Large Installation System Administration

Makealiases - a mail aliasing system

Gretchen Phillips - SUNY@Buffalo
Don Gworek - Sun Microsystems

September 1988

1. Introduction

The State University of New York at Buffalo has an ever growing Unix based computing environment. The primary instructional Unix machines now include a Vax 11/785 and a Sperry 7000/40 both running Berkeley 4.3 as well as an Encore Multimax running Umax 4.2 Release 3.1. Additionally, Sun workstations are beginning to proliferate on the campus. Our Unix users range from freshmen who have accounts so that they can "learn the system" to graduate students in Computer Science and Electrical and Computer Engineering who are primarily interested in operating systems and networking.

The total number of unique usernames is over 1000. This is broken down to about 500 undergraduates, 120 faculty and staff, 50 system logins, 25 support staff, 50 miscellaneous accounts and the remainder (about 300) graduate student accounts. Users are assigned accounts based on their particular needs and the ability of the machines to accommodate users. Users needing specific products (e.g. macsyma) are assigned to the machine with that resource. Users needing general access are assigned to the most lightly loaded machine. This may change from semester to semester depending on what classes are offered and the research projects that are particularly active. Nonetheless, users still need to receive their mail in some reliable, consistent (and easy to maintain) fashion. To this end we have developed a program *makealiases.*

2. Philosophy of *makealiases*

When users have multiple accounts on many machines within a domain it is possible for them to receive mail on any (or all) of the machines. *Mail* provides the feature of a *.forward* file for mail forwarding. This is not a satisfactory method for mail delivery considering that nearly one third to one half of our Unix users are neophytes. They are primarily concerned with getting their programs edited and running and not with where their mail is being delivered. Additionally, it was possible for a student to have accounts on one or all machines depending on their class enrollments. This means that mail would be delivered in, what appears to them, some random manner.

We decided that it would be easiest to generate an aliasing system within the domain that would assign a single machine as a users primary (and only) mail drop. All mail sent from within (an from outside) the domain would be delivered to one machine for each individual. The machines would be assigned a priority for delivery for each group. This allowed faculty to receive their mail on the machine that they do their research and undergraduate to receive mail on a machine that they would be doing their class work. If a user has accounts on multiple machines this priority system decides where to deliver the mail. If a user only has a single account, then their mail is delivered to that machine.

One significant problem that this overcame was that the sender does not need to be concerned with knowing where an individual reads their mail, or even on what machines they have an account. All accounts within the domain can be addressed simply by the username. Perhaps this situation is a common one, but we are satisfied with our solution.

3. *makealiases* implementation

Makealiases updates and distributes /usr/lib/aliases to all machines in a local network. It provides a single mailing address for users with multiple accounts, and expands special group aliases that are local to a host. *Makealiases* should be run after new accounts are created, to keep /usr/lib/aliases up to date. *Makealiases* should be run on only ONE host in the network.

3.1. Format of *aliases* file

The postmaster makes changes to system mailing addresses in the file /usr/local/lib/mail/aliases. This is a copy of /usr/lib/aliases. The file *aliases* should be organized in four sections:

(1) Distribution and priority information should be at the start of the file. Each of these lines begins with a '#', followed by the word **dist** or **priority.** For a **dist** line, the second word is **special, hosts** or **yphosts. special** is for hosts that do not need the aliases file to be installed and do not want the mail list. It merely gets a copy of the aliases file. **hosts** defines all hosts that are to get a copy of the final aliases list. Any hosts preceded by a **!** does not have it's /etc/passwd file sampled and therefore will not have any mail delivered to it. **yphosts** defines hosts that have yellow page listings that must be updated. **priority** lines define what order mailboxes are to be assigned depending on the prefix specified in the line. If a username is not prefixed by a priority prefix, then the **default** definition is used. This is the order in the **hosts** list if a **default** is not defined. Only hosts in a priority list are required to be up for a run. Information from other hosts are used only if they are up. This section is ended by the line

MULTI-USER ADDRESSES ###.

(2) Multiuser aliases and special groups should be listed between the

MULTI-USER ADDRESSES ### and the ### FORWARDING ADDRESSES ### line. *Makealiases* expands the special groups for the specified host.

(3) Forwarding addresses. Users who desire mail on a specific local host, or users who receive mail nonlocally, should be listed between the

FORWARDING ADDRESSES ### and ### AUTOMATED_BEGIN ### lines. This section of the file is optional.

(4) Automatically generated individual aliases appear after the ### AUTOMATED_BEGIN ### line. After this point, *makealiases* appends aliases for users who receive mail on remote machines.

3.2. Other related files

The postmaster can specify expansions for special groups local to a host in *specialgroups.host* files found in the group directory. Special group files are optional. These expansions usually include a local archive file name, or a pipe to a local program. In general these are addresses not desired on all hosts. The syntax and format of the expansion is the same as a normal mailing alias. Previous copies of /usr/lib/aliases are located in the archive directory.

3.3. Running *makealiases*

When *makealiases* runs, it first checks that all required hosts are up. Then it forks off a child to run in the background, and the parent dies. The child collects all usernames from the hosts. And then each host is individually updated with the a new aliases file and a new /usr/lib/mail/mail.local. (This is a file listing all users, where their accounts are, and where their mailbox is located.) After all hosts are updated makealiases sends a letter to the user of the program, to the postmaster, and, locally, to a special group *makealiases.* This is a mail-record file of all runs of makealiases. The letter contains a transcript of the run, including the differences that were found between /usr/lib/aliases and the seed *aliases* file, and the output from the *newaliases. Rmakealiases* is a shell script run by cron once a week to update /usr/lib/aliases and /usr/lib/mail/mail.local on nonequivalent hosts (locally, our Sun workstations). *Cronmakealiases* is a shell script that is run nightly by cron. If there is a difference between /usr/lib/aliases and /usr/local/lib/mail/aliases it then runs *makealiases.* With *cronmakealiases,* changes to aliases can be made in the daytime, and *makealiases* run at night when load is low.