TCP/IP COMMUNICATIONS IN A MUMPS ENVIRONMENT

Lee Hirz
Washington ISC
Department of Veterans Affairs

ABSTRACT

Systems that use MUMPS have often been accused of being isolated from non-MUMPS systems. TCP/IP is a communications protocol that is used by many different computers and many different operating systems. It is a door that can open up MUMPS systems to interaction with non-MUMPS systems. This paper describes an implementation of electronic mail application written in MUMPS that uses SMTP (Simple Mail Transfer Protocol), the standard for messaging that is used when systems that use TCP/IP send and receive messages. This is the standard for messaging that is used predominately on the Internet.

OVERVIEW

MailMan is the MUMPS based electronic mail (email) system used in the Department of Veterans Affairs Decentralized Hospital Computer Program.

As the Department of Veterans Affairs (VA) Decentralized Hospital Computer Program (DHCP) continues to grow, it has become more important to transmit textual data and to also be able to transmit non-textual information such as images, worksheets and other bit oriented data. MailMan's original design adhered to the TCP/IP Simple Mail Transfer Protocol (SMTP) standard, but it could only communicate with other SMTP systems through an assortment of gateways that made communications non-optimal. But a recent prototype of VA MailMan sends and receives messages directly across any TCP/IP channel as a peer with other SMTP compatible mail systems and also uses FTP (the TCP/IP File Transfer Protocol) to send bit oriented files to which are then linked to messages. This adds a lot of functionality to a MUMPS system especially in conjunction with the VA's imaging capabilities. Doctors may now send xrays and other images when requesting consults via email. Administrators may send spread sheets and graphs when illustrating their projects. It is expected that MailMan will continue to satisfy the needs of the VA for data transmission and multimedia mail into the 21st century.

PREPROJECT ANALYSIS

TCP/IP transmissions are made on an error-free synchronous communications channel. X.25 communications is the base of TCP/IP configuration that can send and receive data at much higher speeds than previously thought. The requirements of the DHCP Imaging Project were considered during the development of this functionality. It was decided that the functionality could be implemented using current technology and that it was necessary and desirable. It furthered our goal of providing the best possible patient care and fit into the DHCP paradigm for systems development. Error free communications would be provided more efficiently and with faster delivery times at a reasonable cost. System security would not be compromised. The package would be portable. Current systems would function more efficiently making better use of system resources, operations simplified.

The project was initiated because there is a great need to transfer data quickly and efficiently between different hardware configurations. The 'imaging project' at the Washington ISC was at a point where the progress they had made could be incorporated into MailMan -- Multimedia Mail. A prototype was developed before the June 1991 MUG meeting for multimedia mail. This proved that the concept was doable with current technology. The future needs have become clearer with the passing of time as imaging has been installed at more hospitals and users at these hospitals now could send images to each other.

HISTORICAL BACKGROUND

MailMan started using a very wide variety of methods to achieve the basic goal of communications when the ability to send and receive mail from remote locations (Network MailMan) was first released. Phone lines, direct lines between computers, ethernet (DDP), and an asynchronous connectivity through a wide area network (WAN) were capabilities in the first release of Network MailMan. All of these required application level error checking to make sure that no data was changed during transmission.

Over the years (Network MailMan has been in use for over 7 years) the amount of data sent across the network in the form of messages has increased dramatically. FORUM, the VA's national mail node, now has over 30,000 users, with more than 10,000 logons per day and normally sends over 4,000 messages to recipients at other locations each day. The speed of transmission of these messages has increased from 15 characters per second (cps) to over 200 cps. Over the long term (3-5 years) we expect that this will not be sufficient.

ANALYSIS

What are the limitations to how fast data can be sent across the network ? First, there is the speed of the physical layer that is used. Modem speeds are now possible at 38,000 bits per second (bps). On our private wide area network we are experimenting with speeds of 19,600 bps. Between some nodes we are running direct 56,000 bps lines. The physical link is not the problem at this time.

Second, there are constraints on how fast we can get data off of the disk. On some machines, this can be a limitation. However on the great majority of systems, this is not a big problem, except during prime time computer usage hours. However, even here greater speed can be achieved if the disk hits are made quickly enough that blocks accessed remain in cache, before they are swapped out because of lack of use. This is the primary concern on a heavily used system.

Third, there are the resources used to do error checking. Error checking done at the application level in MUMPS is very expensive in terms of computer resources. Even when done from MUMPS using a $ZCALL, the process is fairly intensive. If the error checking can be passed off to some other CPU, this will speed up transmissions. At the same time, if the channel to be used is 100% reliable, then there are no retransmissions of data. Fewer retransmissions will mean more efficient use of both the WAN and local and remote machine resources. Avoiding analysis of the data stream to see if it is necessary to retransmit data will also save valuable computer resources at both the sending and receiving sides of the transmission.

How to best use time on the WAN is also a consideration. There is always a lag time between the sending of a stream of characters across the WAN. This can be expressed in seconds. The faster methods of transmitting mail using MailMan tried to optimize use of this lag time. But there was always some lag time that was not efficiently used.

All of these factors effect costs. Saving current resources reduced the need for future system expansion. All of our VAX sites have the hardware and software that they need for implementation of TCP/IP channel communications. Studies are progressing on supplying this capability to our 486 sites. Currently it would appear an inexpensive ($10,000 to $15,000 per site) proposition to bring up the 486 sites on TCP/IP MailMan, using our more primitive asynchronous transmission methods as backup.

Network MailMan has always been programmed to allow it to run in a hardware independent manner and to use standards. The initial implementation of the Simple Mail Transfer Protocol (SMTP), which is a member of the suite of TCP/IP protocols, is now successfully communicating with any other SMTP compatible mail system. Virtually all mail systems that are capable of using the Internet are SMTP compatible. We have been successful in communicating with numerous non-MUMPS (unix, ultrix, VMS...) mail systems. This is standard DHCP approach. While the first implementation developed ran under VAX DSM. We have already been successful running under Ultrix.

Systems security has not changed significantly. TCP/IP MailMan does not create any security problems that did not exist previously. TCP/IP MailMan is significantly faster and more efficient. Speeds of up to almost 5000 cps have been recorded on tests to remote sites and 40,000 cps under ideal conditions across a local ethernet. Operations are easier to maintain. There are fewer problems that can occur, especially during the actual transmission of messages.

This is because TCP/IP channels require no application level error checking and make use of all network lag time. Error checking done at the physical layer of the protocol is CRC-16 and is done on a CPU separate from that of the computer (usually on an X.25 card). This method of message transmission and reception is therefore fast , free from error, and dependable.

How does it work?

RECEIVING MESSAGES:

o A foreign TCP/IP SMTP mailer requests service on your TCP/IP channel.

o A job that runs separately from MUMPS and that controls the TCP/IP channel intermediates this request which comes in on logical channel (socket) 25 and negotiates a channel (say, 1056) on which to actually do the work.

o When the channel on which the message transmission will take place has been decided, control of this channel (1056) is passed to MailMan. The TCP/IP job then goes back to monitoring socket 25 for the next service request.

o MailMan then provides the SMTP receiver services to the remote (sending) email system. When the session is over MailMan halts. The logical channel is closed.

SENDING MESSAGES:

o MailMan uses the DEC supported open parameter to establish contact with a foreign CPU via a TCP/IP channel on socket 25 (where requests for SMTP service are made). This is done entirely in a transmission script. Though the open appears to be opening a local device 25, it is really opening a connection to the remote host, while at the same time specifying SMTP service. The actual communications channel is negotiated. This means that multiple outgoing transmissions can be opened concurrently.

o After the channel has been successfully established, the transmission will continue using the SMTP protocol for data exchange.

o When there is no mail left to send (or receive if the TURN command is executed) the channel will be closed.

A copy of the instructions for installing TCP/IP MailMan a VAX running VMS may be obtained by writing to Lee Hirz at the Washington Information Systems Center, 8402 Colesville Road, Suite 200, Silver Spring, Maryland 20910.

The author owes much to Peter Eklund and Steve Pollock for their roles in making this possible. Others who provided valuable help include members of the Washington ISC technical support staff (Milt Roberson and Jack Divers) and people who participated in the alpha testing of the product (Randy Edwards and David Bradley).

CONCLUSION

TCP/IP communications are now a fact within DHCP. There are now a significant number of sites using TCP/IP channels between them successfully and with minimal maintenance. Mail is exchanged at high speeds and error free. We are now alpha testing multimedia electronic mail functionality using TCP/IP channels and have been successful in porting the capability to multiple hardware configurations. Our MUMPS electronic mail system exchanges mail with non-mumps systems daily (in production mode) using the TCP/IP SMTP protocol . New code did not need to be written. The functionality has lived up to its promise of providing MUMPS the capability to exchange information on a peer to peer basis with non-MUMPS systems.