since TCP/IP Version 2.1:

- Version 3.0 supports Trailer Encapsulations as defined by Request for Comment (RFC) 893, so you can communicate with TCP/IP implementations that support trailers. This version corrects a problem with trailers that occurred in Version 2.1.

- Prior to this release, the **telnet_server** and **ftp_server** would try to initialize before the **tcp_server** started running and so the servers failed to start. Version 3.0 corrects this problem. The **telnet_server** and **ftp_server** now wait until **tcp_server** is running before they start initializing.

- The **telnet_server** now negotiates echo options correctly.

- Version 3.0 fixes a problem with the **telnet_server** that caused it to hang if you tried to log in to a DOMAIN TCP/IP node, but exited before you completed the log in procedure.

- Prior to this release, if you passed a bad data buffer address to a get or put operation, the **tcp_server** would hang and become unusable. Version 3.0 corrects this problem.

- **telnet** now exits (correctly) when the other side resets the connection.

- If you logged in from a remote host to a DOMAIN node running DOMAIN **telnet** and then tried to run in a C shell, the C shell would hang. Version 3.0 corrects this.

- If you logged in to a remote system using **telnet**, and the other side sent null characters, the null characters would show up as triangles on our side. Version 3.0 now ignores null characters, so you won't see any triangles.

- Version 3.0 can now handle TCP/IP windows larger than 32K bytes.

- The **rip_server** would sometimes stop working properly after it had been running for a while. That is, it would be running, but wouldn't do anything. Version 3.0 corrects this.

- Conditional put operations of 4K bytes or more wouldn't work at times. Version 3.0 corrects this.

- For BSD4.2 TCP/IP users, you can now specify a backlog of 0 in a call to **listen()**. This is equivalent to setting a backlog of 1.