

University of Alabama in Huntsville

LOUIS

Honors Capstone Projects and Theses

Honors College

Summer 1997

Mail Server Installation and Integration

Tim Lewis

Follow this and additional works at: <https://louis.uah.edu/honors-capstones>

Recommended Citation

Lewis, Tim, "Mail Server Installation and Integration" (1997). *Honors Capstone Projects and Theses*. 465.
<https://louis.uah.edu/honors-capstones/465>

This Thesis is brought to you for free and open access by the Honors College at LOUIS. It has been accepted for inclusion in Honors Capstone Projects and Theses by an authorized administrator of LOUIS.

1997 - 2.0

Mail Server Installation and Integration

Honors Senior Project

Tim Lewis

Summer, 1997

Table of Contents

INTRODUCTION	1
REQUIREMENTS	1
SED EMAIL SYSTEMS INVENTORY	3
EMAIL SYSTEMS STUDY.....	3
VMS MAIL.....	3
POP MAIL.....	4
EXCHANGE MAIL	5
CC:MAIL	6
EMAIL STUDY RESULTS	8
SERVER INSTALLATION	9
BACKUP POLICIES	10
SYSTEM BACKUP PROCEDURE.....	11
RESTORING THE SYSTEM	11
USER MIGRATION	12
MIGRATING POP MAIL USERS	12
MIGRATING CC:MAIL USERS	13
MIGRATING OTHER MAIL USERS	13
PROJECT NOTES.....	13
CONCLUSION	14

Introduction

In recent years companies all over the world have begun to rely on computer networks to do their business. Network communication takes many different forms, but one of the most important is email. The goal of email systems is to provide fast, reliable communication and file transfer between individuals on a network. Unfortunately, there are many shortcomings in the currently state of the email industry. This is partly due to the fact that there are so many different approaches to providing users with email. As might be expected, compatibility problems between diverse email systems abound. At the Software Engineering Directorate of the United States Army (hereafter referred to as the SED) there is a large user community (about 800 users) that requires email connectivity. At the beginning of this project, many of these users had email of some type. Unfortunately, there were many different email systems represented in the building, and the resulting compatibility problems often caused frustration among the users. Of particular concern was the problem of email attachments. Many of the users at the SED could not send attachments to others users with any reliability. Though some users were able to send and receive attachments, even these users often had problems with email that arrived from outside the building. The proposal for this honors senior project was to migrate the SED to a single email platform without interrupting email service to any of the users. This project required much planning and forethought. The following outline shows the steps taken:

Project Outline:

- 1) Create an email requirements specification for the SED, and an inventory of the current email systems in use there.
- 2) Research available email solutions in the following areas:
 - Hardware requirements
 - Software requirements (Operating system, network software, etc.)
 - Compatibility with current email systems in use at the SED
 - User interface and capabilities (including attachment encoding and decoding)
 - System management issues
- 3) Set up and thoroughly test a candidate email solution in a small-scale test environment.
- 4) Based on the test results, begin building-wide migration to a new email platform
- 5) Write up the results of the study

Each of these goals was accomplished by the end of the project. Though many of the problems described above are still evident at the SED, significant progress has been made toward solving them. A new mail server has been chosen, successfully installed, and is currently being tested. Some of the problems users at the SED were having with attachments have been solved. In addition to the obvious benefits to the SED user community, this project also served as a case study in the field of information technology. System administrators of local area networks often need to upgrade network services on a LAN without interrupting service to any of their customers. This task has always been a delicate one. This case study will provide a guideline for smooth network service upgrades.

Requirements

What kind of email functionality do the users at the SED require? The email system chosen as the SED standard had to provide all the users at the SED with at least the same functionality they had before the new system was installed. Many users at the SED did not have adequate email services when this project began. In these cases, increased functionality was required. The following section is the original requirements list for the email system at SED. To create this requirements list, it was necessary to analyze the email tools currently in use at the SED. The functions of these tools were compiled into the requirements specification. Also, some tools that are not in use at the SED were analyzed to determine if they had capabilities that might prove useful here.

Included with this requirements specification is an inventory of the email systems currently in use at the SED. The information in this table was gathered by talking to the various email system administrators at the SED. The goal in putting this table together was to smooth the transition from the existing email systems to the new system.

Email System Requirements:

- **Internet email connectivity**

Some email systems only provide connectivity within a local area network. This is clearly unacceptable at the SED. The users at SED require the ability to send and receive email from locations all over the world.
- **Support for sending and receiving attachments**

Email attachments are becoming one of the most vital features of enterprise email systems. There are many different encoding techniques for attachments. The email package selected for the SED must provide the ability to encode attachments using all of the common encoding techniques. This insures that when users at SED can send attachments to others outside the building, they have the correct encoding technique for the party they are sending to. The selected package must also decode attachments that have been encoded using any of the common encoding techniques.
- **Connectivity with other email systems at SED**

Because the migration of users from their current email systems to the central SED email server will happen gradually, it is very important that the email package chosen for SED be compatible with the other email systems in the building. In this way, the users at SED can be smoothly migrated to the new system.
- **Directory Services**

Currently, most of the users at the SED do not have any kind of centrally administered directory service associated with their email system. Some users do not even have the ability to store a personal directory of commonly used addresses. Clearly, it is necessary for users to have at least personal directory capabilities. The need for centrally administered directory services is not as obvious, until one considers that those users who now have a centrally administered directory will probably want to keep that feature. The objective of this project is to provide an email system that will satisfy the requirements of all the users in the SED user community. Thus, a centrally administered email directory is in order.
- **Server side message storage**

Some email systems require users to store their mail messages on the mail server itself. Others require the user to keep mail messages on his personal computer. Still other packages allow the users to choose where they will store their mail. Each storage location has certain advantages and disadvantages. It has been determined that the SED user community would be best served by server side message storage. This decision was made for the following reasons:

 - 1) Server side message storage allows users to access all their mail from multiple workstations and from home. This is a feature that many SED users require.
 - 2) A regularly backed-up email server is a safer place for users to store their email than on their own workstations, which in many cases have no backup mechanism.
 - 3) Allowing users to store their email on the server will conserve disk space on their machines, many of which have small hard disks.
- **Low server maintenance**

It is very important that the email system chosen for the SED require low maintenance. The closest thing at the SED to an IT department is the network support team. The support team is small however, and has limited resources. A mail system that requires high maintenance could seriously impact the ability of the network support team to support the SED user community.

SED Email Systems Inventory

The following list is an inventory of the email systems in use at the beginning of this project, and the specifications of each.

Server Name	Operating System	Mail System	Number of Users (approximate)
Patriot Server	Novell Netware Windows NT	CC: Mail rev. 6	70
C3I Server	Novell Netware DOS	CC: Mail rev. 4	100
UAV Email	Windows NT	POP 3	100
CADM Alpha	VMS	VMS Mail	40
IFTE Server	Linux / SunOS	Sendmail / POP 3	25
SED Email	VMS	VMS Mail and POP 3	250

Not all the email systems at the SED will be converting to the new, unified system. Specifically, the Patriot group wants to keep their email server independent of ours. However, users from each of the other email systems have expressed interest in moving to a unified platform. The administrators of the C3I server are eager to hand their email system over to someone else, since they are not full time system administrators. Likewise, the administrators of the UAV and CADM servers have expressed interest in moving their users over to the new system when it is available.

Email Systems Study

This following section is the result of an extensive study of email systems. In this phase of the project, the email systems currently in use at the SED were analyzed along with several candidate solutions. In the end, it was decided that a Microsoft Exchange server was the best option for a building wide email server for the SED. This section details the findings of this phase of the project, and describes the reasoning behind choosing the Exchange server.

VMS Mail

Two of the large servers at the SED are running VMS. The default email package for VMS is VMS mail. VMS mail has a command line interface which new users generally find difficult to learn. There is a graphical interface for VMS mail that is part of the MOTIF windows system for VMS. The graphical interface makes VMS mail much easier to use. Unfortunately, the only way for a user to obtain access to the graphical interface is to either use a VAX station that supports graphics, or run an X-Windows emulator on their PC. Since very few users have ready access to a VAX station, those who use VAX mail either do so from the command line, or with an X Windows emulator. The X Windows emulator solution is not a particularly good one however, since emulators generally run slowly and use lots of system resources. Also, it is questionable whether the cost of an X Windows emulator is justified if the emulator is only to be used for reading email.

A major problem with VMS mail is that it does not support the encoding or decoding of attachments. This is a fatal flaw for an email system in today's workplace. Third party encoding and decoding software may be used to work around the problem, but this can become very time consuming and frustrating for users. So, although VMS mail supports many other email features, it is not a viable email solution for the SED today.

POP Mail

POP mail is a simple and efficient mechanism for email distribution. POP is an acronym for Post Office Protocol. POP mail is also known as "Internet mail." The majority of the SED user community uses the POP mail protocol. In a POP mail system, mail messages come into a central post office, which runs on a server. Email that comes into the post office is stored there temporarily until the mail client to whom it is addressed connects to the post office and downloads it. If the mail client never connects to the server, the mail will simply continue to collect on the server. Whenever a client does connect, the server checks to see if there is any mail for that client. If there is, all the mail is copied from the server to the client, and unless the user specifies otherwise, it is deleted from the server. The only permanent storage location for mail in the POP 3 system is on the client machines. The server is designed as a temporary message store only. This can cause a problem for users who download their mail from more than one computer. The problem is that each computer used to connect to the POP server will download the latest mail, excluding the other computers from downloading it. Thus, each computer only gets a part of the user's total mail. This can be frustrating for users who want to be able to reference all their mail at once. There are several workaround solutions for this problem. Users can leave selected mail messages on the server, but this is not the default behavior for a POP client. In most POP clients, the system is set up to either leave all the mail on the server or none of it. Thus, it can be awkward to try to leave individual messages on the server. Deleting messages can also become unnecessarily complex since they must be deleted from each individual client machine and also from the server.

Another disadvantage of POP mail is that it is insecure from a network administration perspective. The POP protocol calls for POP clients to transmit the user's account information (username and password) to the server using plain text. Also, the mail messages sent and received by a POP mail client are in plain text. While most users never see anything that occurs on a computer network unless it is destined for their own computer, there are ways of "spying" on network traffic destined for other machines. The plain text account information and mail messages used by the POP protocol is a choice target of computer hackers who want illegal access to other people's email, data, and computer resources. Furthermore, because the POP client must reconnect to the server periodically all through the day to check for new mail, the user's account information is regularly (sometimes as often as every 10 seconds) broadcast across the network. Any of the other email systems discussed in this paper are more secure than a POP mail system. In answer to this problem, some POP mail vendors have implemented encoding techniques for mail messages. There is no standard in this area however.

Most POP mail users at the SED have Eudora Light (shareware), Eudora Pro, Pegasus, or the Microsoft Exchange client with Internet mail service. Each of these packages has unique advantages and disadvantages, as described below:

Eudora Light and Eudora Pro are produced by Qualcomm, Inc. Eudora Light is a shareware version of Eudora Pro. Both programs offer POP service with a user-friendly interface and setup screens. They both allow users to specify whether or not messages will be left on the server. The user interface includes a personal directory, a spelling checker, message encryption, and many other features. While both Eudora programs can send attachments in any of the three common encoding formats (MIME, UUENCODE, and BINHEX), Eudora light has problems decoding some attachments. Also, it has a clumsy interface for retrieving attached files. This is intentional, since Qualcomm gives the product away for free, in hopes that users will buy Eudora Pro. The version of Eudora Pro tested for this study (the latest available) had no problem decoding attachments of any kind.

Pegasus is a freeware POP mail client written by David Harris. Unlike Eudora Light however, there is no commercial counterpart. Pegasus has a spelling checker, directory services, and many other nice features, and it handles attachments well, like Eudora Pro. One nice feature of Pegasus is that the user can very easily specify the encoding method for each attachment he or she wishes to send. There are a couple of drawbacks to the Pegasus client. First, many users don't like Pegasus' interface. The interface is functional, but a little clumsy. A second problem is in the area of security. Pegasus stores user passwords permanently. This allows anyone who can gain physical access to a user's computer to read the user's mail and send mail in the user's name without knowing his password.

Installing a POP server is not complicated and managing one is not a difficult either. Most of the POP servers at the SED are actually running on top of other mail systems. For example, the primary POP server at the SED is a VAX system running VMS mail. A POP server program has been installed on top of VMS mail so users can access their mail with a POP client. The POP server software uses the normal VMS user accounts and passwords. This makes account management easy, since the normal VMS account tools are used. Most POP servers work this way, allowing the operating system to manage user accounts, passwords, and even mail storage. As a result, POP server software is relatively small and uncomplicated.

Exchange Mail

The Microsoft Exchange client is a much larger communications package than the simple POP 3 mail clients. The Exchange client is also much more versatile. It can interoperate with many different kinds of mail servers, including POP servers, cc:Mail servers, Microsoft Mail servers, and of course, Exchange servers. The Exchange client uses a different information service connector to deal with mail from each kind of server. Service connectors exist for many different kinds of mail servers. The user only has to install the ones he or she intends to use. In this way, the Exchange client can be optimized to meet the needs of each individual user. In testing of the Exchange client, several problems were discovered. Like many applications, the Exchange program expects to be the user's only email program. It doesn't operate well with some other email clients. Also, each of the information service connectors work well alone, but when more than one service connector is simultaneously installed, it can cause problems. The Internet mail connector (POP3) seems to operate well with other information services. However, serious problems arose when both the cc:Mail connector and the Exchange connector were installed together. Under this configuration, the Exchange client would crash after only a few minutes of running. Fortunately, there is very little need for anyone to use both of these connectors together. Another possible problem with Exchange comes from the client hardware it supports. The Exchange server comes with client software for PCs running all versions of Microsoft Windows and DOS (as might be expected), but it does not support Macintosh computers. Only a very small percentage of the SED community uses Macintosh computers, so this may not create a significant problem. Users who want to use Macs could still access an Exchange server using a POP client.

Many of the users at the SED use the Exchange client. Currently the only Exchange servers in the building are testbed systems, and have very small numbers of users. The Exchange clients used at the SED are used with the Internet service connector to connect to the various POP 3 servers here. Reports from the users who have used Exchange for POP 3 access are positive. The only problem found upon examination of the POP 3 client was that it does not give users the option of leaving their mail on the POP server. This is a serious problem for users who want to download their mail from multiple workstations, but for most users it does not make much difference. When the Exchange mail client is used with an Exchange mail server, it provides the user with many more features. These include the following:

- 1) Centrally administered directory services and mailing lists
- 2) "Bulletin boards" for posting public notices
- 3) Both server side message storage and client side storage, as the user desires
- 4) Access to Internet news groups

The exchange mail system provides attachment encoding and decoding services in all of its forms. One nice feature of Exchange is that attached pictures appear inline in the mail file. Exchange also provides a way for users to password protect their personal mail folders so that even if someone gains access to their personal workstation, their mail will be safe.

The Microsoft Exchange Server is feature-rich mail server, and as a result is also very large and complex. The Exchange mail server can act as a POP server as well as an Exchange server. It provides gateways to cc:Mail, MS Mail, and other common mail systems. It also has an optional feature called web mail. The web mail component of Exchange works with the Microsoft Internet Information Server. It allows users to read and write email directly over the World Wide Web, with only an Internet browser as a client. The administration tools provided for Exchange server are well designed, with graphical interfaces and hierarchical menus that help deal with the complexity of managing the server. In spite of the complexity of the Exchange server, setting it up is easy. The system installs and configures itself painlessly. Any change to the installed configuration requires some study on the part of the administrator, however. One feature of Exchange administration that is something of a mixed blessing is the way user accounts are handled. Exchange users have accounts both in the Windows NT account manager and in the Exchange recipient list. The two accounts are synchronized somewhat through Exchange, but not completely. For Exchange users, this is good because it means that a single username and password is all they need. Also, the security of the Exchange accounts is the same as that used for Windows NT logins, which uses encryption for all account information. For the administrator however, the management of two different user account lists can be a lot of trouble. Over all however, the Exchange mail system appears to meet all the requirements for an email server for the SED.

cc:MAIL

The cc:Mail package is produced by Lotus, and is very similar to the Exchange mail package. cc:Mail is a much older product than Exchange, and as a result there is a large base of cc:Mail users all over the Redstone Arsenal, including at the SED. There are two cc:Mail servers at the SED, one maintained by the Patriot project, and the second maintained by the C3I project. These two servers account for about 30% of the total mail users at the SED. On the Redstone Arsenal as a whole, cc:Mail is very common. There is a centrally administered cc:Mail system on the Arsenal that provides users with public bulletins and Arsenal wide directory services. The cc:Mail post offices in each building can connect to this central mail server to receive bulletins and directory service, and to rout mail to other cc:Mail servers.

There are several different cc:Mail client programs. This is another result of the fact that cc:Mail has been around for some time. Each new version of cc:Mail server that Lotus produced came with a different set of client programs. For this study, version 6 and version 8 (the latest version) of the cc:Mail client software for Windows were reviewed. These two clients were functionally similar, though the look and feel of them was different. Unlike Microsoft Exchange, the cc:Mail clients are designed to communicate only with a cc:Mail server. They do not communicate with any other kind of email server. Unlike the Exchange however, cc:Mail comes with client software for both Macintosh computers and PCs. The cc:Mail system provides users the same important features that Exchange mail provides: access to Internet mail, attachment encoding and decoding, centrally administered directory services, both server side and client side message storage, and electronic bulletin boards for posting public messages.

The chief drawback of the cc:Mail system as a whole is that it seems to have been designed to communicate primarily with other cc:Mail systems. It does not seem to have been designed with interoperability in mind. Although communication with other mail systems and with the Internet is available, it is clumsy. For example, if a user wants to send mail to someone else using a connected cc:Mail post office, addressing the mail is easy: The user can either pick the recipient off the cc:Mail directory list, or begin typing the recipient's name in a blank. The cc:Mail client will actually guess the recipient's full address before the user has to finish typing it. The whole process has been streamlined. To address a letter to someone on the Internet however, requires more work. The user cannot simply type an Internet address in the recipient blank. The cc:Mail system does not recognize any addresses other than it's own. Instead the user has to specify that the following address is an Internet address. This is done by

selecting "Internet" on the cc:Mail directory tree, and then specifying the Internet address in a separate recipient window. The process is still fairly painless, but it illustrates the mentality of the cc:Mail system. A better example perhaps, is the problem with cc:Mail's mailing lists. If a user wants to create a mailing list where all the recipients are cc:Mail users, there is no problem. The process is quick and painless -- a simple matter of selecting the desired users from the primary cc:Mail directory. The mailing list can then be saved for later use. But if any of the recipients on the list are Internet mail users there are problems. The Internet users have to be added separately from the cc:Mail users, using separate entry windows. Also, mailing lists with Internet users cannot be saved, and so they have to be recreated every time they are needed. This is so frustrating for mail users that the cc:Mail administrators at the SED have added large numbers of Internet mail aliases to their cc:Mail directories to work around the problem.

Two different versions of the cc:Mail server were installed for this study, version 6 and version 8. Version 6 installed without very much trouble. Unfortunately, version 6 of cc:Mail does not come with an Internet mail connector. The Internet mail connector (and connectors for any other mail system) must be purchased separately. This is yet another example of how the cc:Mail system appears to be designed to communicate only with itself. As it turns out, only version 8 of cc:Mail (the latest version) actually includes the Internet mail connector with the system. The version 6 server software works well, but has a few drawbacks. First, much of the software seems to have carried over from prior versions of cc:Mail. There is no graphical interface for administrating the cc:Mail system. Instead, the administration tools run only under MS DOS, and use either a command line interface or a character-based menu system. Much of the software is 16-bit, and thus does not take full advantage of the power of modern 32-bit processors. Even so, the administration of the cc:Mail post office is not terribly difficult. Creating user accounts is easy. The Internet mail connector for cc:Mail version 6 was not available for installation, but from the study of cc:Mail version 8, this might have proved difficult.

Installing cc:Mail version 8 turned out to be very difficult. An attempt was made to install it on a PC running Windows NT server. Unfortunately, certain components always failed to install. After much time spent trying to work around the problem, the cc:Mail technical support group was contacted, and using their assistance the product was finally installed. It turned out that there was a bug in version 8 that had to be worked around. After this initial difficulty, the installation went smoothly and the server came up without a problem. Like Exchange server, cc:Mail v.8 provides web mail access and includes a POP mail server. The version 8 server has a graphical administration tool and automatically sets up and schedules post office maintenance. The graphical administration tool is not as well designed as could be desired, however. One major drawback is that the window is small and fixed in size so that only a small amount of information can be viewed at a time. However, account creation and management is still straightforward. Unfortunately, some of the version 8 software is still 16-bit code. It is assumed that Lotus does this for backward compatibility.

There were a couple of other problems with the cc:Mail system as a whole. Unlike Exchange, the cc:Mail system maintains its own separate user account list. This is bad for both users and system administrators. It's bad for the users because every user has to have two accounts: one to gain access to the mail system drive shares provided by Windows NT, and the other to actually use the cc:Mail system. It's bad for the mail administrator because (as with Exchange mail) they have two separate account lists that must be continually synchronized. Another problem discovered after installing the version 8 server on the PC was resource usage. The computer used for testing the cc:Mail server has 64 megabytes of physical memory and the same amount of virtual memory. With cc:Mail server v.8 installed and running, this memory space was about 90% used. This kind of resource usage seems excessive for a testbed mail server with only five users. Finally, it was discovered that the cc:Mail router and Internet gateway were running as user processes. This meant that whenever the current user logged out of his Windows NT session, the mail router and Internet gateway stopped running. They would not be restarted until someone logged in again. This of course, is unacceptable. The mail gateways must run continually. It is clear that the cc:Mail authors intended for the mail administration user to be continually logged in, so that the gateway software would always run. But this is not acceptable either. It is a security risk, and potentially a violation of privacy. Leaving the mail administrator logged in all the time grants administrative privilege to anyone who has physical access to the mail server. An attempt was made to work around this problem by installing the cc:Mail router and Internet gateway as NT services. This would have made them detached

processes that run regardless of who is logged in. This effort failed however. The processes cannot run detached because of the way they are written. In spite of these problems however, the cc:Mail system appears to meet all the requirements for an email server for the SED.

Email Study Results

After reviewing the email systems listed above it was clear that either cc:Mail or Exchange mail would satisfy the requirements for an SED email server. Each had certain advantages and disadvantages, but it was the conclusion of the author that Exchange Mail was the best choice. This conclusion was evaluated and discussed with other member of the network support team, and with email system administrators both at SED and elsewhere. The consensus was the same: that Microsoft Exchange was the best email platform for the SED. Some of the reasoning behind this choice has been elaborated below:

- Internet-friendliness

As has already been discussed at length, the cc:Mail system simply not well designed for connectivity to other email systems. Though the capability is there, the Exchange system is a much more elegant solution.

- Security

The problem with cc:Mail requiring an "eternal login" is not insignificant. The users at the SED need a secure email system, and the Exchange mail system provides much better protection than cc:Mail does.

- Efficiency

In addition to the problems already cited concerning the resource inefficiency of cc:Mail version 8 and the 16 bit code that it uses, an additional factor made Exchange mail more attractive. The server chosen for the SED email system is an Alpha 2100 running Windows NT Server. The Alpha can run binaries for Intel-based PCs, but it has to run them through an Intel emulator. While this emulation software is freely available, it is preferable not to use it. This is because binaries that are native to the Alpha processor run much faster than Intel binaries run through the emulator. As it turns out, Exchange mail comes with Alpha binaries on the same CD as the Intel binaries while cc:Mail must be run though the emulator. Thus, the Exchange Alpha binaries are greatly preferred.

- Migration path

Finally, it has already been stated that many of the users at SED already use the Exchange client to access POP mail accounts all over the building. A significant number of users also use the cc:Mail interface. Evaluating which migration path would be the smoothest for users is fairly difficult as a result. To move users from the Exchange POP client to the Exchange mail client is a painless task. The user's old mail messages and folders are left intact, and the new mail system capabilities are added. Likewise, to upgrade users from one cc:Mail server to another is normally a simple matter of reconfiguring the client software. However, in this case it is much more difficult.

The cc:Mail servers at the SED are both using Novell networking to provide their users with drive storage space and email connectivity. If cc:Mail were installed as the primary SED email server, it would be done on the LAN Manager network system, not the Novell system. This would require a complete overhaul of the network software on any machine that moved from one of the current cc:Mail systems to the new cc:Mail system. So, it was clear that the cc:Mail clients in the building would have to be overhauled no matter which server was chosen. However, the Exchange clients would not need to be reinstalled if the Exchange server were chosen. Additionally, most of the cc:Mail clients in the building use the Patriot server, and are not expected to move to the new email server.

For all of these reasons, it was decided that the Microsoft Exchange server was the best choice for the SED primary email server.

Server Installation

Microsoft Exchange must be installed on a computer running the Windows NT operating system. Both Windows NT and Exchange come with binaries for both Intel-based and Alpha-based computer. The platform chosen for the Exchange server is an Alpha server 2100/5 with four processors and 256 megabytes of memory. It has five SCSI hard disks, each having two gigabytes of storage, a CD ROM drive, a floppy disk drive, and a 4mm DAT tape drive. The Alpha has two SCSI drive controllers. One is a normal SCSI controller that treats each device separately. The second is a SCSI RAID controller¹. Briefly, a RAID system provides a way to speed hard disk access and recover data when a disk crashes. RAID is an acronym for Redundant Arrays of Inexpensive Disks. There are several ways to configure a RAID system. One of the simpler configurations is called a mirror set. There are two disks in each RAID mirror set. Both disks in the set store exactly the same information. Thus, they are "mirror images" of one another. All read and write operations are done on both disks so that the mirroring is maintained. If one of the disks in a mirror set fails, all the information can be recovered from the other one. The disk that failed can be replaced and all the drive contents copied to the replacement disk so that the mirror set is restored.

The Alpha Server used in this project was formerly used for other applications, and its initial configuration was inconsistent with the project's requirements. The Alpha was originally configured for use with two operating systems: Digital UNIX and Windows NT version 3.51. It also came with three network cards. Three of the six original disks were connected to the RAID controller for use with the UNIX operating system. The other disks were connected to the regular SCSI controller for use with Windows NT. The hardware was reconfigured so that all the disks were connected to the RAID controller, and arranged in two mirror sets and an extra disk. The sixth disk failed shortly after it was received. Configuring the remaining disks involved learning to use the RAID setup utility and the setup programs stored in the Alpha's ROM. Next, the firmware on the Alpha had to be upgraded. The firmware that came on the system was incompatible with Windows NT version 4.0, the operating system chosen for the Exchange server. Again, special firmware upgrade software had to be loaded using the Alpha setup programs. After the firmware was upgraded, the drives had to be configured so that the Windows NT system could be installed. This turned out to be something of a challenge.

Windows NT boots in two phases. In the first phase it loads a program called "osloader" that does nothing but load the rest of the Windows NT operating system into memory. On an Alpha system, the "osloader" program and its associated startup files must be contained on a disk partition called the system partition. The rest of the Windows NT operating system can be stored on the system partition or on a separate partition that is then called the boot partition. System partitions are created using a special tool that comes with Windows NT. In the first attempt at installing NT, one entire drive was set up as a system partition with the intent of installing the entire NT operating system there. But as it turned out, the system partition can only be formatted for the file system called FAT. FAT is the outdated file system that the original DOS PCs used. It does not support long file names, and more importantly, it does not support any kind of file security. Since security is very important on an email system, it was considered essential that the Windows NT operating system be installed on a drive partition that used the NTFS files system. So, the operating system had to be split across two partitions. The system partition had to be FAT and the boot partition had to be NTFS. The system partition needs only to hold the "osloader" program and a few other setup files, so it does not have to be large. The system partition created on the Alpha was only six megabytes, and the boot partition used the rest of the two-gigabyte disk. Further problems developed when Windows NT was installed on these partitions due to the way that the Windows NT drive letters were

¹ The Windows NT operating system supports software-based RAID that provides the same kind of error recovery provided by this RAID controller. The reason that the RAID controller was used for this application is that the RAID hardware is faster than its software equivalent.

assigned to the partitions. The partitions had to be recreated in yet another configuration, and Windows NT reinstalled before the system became stable.

Installing Windows NT and Microsoft Exchange is not difficult once the drives are correctly configured. The system was set up so that the operating system resides on one RAID mirror set and the data files for Exchange (which can be quite large in high-use mail system) are stored on the other mirror set. The networking software for Windows NT uses only one of the Alpha's original three network cards. One of the cards was removed, and the remaining card was kept as a backup in case the working card fails.

Backup policies

The RAID controller in the Alpha Server provides failure recovery when disk storage hardware fails. But what if the data on a RAID mirror set is corrupted by a program? What if a virus infects the system? In cases like these, the RAID system will not help because the bad data will be written on both disks in the mirror set. A different kind of backup is required. There are many different kinds of backup devices. The Alpha Server came with a 4mm DAT tape drive. The 4mm DAT tapes hold two gigabytes of information. This is a reasonable backup mechanism, since each RAID mirror set that needs to be backed up has a maximum of two gigabytes of storage. Learning to back up a Windows NT system is more complex than simply putting a tape in the drive, however.

The first question that must be considered in construction a backup policy is the relative importance of the information stored on disk. In an Exchange email system, the Exchange post office data is the most important information on the system because it is irreplaceable, and in a continual state of change. The Windows NT operating system and Exchange mail software is certainly important, but there is no chance of losing it completely. All of that software is on CD-ROM and can be reloaded (in the worst case scenario) if it becomes corrupt. Certain parts of the operating system are also very sensitive. For example, the system account information and the configuration information for the disks and other hardware cannot be reloaded from the CD-ROM. This information does not change as often as the mail post office data however, so backups of this information do not have to occur as regularly. Furthermore, the hardware configuration and even the system accounts can be reconstructed from scratch if need be.

There is tape a backup program that comes with Microsoft Exchange. It is called "NT Backup," and it is an extension of the backup program that comes with Windows NT. The primary difference between the two programs is that the Backup that comes with Exchange is "aware" of the Exchange server. The normal backup software allows the user to back up drives, directory trees, and/or specific files. The Exchange version adds the capability to back up all the Exchange post office data files with a single command. One common problem with making tape backups is that files which are in use by the system cannot usually be backed up. The Exchange post office backup procedure has been specially written to avoid this problem, however. All the Exchange data can be backed up while the system is still running. The backup task can be scheduled using the Windows NT scheduler service so that the entire Exchange database is backed up at regular intervals. The system administrator does not need to get involved except to insure that the right tape is in the tape drive.

Actually configuring the Windows NT scheduler to perform the necessary backups turned out to be a more difficult task than was anticipated. Configuring the scheduler to run a task at a give time of the day or week is not hard. The difficulty is that the task to be run can only be specified on a command line. The command line interface for the normal Windows NT Backup program is well documented, but there are special command parameters for backing up the Exchange data that are not in the original documentation (since they were not part of the original software). These commands are documented in the Microsoft technical information database and on Microsoft's Internet newsgroup. Once the correct commands were obtained, some simple experimentation with the command line interface was all that was required to set up the scheduler.

While the Exchange database information is the most critical to back up, the rest of the system must be backed up as well. Though most of the system can be rebuilt from CD if necessary, it is certainly much more efficient to restore it from tape. Installing a system from scratch takes a lot of time, and all the settings have to be configured. Restoring a tape backup takes less time, and all the settings are restored with the backup. There are problems with backing up a Windows NT system, however. Unlike the Exchange backup process, the operating system cannot be fully backed up while it is running. Thus, in order to do a full system backup, the system must be shut down as far as possible without disabling its ability to run the tape drive. In Windows NT, this is done by shutting down all the operating system services. Some operating system services cannot be shut down, however. So, even with the minimal system running, some files do not back up correctly. Fortunately, these files can be completely restored from CD ROM if they become corrupt. So, between the CD ROM and the tape backup, the entire system can be restored. Also, in most cases where the tape backup would have to be restored, the one or two files that could not be restored would not be corrupt anyway, and so the CD ROM would not have to be used.

Restoring a tape backup of the entire Windows NT system is a measure reserved for catastrophic system failure. There are less extreme solutions for problems that are not that bad. One of the most important information stores on a Windows NT machine is the system registry. The registry holds almost all the configuration information for the operating system. Another important information store is the drive configuration parameters. These information stores can be backed up and restored individually. They are small, and so the backups are done to disk rather than to tape. Thus, there are three different kinds of backups:

- The Exchange database backup
- The entire system backup (all drives in the machine)
- The registry and disk configuration backup

The Exchange database backup is currently performed automatically, every night after work hours. The other two backup procedures are not scheduled automatically. The system backup only has to be done when the system changes -- when software is installed or upgraded. The registry backup has to be done whenever the system settings change. Since the system settings include the user account database, this may happen quite often. The current plan is to perform a complete system backup once a week, late in the evening. It would be better to be able to schedule the backup procedure to occur automatically over the weekend, but this is currently impossible because the full backup requires a user to switch tapes and disks. In the future a second tape drive may be ordered for the Alpha, which allow the backup procedure to be fully automated. For the present time then, the following backup procedure will be followed:

System Backup Procedure

- Stop all Windows NT services that can be stopped
- Use NT Backup to back up the system disk (drive C) to tape.
 Include the NT registry in the backup.
- Use NT Backup to back up the data disk (drive D) to tape.
- Recreate the Windows NT Emergency Repair Disk using the RDISK program
 Use the /s option on RDISK to back up the user account database also.
- If the Drive configuration has changed, use the Disk Administrator program to save it to a floppy disk.
- Save the System and Disk registry keys to a floppy disk using the Registry Editor program.
 (This is done so that these keys can be restored separately if necessary)
- Reboot the machine to restart all services.

There is no simple way to write a set of instructions for restoring a corrupted system, since there are so many different things that can go wrong. But, a general guideline for restoring the system from a catastrophic failure is as follows:

One of the tests performed on the POP clients that were studied for this project was to send them attached binary files. This test was designed to confirm that each POP client could successfully encode and decode messages that used the three common methods of attachment: UUEncoding, BINHEX, and MIME. All of the POP clients tested could successfully encode and decode messages that used UUEncoding and BINHEX. Decoding MIME attachments was a problem for all of them, however. All of the programs were able to encode attachments using MIME, but none of them could decode them -- not even the very attachments they had created. This seemed suspicious, so further tests were performed. All the initial testing had been performed using the primary POP server at the SED and the primary SMTP gateway. When different servers and gateways were used, it was discovered that MIME attachments decoded successfully. Test messages including MIME encoded binaries were sent to two different POP servers using four different SMTP gateways. When the received test messages were gathered and analyzed, it was observed that all the messages that failed to decode had their content headers at the bottom of the message, while those that decoded successfully had their content headers at the top. The problem was finally traced to a setting on the primary POP server for SED, which had a setup parameter allowing the administrator to specify that headers should appear at the bottom of messages. This option had been selected for the convenience of the users. If the headers were at the top of each mail message, the users had to scroll past them all to see the message text. At the bottom, the headers didn't get in anyone's way. However, this also prevented MIME messages from decoding. As a result, the setting for headers was changed so that MIME messages would decode correctly. This incident, though not directly related to the project, helped many of the users at the SED who were using MIME attachments.

After the Exchange server was brought up the first time, it had to be tested by members of the SED network support team. So, to make the system easily accessible, it was placed in the network support team's NT domain. The system was observed to work well in this environment, but problems arose when a user from outside the network support domain was added to the system. In the prior section detailing the email systems study, the unified login feature of Exchange was discussed. Briefly, the Exchange server uses the regular Windows NT account database to verify its users. This is an advantage to mail users because it means they do not have to log in twice -- once for the operating system and again for the email system. The problem is that all the users who want to use the Exchange server have to have accounts in the same domain as the Exchange server. Since the Exchange server is designed to be used by everyone in the Software Engineering Directorate, the NT domain that hosts it will have a lot of user accounts. This is unacceptable for the network support domain because of security and administrative concerns. So, the Exchange server had to be moved to its own domain. This meant that the Alpha had to be changed from an NT domain member to a domain controller. But there is no way to modify an NT system to make it a domain controller. As it turns out, the system has to be completely rebuilt. So, the server was taken back down again, and rebuilt from scratch. The resulting system is much better, since the user account management will be kept separate from all other systems. This makes account management simple and straightforward.

Conclusion

The Microsoft Exchange server is currently in its testing phase. Certainly there have been some unexpected problems with the server, but so far all the users that have tested it are pleased with its performance. Clearly, there will always be users and projects that want to use their own mail systems and servers, but availability of a modern, unified mail system will still benefit the organization greatly.

There are several ways that the current email system can be upgraded in the future. One of the most important upgrades will be remote mail retrieval. Currently, it is possible for users to access their mail while they are away from the SED, but it is not easy. The SED has a modem dial-in server, but currently it is not set up to support Point to Point Protocol, the protocol that gives a PC direct network access. As things stand, users must get a PPP account through the Redstone Arsenal's central information services group, or through one of the other project groups at the SED. To solve this problem, the current modem dial-in server may be modified to allow PPP access. Alternately, some modems may be added to the Alpha Server to allow dial-in access there. This will allow users who travel with their computers easy

access to their email. This will probably be the next major upgrade project for the SED email system. Another upgrade for the email system that is being considered (as was discussed previously) is the addition of a second tape drive. This will enable the full system backups to be performed as a scheduled task, at hours when no one will be using the system. Finally, there has also been some discussion about running an Internet web server on the same Alpha Server that the Exchange server is on. The Alpha should be powerful enough to handle the web server in addition to Exchange. Also, there is a possibility that this will enable users to access their email over the Internet when they are at a remote location, depending on how the Internet services for SED are set up.

The application of this project to other networks is fairly straightforward. The features of cc:Mail, Exchange mail, and the POP mail systems are all discussed in detail. The findings of this study show why Exchange is a preferable email server for this particular network, but administrators of other networks can use the same information to evaluate these systems for their networks. The information about the process of installing the Exchange server, and the problems encountered is applicable to other installations of Exchange server. The backup policies described in this document are applicable to any network server system, but especially to Exchange servers. Beyond all the technical details and problems, this project demonstrates the overall process of installing a network server so that it is fault tolerant. It is amazing how often servers are set up with no backup capabilities whatsoever. It is only after something breaks that some system administrators start looking at the fault tolerance features of their server. Often by that time however, valuable information is lost. Even in the case where no information is lost, the recovery process usually takes much longer if no preparation has been made for system failure. In a business environment, system downtime can cost a lot of money. So it is important to prepare for system failure. It is hoped that the information and example provided by these documents will aid other system administrators in making good decisions about their networks.