Convergent Technologies

RELEASE NOTICE
for
3.2 CTIX TCP/IP
(SNC-20Ex, SNC-20Fx)

Revised 20 December 1989

Trademark Notice

Convergent Technologies, NGEN, and S/Series systems are registered trademarks of Convergent Technologies, Inc.

WGS/Office, WGS/Desktop Manager, WGS/Word Processor, WGS/Spreadsheet, WGS/Mail, WGS/Calendar, Workgroup Solutions, PC Exchange, S/640, S/320, S/222, S/221, S/220, S/120, MiniFrame, AWS, IWS, Server PC, PT, GT, CTIX, CTIX/386 and CTOS are trademarks of Convergent Technologies, Inc.

CTIX and CTIX/386 are derived from UNIX System V software, under license from AT&T. UNIX is a trademark of AT&T.
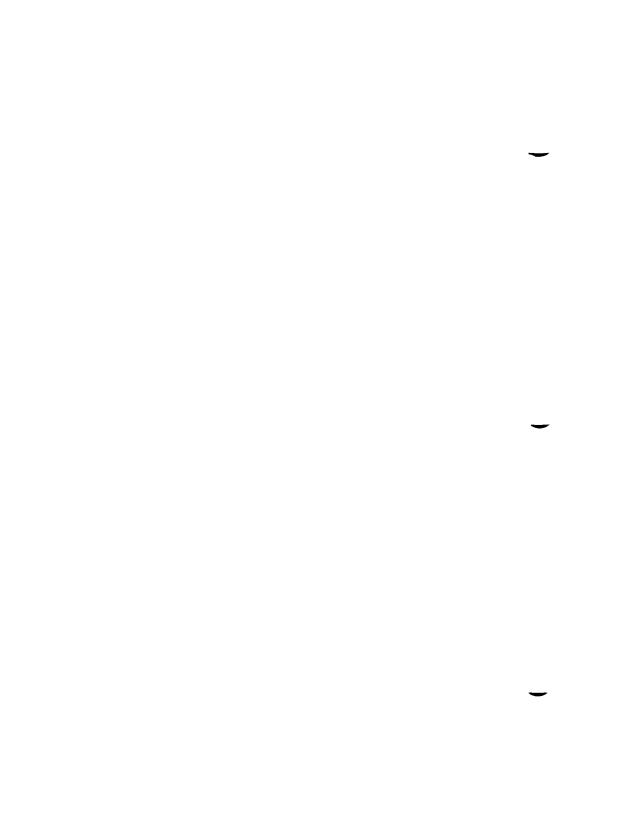
TABLE OF CONTENTS

1. Contents of the Release Package

The contents of CTIX TCP/IP 3.2 package consist of the following:

For S/80 through S/320
Release Notice   (Part Number B-09-01453-01-C)
Quarter-Inch Tape   (Part Number 71-03336-01)

and the following media:

For S/480 through S/640
Release Notice   (Part Number B-09-01453-01-C)
Quarter-Inch Tape   (Part Number 71-03360-01)

This product is documented by:

- *CTIX Network Administrators Guide* (DNC-180)

2. General Description of the Release

This release notice describes CTIX TCP/IP as provided at release level 3.2.

CTIX TCP/IP 3.2 is a general networking product based on the Department of Defense Internet protocol suite. It allows the interconnection of machines on one or more networks based on either IEEE 802.3 Ethernet and/or asynchronous RS-232 links.

CTIX TCP/IP provides high performance file transfer, remote terminal, and mail in a local area network environment. It provides the same services transparently through wide area leased line networks and public data networks (PDNs).

Device related software required by CTIX TCP/IP 3.2 is included in CTIX 6.2. Ethernet, public X.25, and asynchronous serial lines are currently supported.

3. Changes from Previous Releases

3.1 Changes from CTIX TCP/IP 3.1

Performance enhancements based on Van Jacobson's slow start algorithm, congestion window avoidance and Phil Karn's retransmit timer calculation have been implemented.

Direct SLIP and Switched SLIP now support streams tty drivers in CTIX 6.2. However, a non-streams version of SLIP is also supported in CTIX TCP/IP 3.2 as riop terminals still use non-streams version of tty drivers.

Security enhancements have been made to *rcp*. Shadow password support exists in all the relevant network commands.

Includes all the functionality of previous releases with bug fixes in some cases.

3.2 Changes in CTIX TCP/IP 3.1 from CTIX TCP/IP 3.0

CTIX TCP/IP 3.1 includes the *ctrlogin* facility, which has provisions for preserving terminal escape code sequences sent over the network.

Security enhancements have been made to *fingerd* and *ftpd* as a result of the November 1988 Internet virus (worm) attack.

Some files included in both CTIX TCP/IP 3.1 and CTIX 6.1 have been removed.

3.3 Changes in CTIX TCP/IP 3.0 from CTIX TCP/IP 2.3

CTIX TCP/IP 3.0 implements the same functionality as CTIX TCP/IP 2.3. The main purpose of this release is to support UNIX System V.3 Streams.

In order to maintain compatibility with existing 4.2 BSD machines, the default Internet broadcast address in CTIX TCP/IP 2.3 was set to 0's whereas most 4.3 BSD implementations use 1's. With CTIX TCP/IP 3.2, the default broadcast is the 4.3 BSD (and Internet) standard 1's. Other

broadcast formats may be specified with *ifconfig* as explained in the manual page.

3.4 SPR's Fixed in this Release

1. SPR# - 14422. Trailers are turned off in /etc/enpstart.

2. SPR# - 14709. The *ftp* mget command with globbing enabled, the pathname list from the server may be separated by NL or by CR/NL.

3. SPR# - 15342. Slip running between s/80 and s/120 at 9600 does a lot of retransmitions. Fixed with the new streams Slip and Van Jacbson TCP/IP.

3.5 Functionality in CTIX TCP/IP 3.1

All these functionalities are listed by the release numbers in which they were first introduced.

3.5.1 ctrlogin

The *ctrlogin* facility is comprised of the *ctrlogin* client, the *ctrlogind* daemon, and *ctrcmd*, which is a version of *rcmd* modified to call *ctrlogin* instead of *rlogin* when called with no arguments.

ctrlogin uses the VMIN and VTIME facilities of the terminal driver to detect terminal escape sequences (function keys). It then uses a special protocol to send these over the network to *ctrlogind*, where they are passed to the application in a such a way that the VMIN and VTIME facilities mimic a hardwired terminal.

3.6 Functionality in CTIX TCP/IP 3.0 and 2.3

3.6.1 STREAMS support

CTIX TCP/IP 3.2 is implemented within the framework of AT&T System V.3 STREAMS. This flexible network architecture allows easy configuration of networking services, as well as support of AT&T Remote File Sharing (RFS). (RFS is a separate product) The *slink* daemon and the */etc/netcf* configuration file are used to configure STREAMS.

3.6.2 Routing Daemon

CTIX TCP/IP 3.2 includes *routed*, which implements the Routing Information Protocol. This protocol allows automatic configuration of the routing tables which tell the system which gateways to send traffic for distant networks to.

3.6.3 Switched SLIP

Modems, PBX's and other switched serial media are supported by CTIX TCP/IP 3.2 via *slipd*. This allows dial-up connections to be specified in a manner similar to *uucp*. Connections are established when traffic for a given destination exists.

3.6.4 Internet Super-Server

Internet services are supported in CTIX TCP/IP 3.2 via *inetd*. This allows easy configuration of services via a single file (*/etc/inetd.conf*), and reduces the number of processes on the system.

3.6.5 Diagnostic Utilities

CTIX TCP/IP 3.2 includes *ping*, which is used to check connectivity, and *arp*, which is used to check mappings of Ethernet addresses to Internet addresses.

3.6.6 Other Utilities

CTIX TCP/IP 3.2 includes *finger*, to provide user information, and *talk*, for online conversations.

4. Product Dependencies

4.1 Software Dependencies

The software is designed to work with the minimum revision levels listed below:

Software	Mktg Number	Minimum Revision
S/80 CTIX	SAC-20Bx	6.2
S/280 CTIX	SAC-2x14x	6.2
S/120-320 CTIX	SAC-20Cx	6.2
S/480 CTIX	SAC-20Fx	6.2
S/640 CTIX	SAC-20Fx	6.2

4.2 Hardware Dependencies

CTIX TCP/IP 3.2 runs on the entire S/Series systems line.

CTIX TCP/IP is designed to work with the minimum hardware revision levels listed below:

Board	Part Number	Minimum Revision
Ethernet/RS232	B-60-00436	G
VME Ethernet	B-60-00382	A
RS232 10-port	B-60-00357	C
RS232 20-port	B-60-00368	C
IO Processor	B-60-00367	E
VME Interface	B-60-00356	L

4.3 Memory Requirements

The product will run reasonably well in 250KB of memory (not including CTIX). As the number of users is increased, memory must be expanded (see section 7.4, *Increasing Memory Allocation*).

4.4 Disk Space Requirements

Disk space required for installation: 1200 root blocks, 1600 /usr blocks.

Disk space required for use of the product (temporary files, etc): *rwhod* creates some data files in **/usr/spool/rwho.** Other utilities produce files under user control.

5. Contents of the Distribution Medium

The distribution medium is your master copy. Store the master safely.

5.1 Quarter-Inch Tape Distribution

The CTIX TCP/IP (Part Numbers 71-03336-01 and QB) are released on 1 Quarter-Inch Tape, which contains the files to install the product on a fixed disk. All the files with **boldface** are new to this release.

The release medium contains the following tape files:

Tape File   Description

0-4	Reserved
5	installation tools
6	installation control files for TCPIP group
7	customizable files for TCPIP group
8	required files for TCPIP group

./dev
./dev/inet
./dev/inet/arp
./dev/inet/ip
./dev/inet/tcp
./dev/inet/udp
./dev/inet/icmp
./dev/ttyp16
./dev/ttyp17
./dev/ttyp18
./dev/ttyp19
./dev/ttyp20
./dev/ttyp21
./dev/ttyp22
./dev/ttyp23
./dev/ttyp24
./dev/ttyp25
./dev/ttyp26
./dev/ttyp27
./dev/ttyp28
./dev/ttyp29

```
./dev/ttyp30
./dev/ttyp31
./dev/vt16
./dev/vt17
./dev/vt18
./dev/vt19
./dev/vt20
./dev/vt21
./dev/vt22
./dev/vt23
./dev/vt24
./dev/vt25
./dev/vt26
./dev/vt27
./dev/vt28
./dev/vt29
./dev/vt30
./dev/vt31
./etc
./etc/lddrv
./etc/lddrv/arp.o
./etc/lddrv/ip.o
./etc/lddrv/ip.export
./etc/lddrv/tcp.o
./etc/lddrv/udp.o
./etc/lddrv/udp.export
./usr/lib/adman
./usr/lib/adman/netman
./usr/lib/adman/english_usa
./usr/lib/adman/english_usa/help
./usr/lib/adman/english_usa/help/NAddem.hlp
./usr/lib/adman/english_usa/help/NAddeq.hlp
./usr/lib/adman/english_usa/help/NAddmeq.hlp
./usr/lib/adman/english_usa/help/NAdmin.hlp
./usr/lib/adman/english_usa/help/NAhost.hlp
./usr/lib/adman/english_usa/help/NConn.hlp
./usr/lib/adman/english_usa/help/NInt.hlp
./usr/lib/adman/english_usa/help/NMem.hlp
./usr/lib/adman/english_usa/help/NNet.hlp
./usr/lib/adman/english_usa/help/NNst.hlp
./usr/lib/adman/english_usa/help/NProt.hlp
./usr/lib/adman/english_usa/help/NRout.hlp
```

./usr/lib/adman/english_usa/help/NRup.hlp
./usr/lib/adman/english_usa/help/NRwho.hlp
./usr/lib/adman/english_usa/help/NUsereq.hlp
./usr/lib/adman/english_usa/netman.rf
./usr/lib/adman/english_usa/net_msg.rf
./usr/lib/adman/english_usa/netman.sh

9   installation control files for INET group
10   customizable files for INET group

./etc/ftpusers
./etc/hosts
./etc/inetd.conf
./etc/networks
./etc/named.boot-----------Reserved for future use.
./etc/protocols
./etc/rcopts/NETD
./etc/resolv.conf
./etc/services
./etc/telbanner
./etc/netcf

11   required files for INET group

./etc/fingerd
./etc/ftpd
./etc/hostid
./etc/hostname
./etc/inetd
./etc/init.d
./etc/init.d/network
./etc/mkhosts
./etc/named------------Reserved for future use.
./etc/rc0.d/K75network
./etc/rc2.d/S86network
./etc/rc3.d/S86network
./etc/rexecd
./etc/rlogind
./etc/ctrlogind
./etc/routed
./etc/rshd
./etc/rwhod

```
./etc/talkd
./etc/telnetd
./etc/tftpd
./etc/uucpd
./etc/ouucpd
./usr/hosts
./usr/local/bin/arp
./usr/local/bin/finger
./usr/local/bin/ftp
./usr/local/bin/ping
./usr/local/bin/rcmd
./usr/local/bin/ctrcmd
./usr/local/bin/rcp
./usr/local/bin/rlogin
./usr/local/bin/ctrlogin
./usr/local/bin/ruptime
./usr/local/bin/rwho
./usr/local/bin/talk
./usr/local/bin/telnet
./usr/local/bin/tftp
./usr/spool/ftp
./usr/spool/ftp/bin
./usr/spool/ftp/etc
./usr/spool/ftp/pub
./usr/spool/rwho
```

12	installation control files for KFIG group
13	customizable files for KFIG group
14	required files for KFIG group

```
./install/TcpRelmiti1
./install/TcpRelmiti2
./install/TcpRels80
./install/TcpRels280
./usr/sys/libtcp
./usr/sys/libip
./usr/sys/libudp
```

The INET group also installs the following zero-length files in
/etc/rcopts: KINET, KSOCK and KSTRM.

Following are specific changes made to the contents of the install tape:

1. File */etc/netcf* has been changed to a customizable file.

2. Library files in directory /usr/sys have been renamed without changing there content.

6. Installation Procedures from Tape Media

One type of distribution medium is supported in this release: Quarter-Inch Tape (part numbers A or B).

In the following installation procedures, characters that you must type are shown in **boldface**. Special keys, such as RETURN, are shown in upper case letters.

6.1 S/Series CTIX TCP/IP Installation from Quarter-Inch Tape

To install S/Series CTIX TCP/IP 3.2, perform the following steps:

1. Log in as root. Provide a password, if necessary. The system responds with a # prompt.

2. Bring the system into single-user mode by entering:

   /etc/shutdown RETURN

   Before installing the product, wait for the system to prompt with the message:

      Ok to Reboot or Halt processor

3. If /usr is configured on /dev/dsk/c0d0s3, then execute the following step to mount the /usr on your system :

   mount /dev/dsk/c0d0s3 /usr RETURN

4. Insert the distribution quarter-inch tape into the drive and close the latch.

5. Execute the installation script:

   /usr/local/bin/ctinstall RETURN

6. The following output appears on the screen. User supplied responses are shown in bold type:

ctinstall.sh 6.32

Positioning the Tape for Product Installation

Update, silent update or new installation of CTIX TCP/IP 3.2
('update', 'silent' or 'install')?: **install** RETURN

7. The installation continues with:

Re-mounting /usr.
Please enter your group choices for TCPIP
separated by blanks.
Your choices are:

TCPIP INET KFIG

If you'd like all of the groups, type 'all';
If you'd like none of the groups, type 'none': **all** RETURN

This procedure installs the following CTIX TCP/IP 3.2
groups on your system:

TCPIP INET KFIG

Starting to Install group(s) TCPIP.
Installing group TCPIP

Calculating size required for group TCPIP.
Installation will require an additional 256 root Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on root.)
Installation will require an additional 544 /usr Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on /usr.)

Installing required TCPIP files.

Checking permissions, modes, and omissions on new TCPIP
commands.

Now updating Adman main menu
Adman main menu has been updated

Completed Installation of Group TCPIP

Starting to Install group(s) INET.
Installing group INET

Calculating size required for group INET.
Installation will require an additional 976 root Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on root.)
Installation will require an additional 742 /usr Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on /usr.)

Installing required INET files.

Checking permissions, modes, and omissions on new INET
commands.

Would you like to specify an Internet domain?
Type 'yes' to confirm: **Yes**

Respond with a **Yes** to the above question
if you are going to connect to the Internet.

Say **no** if you are going to have a local
network and not be a part of the Internet domain.

Completed Installation of Group INET

Starting to Install group(s) KFIG.
Installing group KFIG

Calculating size required for group KFIG.
Installation will require an additional 8 root Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on root.)
Installation will require an additional 288 /usr Blocks
(512 Byte Blocks).
(Currently NNNN 512 Byte Blocks are available on /usr.)

Installing required KFIG files.

Checking permissions, modes, and omissions on new KFIG
commands.
Completed Installation of Group KFIG
Rewinding Tape.

Installation Complete.

For information on how to configure and start your
network software, see section 7, "Adding Your Machine
to a Network".

7. Adding your Machine to a Network

Once you have loaded the TCP/IP software from the distribution tape, you must set your machine up to be part of a network. You should have the manuals mentioned at the beginning of this Notice handy while you follow the instructions on the next few pages.

7.1 Configuring your S/Series server to Support Intelligent Ethernet Controllers

If you are going to support local I/O bus Intelligent Ethernet Processors on your system, no configuration is needed other than installing the boards. The rest of this section pertains to VME cards only.

If you are going to support VME Intelligent Ethernet Processors on your system, the first step is to use the program *ldeeprom* to configure your S/Series system for Ethernet. This program actually stores the configuration of your VME card cage in a segment of non-volatile memory which the operating system can examine at boot time (and loadable drivers examine when loaded). The Ethernet software checks the non-volatile memory to decide which Ethernet boards are currently in the system and what their base memory addresses are.

Follow these steps to install an Ethernet controller in your system:

1. Set the jumpers on the Ethernet controller board to the address at which the board will be running (See the controller installation manual for details). Note that the boards are shipped with base address set to C0DE0000, the default address for your first Ethernet board. Therefore, if you are using only one Ethernet board, it is not necessary to change the jumpers.

2. Install the board in your VME Card Cage according to the instructions in the controller installation manual.

3. Edit the file */etc/system* to include your new Ethernet controller and its address. For details about how to do this, see the *S/Series CTIX Administrator's Manual* (DAC-110) section, 'Adding New Peripheral Devices" In brief, it

is necessary to add a line to the system file with the slot number of the Ethernet board, the board type (1 for Intelligent Ethernet Controllers), the base address (see below), and the amount of memory space it uses (131072 for Intelligent Ethernet Controllers). Thus, the line describing an Ethernet board might look like this:

0   1   c0de0000 131072

The /etc/system file is already loaded as a part of your operating system. It may already contain an entry for a Convergent VME Ethernet Controller. This entry will contain a comment character in the first column (*). You can activate this entry by removing the asterisk character.

The base address for the Ethernet controller is assigned according to the following table:

Board Number	Base Address
0	c0de0000
1	c0e00000
2	c0e20000
3	c0e40000
4	c0e60000
5	c0e80000
6	c0ea0000
7	c0ec0000

Note that these addresses correspond to the addresses that you set with the jumpers, as described in the controller installation manual; however, they are preceded here by c0.

4. Load the new configuration into the non-volatile memory by logging in as root and typing:

/etc/lddrv/ldeeprom   RETURN

This program takes a while to load the non-volatile memory.

7.2 Supporting Networks using Asynchronous Serial Lines

To use either form of the Serial Line Internet Protocol (SLIP), you must load the software by creating the empty file */etc/rcopts/KSLIP*.

7.2.1 Direct SLIP

If you want to permanently connect machines using SLIP, explicitly initialize each network connection using the program *slattach*(1NM). This program sets up the appropriate interface and route descriptions within the socket driver. For each serial port that you want to use for SLIP, add a line to the file */etc/rcopts/DEVICES* which runs *slattach* for that port. It is actually not mandatory to run *slattach* at boot time; *slattach* and *sldetach* can be used at any time to create and remove SLIP links. Your decision to run these commands, either from */etc/rcopts/DEVICES* or from the command line, should be based on how often these connections change.

7.2.2 Switched SLIP

Switched SLIP is a form of SLIP that can run over switched serial media, such as PBX's and telephones with modems. To run it, you must create the empty file */etc/rcopts/SLIPD*. See the *CTIX Network Administrators Manual* (DNC-180) and *slipd*(1NM) in the *CTIX Operating System Manual* (DAC-200) for further instructions.

7.3 Additional Steps Necessary to Set Up a Node

1. Make sure */etc/rcopts/NODE* has your correct hostname.

2. If you are adding a new network, add the new network's information to the file */etc/networks*.

3. If your machine has more than one Ethernet interface, edit */etc/rcopts/KENP* to contain the addresses of all interfaces. See "Using Network Gateways" below.

4. On the network master, use *adman(1NM)* to add the new machine's name (and aliases) and addresses to the host file, */etc/hosts*. The name of your machine must be included in the network hosts database. (Do this even if the only machine in the network as yet is your own machine.)

5. On the new machine, if different from the network master, use *adman(1NM)* to add the new machine's name (and aliases) and address to the host file, */etc/hosts,* so that the host database of the local machine reflects that of the network master. As an alternative, you can retrieve the */etc/hosts* and */etc/networks* files from the network master. See the section on "The Hosts File" below.

6. After creating the hosts database, run */etc/mkhosts* to create the */usr/hosts* directory.

7. Set up the appropriate user and machine equivalences using *adman(1NM)*. This must be done on both "old" and "new" machines which want equivalences with the new machine.

8. Reboot the system by typing the following :

   sync; sync; reboot

9. Using *adman(1NM)* set up the servers as explained, in the *CTIX Network Administrators Guide.*

10. If you have a large network or one that you expect to grow to a large network, set up a program to automate this updating of the network databases. See the *CTIX Network Administrators Manual* for more information.

The machine is now up and running on the network.

7.4 Increasing Memory Allocation

For installations using several networking interfaces, or where very high load is put on the networking subsystem, performance and reliability can be enhanced by increasing the amount of memory allocated to STREAMS. If some commands fail with an error message related to STREAMS resources, or *netstat -m* reveals allocation failures, more resources should be allocated.

The memory allocated to STREAMS, as shown by *netstat -m*, can be configured either by editing */etc/system* and running *uconf -w*, or by editting the appropriate *dfile* in */usr/sys/cf* and rebuilding CTIX. In general, Ethernet uses class 4 (256 byte)

data blocks, and SLIP uses class 6 (1024 byte) blocks. A general strategy is to allocate a lot of memory (enough to avoid allocation failures), run the system under load, then check the "Hiwater" counts shown by *netstat -m*, and set allocation to about 10 percent over high water.

For example, to set class 4 allocation at 512 buffers, either add the line **v_nblk256=512** to the *TUNEABLES* section of */etc/system* and run *uconf -w*, or add the line **nblk256 512** to your *dfile* and rebuild CTIX. Reboot the system to initialize the new value.

7.5 Using Network Gateways

If your network consists of more than one physical network connected through common machines, it is called an internetwork. In an internetwork, data from one network is passed to other networks through the connecting machines, which are called gateways. To the user an internetwork looks exactly like a large single network. However, the internetwork approach has significant advantages over a large single network in many situations:

- High Speed Local Area Networks and slower Wide Area Networks can be mixed to allow for maximum performance/economy balance. Typically, this means that people and machines within a workgroup or other local area can get very high performance networking between one another and still have transparent access to remote people and machines.

- The use of gateways allows high network traffic areas to be localized. On a large Ethernet, traffic from a small set of stations can jam the entire network. With localized traffic, each part of the internetwork can have sufficient bandwidth not to get jammed. This approach assumes that the localization decisions can be made in a reasonable fashion (i.e. there are actually local traffic areas within the network).

- With internetworking, previously existing networks can usually be combined into an internetwork by connecting a single machine to both networks and having it act as a

gateway.

If you have more than one ethernet interface on your machine, you will have to assign multiple addresses to the machine by adding extra entries to */etc/hosts*. Then put the names in the file */etc/rcopts/KENP* in the order of the ethernet unit associated with each network. Thus, if you have net 3 on unit 0, and net 4 on unit 1, and AddrA is 3.1 (a net 3 address) and AddrB is 4.1, */etc/rcopts/KENP* would contain the single line **AddrA AddrB**.

If you want to configure your machine to utilize the gateway functionality, then add the following line to the TUNEABLES section of the file */etc/system*, run **uconf -w** and remember to reboot the machine before using it. The line to be added follows:

net_ipforwarding=1

In order to use the internetworking features of CTIX TCP/IP, you must set up 'routes'. A route is a record kept by IP of which gateway to use to get to a certain remote network. You can add and delete routes from the table kept by IP using the *route*(1NM) command, or by running *routed*(1NM). Once a route to a remote network has been established, users can use all the networking commands and refer to machines on that network as if they were on the local network (assuming the machine on the other end also has a route to get back). It is important to remember to use the metric option of the route command when creating routes through gateways.

If you want to set up routes at boot time for permanent use, add the route command lines to */etc/rcopts/ROUTING*. The route commands should have the following form:

route add DEST Gateway N

where DEST is the name or network number of the network to which you are connecting, Gateway is the name or internet address of the gateway machine, and N is the number of gateways between this network and that one (at least one). A special case of DEST is 0 (zero), which indicates that this route is the "default route", to be used to get to any destination not

specified in another routing entry (N should be 1 for the default route). Routes are usually added from the m/cs not running as gateways.

If you want to run *routed* to automatically handle your routing, either delete */etc/rcopts/ROUTING* or call *routed* from it. No explicit *route* commands should be used in conjunction with *routed*. See routed(1NM) and gateways(4N) for more information.

7.6 The Hosts File

There are several ways that a machine can keep its host and network files up to date. The first method is simply to use either *adman*(1NM) or an editor to keep an up to date list of all the machines that the users of the machine wish to know about. This method is useful if the number of machines of interest is small and fairly constant.

In a larger, more dynamic network configuration, it is a better idea to maintain a central master copy of the hosts known to all the machines on the network, and update each machine periodically. This way when a new machine is added to the network, only one file needs to be updated for the whole network to find out about the new machine. You can use the *cron*(1M) command to pick up a new edition of the master file each night. See the *CTIX Network Administrators Guide* for more details on implementing this method.

In either case, you must set up a host file (*/etc/hosts*) to use most of the commands. This file can be set up with the add and change hosts commands available from *adman,* or the file can be edited directly. See *hosts*(4N) in your *CTIX Operating System Manual* for a description of the format of this file. If you are using the master copy approach, you need only enter your machines own name and address into the host file, then use *ftp* to explicitly pick up the master hosts file from the machine it's on. In this use of *ftp,* use the internet address of the destination machine rather than its name. Do this only after starting the Ethernet board by rebooting or running *enpstart*(1NM). Here is an example of using *ftp* to get a master hosts file (user input is boldface):

```
$ ftp
ftp> open 3.0.0.16
Connected to 3.0.0.16.
Name (3.0.0.16:root): root
Password: (root password for machine 3.0.0.16)
ftp> cd /etc
ftp> lcd /etc
Local directory now /etc
ftp> get hosts
ftp> get networks
ftp> bye
$
```

Other methods of maintaining host table information include the Internet Name Server and the Yellow Pages (SUN Microsystems).

8. Hardware Requirements

CTIX TCP/IP 3.2 requires an S/Series system with 2 MBytes of memory. You can use one or more Convergent local I/O bus Ethernet/RS232 controllers, and/or VME bus Ethernet controllers, and/or a serial link (see *slattach(1M)*).

In addition to the Convergent equipment required for Ethernet, each controller requires a drop cable and transceiver, and one Ethernet backbone cable is required per network.

8.1 CB-510 Ethernet/RS232 Controller Requirements

If you want to use a local I/O Ethernet/RS232 controller (CB-510) no extra hardware is needed. However, some restrictions apply.

The S/80 and S/280 hardware use SCSI LAN DMA channels. The S/120 to S/320 hardware contains two DMA channels for initelligent processors. The S/480 and S/640 have three DMA channels each. One is assigned for VME bus controllers and the other is assigned for the IOP or RS422 controller. One or more Ethernet/RS232 controllers can share **one unassigned** channel.

The system can contain one or more Ethernet/RS232 controllers and also one or more Convergent VME controllers.

The system can contain one or more Ethernet/RS232 controllers and either an IOP terminal I/O processor or an RS422 controller.

The S/320 **cannot** have an Ethernet/RS232 controller and **both** an IOP/RS422 controller **and** a VME controller.

On an S/Series system up to four Ethernet/RS232 controllers can be used.

On an S/220 only two Ethernet/RS232 controllers can normally be used. If 422 and IOP boards are absent then 3 Ethernet/RS232 boards may be installed.

8.2 EC-500 Ethernet VME Controller Requirements

If you want to use VME Ethernet, you will also need a VME Expansion (CV-500), and an intelligent Ethernet controller (EC-500). To use more than four VME Ethernet cards in a system, you will require a S/Series System VME Cabinet (CG-3502-V for 110 Volts or CG-4602-V for 220 Volts).

9. Documentation

9.1 Additional Documentation

- CTIX Operating System Manual, Version C, Vols 1 through 4 (DAC-200)

- CTIX Network Administrator's Guide, First Edition 09-00665-01   (DNC-180)

- CTIX Administrators's Guide, (3rd Edition)

10. Known Errors, Warnings, and Restrictions

10.1 Warnings

- If you plan to use NFS and you are using the /etc/hosts file for the hosts data base, Make sure you add the Internet Domain name to the NFS server name as an alias in the hosts file, along with the server name. For example:

Internet Domain	:	Mysite.COM
NFS Server	:	nfsserver
Internet Address	:	3.0.0.1

  For the following configuration, the entry in the /etc/hosts file should look like: 3.0.0.1 nfsserver nfsserver.Mysite.COM

- Add /usr/hosts to user PATH, if you plan to use remote commands over the network.

- The shell layers system (*shl*(1)) is not implemented for the pseudo-terminals used by *rlogin* and *telnet*.

- CTIX TCP/IP 3.2 may not work correctly with hosts that set the high-order bit in sequence numbers.

- The **cancel** (break) key can cause undesirable effects, depending on the program you are running.

- If you are using the *tftp* put command, make sure a null file existing on the remote system, with read-write permissions. Otherwise *put* command may never be successful.

10.2 Restrictions

Named will not be supported in S/Series TCP-IP 3.2 release.