IP/TCP Network Support

symbolics

Cambridge, Massachusetts

IP/TCP Network Support

 # 998014

April 1985

This document corresponds to Release 6.0 and later releases.

Text written and produced on Symbolics 3600-family computers by the Documentation Group of Symbolics, Inc.

Text typography: Century Schoolbook and Helvetica produced on Symbolics 3600-family computers from Bitstream, Inc., outlines; text masters printed on Symbolics LGP-1 Laser Graphics Printers.
Cover design: Schafer/LaCasse
Cover printer: W.E. Andrews Co., Inc.
Text printer: ZBR Publications, Inc.

Printed in the USA.

Printing year and number: 87 86 85 9 8 7 6 5 4 3 2 1

Table of Contents

List of Figures

1. Overview

The IP/TCP Network Support package allows the Symbolics Lisp Machine to communicate with other systems supporting the Internet protocol family. These systems can be located on either the local Ethernet or another Internet network connected to the local Ethernet through a gateway. The Internet protocols supported include remote login (TELNET), file transfer (FTP and TFTP), and electronic mail and messages (SMTP). All of these services are accessible to the Symbolics Lisp Machine user through the generic network system. It is also possible to use the generic network system to add support for new or special-purpose Internet protocols to the basic system.

IP/TCP runs on Symbolics 3600, 3640, and 3670 Lisp Machines running Release 6.0 software.

The generic network system is described in the section "The Lisp Machine Generic Network System" in the document *Networks*. See Appendix C for detailed information about the Internet protocol family.

2. Installation

The following sections describe the process of installing IP/TCP at your site.

If you are not yet familiar with the namespace editor, read the section "Namespace System" in the document *Networks* before proceeding.

2.1 Restoring IP/TCP from the Distribution Tape

You must have completely installed Release 6.0 before beginning the installation of IP/TCP. To install IP/TCP from the distribution tape, perform the following steps in the numbered order:

1. Place the distribution tape in the cartridge tape drive.

2. Add logical directories for IP/TCP.

   The IP/TCP software will be located in the logical directories ip-tcp:ip-tcp; and ip-tcp:ip-tcp;patch;. Before restoring the software from the tape, you must add appropriate translations for these logical directories to your site directory. Create a file named ip-tcp.translations (on UNIX: ip-tcp.ld; on VMS: IPTCP.LDT) in your site directory describing where the software should be placed. The equivalent **fs:set-logical-pathname-host** should be evaluated at a Lisp Listener. For example:

```
;;; -*- Mode: Lisp; -*-

(FS:SET-LOGICAL-PATHNAME-HOST "IP-TCP"
     :PHYSICAL-HOST "YUKON"
     :TRANSLATIONS '(("IP-TCP;" ">sys>ip-tcp>")
                     ("IP-TCP;PATCH;" ">sys>ip-tcp>patches>")))
```

   Here, the software will be stored in the >sys>ip-tcp> directory on the host named YUKON. If you need more information on logical directories, pathnames, translations, and the like, see the section "Logical Pathnames" in the document *Reference Guide to Streams, Files, and I/O*.

3. Create a system location file for IP/TCP.

   To allow the system to automatically find the IP/TCP system files, you must create a system location file in your site directory. It should be named ip-tcp.system (on UNIX: ip-tcp.sy; on VMS: IPTCP.SYD). It must contain the following text:

```
;;; -*- Mode: Lisp; -*-

(FS:MAKE-LOGICAL-PATHNAME-HOST "IP-TCP")

(SI:SET-SYSTEM-SOURCE-FILE "IP-TCP" "IP-TCP: IP-TCP; SYSTEM")
```

This indicates to the system that the "IP-TCP" system is stored on the logical host IP-TCP and that the system declaration file is in ip-tcp: ip-tcp; system.lisp.

4. Load the distribution tape by typing the following form to a Lisp Listener:

   (dis:load-distribution-tape)

   The distribution loader uses two menus to control how the sources are loaded. The first is `Where to get distribution tape`. Indicate the tape host, if it is remote, and click on [Do It]. A second menu, `Items to be loaded`, appears; click on [Do It] to restore the IP/TCP files from the tape to the logical directories mentioned previously.

   If you interrupt the restoration process before it completes and later attempt to restore the files on the tape again, you should delete all the files that were restored from the previous attempt. Otherwise, **dis:load-distribution-tape** signals an error when it tries to restore those files. If that happens, you can use the Zmacs command Dired (`m-X`) to edit the directory and delete the existing files.

5. Take the tape out of the cartridge tape drive.

6. Cold boot the machine.

2.2 Site Configuration

Two versions of the site configuration instructions appear below. One is for sites that are connected via gateways to the Department of Defense ARPA Internet; the other is for sites that are not so connected.

2.3 Configuring a Site Not Connected to the ARPA Internet

If you have not already done so, you should obtain a Class B or Class C Internet network number for your local Ethernet network. These should be obtained from:

Joyce Reynolds
USC - Information Sciences Institute
4676 Admiralty Way
Marina del Rey, California 90292
(213) 822-1511
ARPANET: JKREYNOLDS@SRI-NIC.ARPA

Perform each of the following steps on your site's primary namespace server:

1. With the namespace editor (**tv:edit-namespace-object**), create a network object to represent the Internet network. Make its name INTERNET and its type INTERNET. This is necessary even though the Internet network shares the physical cable with the existing Chaos network.

2. Perform the following host definition steps for all local hosts which will support the Internet protocols.

    a. Determine Internet addresses for each of the hosts. We strongly suggest that for hosts that also support the Chaos protocols the host part (last octet) of the Internet address be the same as the host part (low-order 8 bits) of the Chaos address. For example, if the Chaos address is 403 (subnet 1 host 3), the Internet address might be 192.10.1.3 (network 192.10.1.0 host 3). Gateway or multihomed hosts should have addresses assigned for each network to which they are connected. Assign Internet addresses on the appropriate networks to each host.

    b. For each of these hosts, edit its host object with the namespace editor (**tv:edit-namespace-object**) to add the addresses assigned above on the Internet network. Be sure to add all addresses for gateway and multihomed systems.

    c. Add any of the service entries from figure 1, page 6, that describe servers present on that machine. The Symbolics Lisp Machine can support the services marked with an asterisk (*). You should consult the documentation for other systems' IP/TCP packages to determine which servers they supply. The ASCII-NAME service is sometimes called FINGER in the documentation provided by other companies.

Site configuration is now complete. IP/TCP will be operational the next time a world containing IP/TCP is booted on a system which has been assigned an Internet address.

```
  SERVICE GATEWAY                  IP  INTERNET-GATEWAY

* SERVICE SHOW-USERS               TCP ASCII-NAME
* SERVICE MAIL-TO-USER             TCP SMTP
  SERVICE STORE-AND-FORWARD-MAIL   TCP SMTP
* SERVICE SEND                     TCP SMTP
* SERVICE LOGIN                    TCP TELNET
  SERVICE LOGIN                    TCP SUPDUP
* SERVICE TIME                     TCP TIME-MSB
* SERVICE FILE                     TCP TCP-FTP

* SERVICE SHOW-USERS               UDP ASCII-NAME
* SERVICE TIME                     UDP TIME-MSB
* SERVICE FILE                     UDP TFTP
```

The entries marked with an asterisk (*) are supported by the Lisp Machine.

Figure 1. Service Entries for IP/TCP

2.4 Configuring a Site Connected to the ARPA Internet

Describing the process of physically configuring the Internet gateways for your network is beyond the scope of this document. It is assumed that your Internet includes one or more Ethernet networks that have been assigned Internet network numbers and that are connected to the rest of your Internet through one or more gateway machines.

Perform each of the following steps on your site's primary namespace server:

1. Cold boot the system.

2. Determine the Internet addresses to be assigned to the namespace server and to its primary gateway to the rest of the Internet. Do the following:

```
(si:login-to-sys-host)
(make-system "ip-tcp" :noconfirm)
(tcp:initialize-internet-namespace)
Name of the namespace to create (INTERNET)? INTERNET
Directory where INTERNET namespace files should be kept
       (default LOCAL:>sys>site>)? LOCAL:>sys>site>
Internet address of your local host?
       address on the Ethernet of the namespace server
Name of your internet gateway? name of the gateway on the Ethernet
Internet address of gateway? address of the gateway on the Ethernet
```

We strongly suggest that for hosts that also support the Chaos protocols the host part of the Internet address be the same as the host part (low-order 8 bits) of the Chaos address. For example, if the Chaos address is 403 (subnet 1 host 3), the Internet address might be 192.10.1.3 (network 192.10.1.0 host 3).

The **make-system** function might print warning messages having to do with changed flavors; these can be ignored.

The function **tcp:initialize-internet-namespace** initializes a prototype namespace for the Internet and makes the local host its primary namespace server. It also adds the new namespace to the site's local namespace's search list and adds an address for the local host on the new network.

3. Use **disk-save** to save the current world load. This world should be used by the namespace server from then on. As it has IP/TCP loaded, it can also be copied to other systems to avoid having to load TCP separately. When the save completes, log in the server normally.

4. Perform the following host definition steps for all local hosts that will support the Internet protocols.

    a. Assign Internet addresses on the appropriate networks to each host. We strongly suggest that for hosts which also support the Chaos protocols the host part of the Internet address be the same as the host part (low-order 8 bits) of the Chaos address. For example, if the Chaos address is 403 (subnet 1 host 3), the Internet address might be 192.10.1.3 (network 192.10.1.0 host 3).

    b. Edit the host object of each host in your site's local namespace with the namespace editor (**tv:edit-namespace-object**) to add the addresses assigned above on the INTERNET network. Be sure to add all addresses for gateway and multihomed systems. Also add any of the service entries from figure 1, page 6, that describe servers present on that machine. The Symbolics Lisp Machine can support the services marked with an asterisk (*). Consult the documentation for other systems' IP/TCP packages to determine which servers they supply. ASCII-NAME is sometimes called FINGER in the documentation provided by other companies.

5. Do the following:

```
(tcp:install-nic-host-table)
Output file (default LOCAL:>sys>site>internet-hosts.text)?
        LOCAL:>sys>site>internet-hosts.text
(neti:read-object-file-and-update "internet" :host)
```

   The function **tcp:install-nic-host-table** reads the NIC host table and loads it into the INTERNET namespace object file. The namespace itself is updated by the function **neti:read-object-file-and-update**.

   You can repeat this step at any later time, to install a more recent version of the NIC host table.

Site configuration is now complete. IP/TCP will be operational the next time a world containing IP/TCP is booted on a system which has been assigned an Internet address.

2.5 Loading IP/TCP on a Symbolics Lisp Machine

To load IP/TCP on an individual Symbolics Lisp Machine, do the following:

```
(make-system "ip-tcp" :noconfirm)
```

IP/TCP is now loaded. You can save the system with **disk-save** at this point if you do not wish to reload the software each time you boot the machine. Saved world loads with IP/TCP loaded can be moved freely among machines. If you do not immediately save the world, you must do the following before using IP/TCP:

```
(neti:reset)
(neti:enable)
```

2.6 Security

Since the Symbolics Lisp Machine system does not have any internal security features, it might be desirable to prevent uncontrolled use of servers with access to sensitive information, such as the TELNET server. It is possible to configure the system so that connections to these servers from "untrusted" networks will be refused.

To put this in effect, edit the local site object with **tv:edit-namespace-object**. Add a new "Secure Subnets" attribute as follows:

`Secure Subnets:` **pair:** *internetwork-name* **set:** *network-numbers*

where *internetwork-name* is the name of your Internet network (usually INTERNET) and *network-numbers* are the Internet network numbers of the networks which are to be "trusted." For example:

```
Secure Subnets: pair: INTERNET set: 192.10.41.0 192.10.57.0
```

would cause only hosts on networks 192.10.41.0 and 192.10.57.0 of the internetwork named INTERNET to be trusted.

2.7 Tuning

IP/TCP is shipped is shipped in a state appropriate for general use on a local area network. Certain parameters can be altered to tune the performance of the IP/TCP system for specific situations. If you change any of these, you should carefully monitor your network's performance for unexpected side effects.

tcp:*ip-default-max-packet-size* *Variable*
Controls the maximum length of IP datagrams that can be sent through gateways. Its normal value is 576 octets.

tcp:*background-interval* *Variable*
Controls the interval between executions of the IP background routing function. Its normal value is 3600 (1 minute). See also the definitions of **tcp:*dead-gateway-ping-interval*** and **tcp:*live-gateway-ping-interval***.

tcp:*ip-debug-flag* *Variable*
Controls whether various unexpected conditions within IP cause notifications. Its normal value is **nil** (no notifications).

tcp:*dead-gateway-ping-interval* *Variable*
Controls the rate at which gateways believed to be down are probed to see if they have come up. Its normal value is 3600 (1 minute). Probing happens only when the IP background routing function runs (see **tcp:*background-interval***).

tcp:*live-gateway-ping-interval* *Variable*
Controls the rate at which gateways believed to be up are probed to see if they have gone down. Its normal value is 36000 (10 minutes). Probing happens only when the IP background routing function runs (see **tcp:*background-interval***).

tcp:*default-window-size* *Variable*
Controls the TCP window offered to remote hosts. Its normal value is 20000 octets.

tcp:*max-window-size* *Variable*
Controls the maximum TCP window that will be used on a remote host. Its normal value is 20000.

tcp:*adaptive-tcp-retransmission-enabled* *Variable*

Controls whether TCP retransmission uses a fixed retransmission interval or attempts to adapt the retransmission interval to the response time of the remote host. Its normal value is **nil** (no adaptive retransmission). Adaptive retransmission algorithms are stable only if the variance of the response time is not too large. If the variance is small enough, setting this to **t** can significantly increase the performance of the TCP connection.

tcp:*tcp-retransmit-interval* *Variable*

Controls the initial retransmission interval for TCP connections. Its normal value is 120 (2 seconds). If adaptive retransmission is enabled (see **tcp:*adaptive-tcp-retransmission-enabled***) the retransmission interval is adjusted to match the remote host's actual response time.

tcp:*tcp-idle-probe-interval* *Variable*

Controls the rate at which TCP "idle probe" and "zero window probe" messages are sent. The normal value is 7200 (2 minutes). "Idle probe" messages are sent over connections over which there has been no traffic during the interval. They contain only the appropriate ACK. They are sent in the hope of causing an RST or ICMP Destination Unreachable if the connection has actually died.

tcp:*tcp-response-timeout* *Variable*

Controls the time after which a TCP connection will be abandoned if the remote host does not respond. Its normal value is 3600 (1 minute).

tcp:*record-tcp-debugging-info* *Variable*

Controls whether TCP header information for the last 64 segments should be recorded for the use of the **tcp:print-recent-tcp-headers** function. This can be used to debug network problems. Its normal value is **t** (recording enabled).

3. Using IP/TCP on the Symbolics Lisp Machine

IP/TCP protocols are used automatically by the Lisp system whenever they are appropriate for performing some network service. For example, TCP/TELNET is used by the Terminal program, TCP/FTP is used when remote files are opened, and TCP/FINGER is used by the **net:finger** command. If a service could be performed using either Chaos protocols or IP/TCP protocols, Chaos usually takes precedence.

If you want to use TELNET between logged in Symbolics Lisp machines, you have to evaluate the form **(neti:telnet-server-on)** on the Symbolics Lisp Machine to which you want to connect.

3.1 Debugging Tools

Metering and debugging information about the status of IP/TCP can be found in the Peek program's Network display. Many items are mouse-sensitive to provide more detailed information.

In addition, the following function can be useful:

tcp:print-recent-tcp-headers &optional *n-headers* *Function*

n-headers is the number of headers to display. If it is not given, all those stored are displayed. The headers are displayed in reverse chronological order. **tcp:*record-tcp-debugging-info*** must be set to **t** to enable recording of headers.

Appendix A
Adding Other Internet Protocols

Application protocols for TCP and UDP are accessed through the generic network system. Protocol 'users' should be defined with **net:define-protocol**, 'servers' with **net:define-server**. (For an example of **net:define-server**, see appendix B.)

A.1 TCP

TCP supports only the generic **:byte-stream** medium. Use the function **tcp:add-tcp-port-for-protocol** to associate a TCP port number with the protocol name you have defined.

tcp:add-tcp-port-for-protocol *protocol-name port-number* *Function*

protocol-name is a keyword symbol, *port-number* a number. The given TCP port number is associated with the given protocol.

```
(tcp:add-tcp-port-for-protocol :smtp 25.)
```

A.2 UDP

UDP supports both the generic **:datagram** medium and the protocol-specific **:udp** medium. Use the **tcp:add-udp-port-for-protocol** function to associate a UDP port number with the protocol name you have defined.

tcp:add-udp-port-for-protocol *protocol-name port-number* *Function*

protocol-name is a keyword symbol, *port-number* a number. The given UDP port number is associated with the given protocol.

```
(tcp:add-udp-port-for-protocol :tftp 69.)
```

:udp *Medium*

The **:udp** medium supports one medium-specific **net:define-server** keyword:

:connection The value of this keyword is a symbol that will be bound to the UDP connection "stream".

Appendix B
Example: Definition of a Server

The following is an annotated example of a server definition, in this case, the **net:define-server** form that specifies the function **smtp-server** as a mail server, a function that handles the SMTP mail protocols. Once this server is defined, it can be provided as a mail server for various network types, including Chaosnet and IP/TCP, as shown. For background information on servers and protocols, see the document *Networks*, especially the chapter "The Lisp Machine Generic Network System."

```
;;; Define a server for the SMTP protocol operating over a byte stream.
(net:define-server :smtp (:medium :byte-stream
                ;;Bind the variable stream to the byte stream created by
                ;;the system.  The stream should use the ASCII
                ;;character set rather than the Lisp Machine set.
        :stream (stream :ascii-translation t)
                ;;Make the server appear in the wholine and in
                ;;Peek's Servers display.
        :who-line t
                ;;Do not automatically close the stream when the
                ;;function exits.
        :no-eof t
                ;;Bind network and host to allow smtp-server
                ;;to identify the user.
        :network network :host host
                ;;Display a notification if an error occurs in a
                ;;server process.
        :error-disposition :notify)
                ;;Actually invoke the server.
        (smtp-server stream network host))

;;;Define the Chaos contact name for the protocol.
(chaos:add-contact-name-for-protocol :smtp "SMTP")

;;;And the same for TCP.
(tcp:add-tcp-port-for-protocol :smtp 25.)
```

Appendix C
Internet References

All documents identified as ARPANET Requests for Comments (RFCs) are available from the ARPA Network Information Center:

ARPA Network Information Center
USC - Information Sciences Institute
4676 Admiralty Way
Marina del Rey, California 90292
ARPANET: NIC@SRI-NIC

For those with ARPA Internet access, they are also available online as

```
SRI-NIC:<RFC>RFC###.TXT
```

where ### is the RFC number.

C.1 References

Reynolds, J. & Postel, J., *Official Protocols*, RFC 880, October 1983.

Information Sciences Institute, *Internet Protocol*, RFC 791, September 1981.

Information Sciences Institute, *Internet Control Message Protocol*, RFC 792, September 1981.

Information Sciences Institute, *Transmission Control Protocol*, RFC 793, September 1981.

Postel, J., *User Datagram Protocol*, RFC 768, August 1980.

Postel, J., Reynolds, J., *TELNET Protocol Specification*, RFC 854, May 1983.

Postel, J., *File Transfer Protocol*, RFC 765, June 1980.

Sollins, K. R., *The TFTP Protocol*, RFC 783, June 1981.

Postel, J., *Simple Mail Transfer Protocol*, RFC 821, August 1982.

Harrenstein, K., *NAME/FINGER*, RFC 742, December 1977.

Postel, J., Harrenstein, K., *Time Protocol*, RFC 868, May 1983.

Crispin, M., *SUPDUP Display Protocol*, RFC 734, October 1977.

Harrenstein, K., White, V., Feinler, E., *Hostnames Server*, RFC 811, March 1982.

Reynolds, J., Postel, J., *Assigned Numbers*, RFC 870, October 1983.