Intro to Unix Mail

<colleen@julian.uwo.ca>

What is Electronic Mail

- E-mail
- Communication Tool
- Text Files
    - personal communication
    - send and receive information from a class
    - review papers
    - small discusion groups
- Electronic Mailbox
- Advantages
    - speed
    - avoid telephone tag
    - cost
- Limitations
    - recipient must use e-mail
    - NOT for large or binary files

Some Mail Programs

- on julian we have:
    - mail (ucb)
    - mailx
    - mh, xmh
    - emh, xemh, email
    - elm

Into to Unix Mail

Objectives

- learn basic commands to send and receive messages
- other mail commands to organize and customize
- where to find help — man pages, **?** command, **~?** command, postmaster
- how to register a name with the Campus Email Directory

Course Outline

- addresses
- create & send message
- check for mail
- get out of mail
- read
- respond
- store
- delete
- types of mail commands
- customize mail program
- alias
- folders
- files used by mail program
- registering with **whois**

Addresses

- where to send mail
- NO spaces
- NOT case sensitive
- see **man mailaddr**

```
user@domain
user@julian.uwo.ca

swatson@julian.uwo.ca
swatson
```

- where to obtain addresses
- contact by other means first
- Name Registry — campus database of local email addresses

    `whois` ***name***

Sending

`mail` ***address***

`mail swatson`

- will be prompted for **Subject:**
- then enter text, pressing return at the end of each line
- terminate message with period (**dot**) in first column (or use **Ctl d**)
- will be prompted for **Cc:**

Sending to Several People

`mail` ***address address***

`mail swatson postmaster`

Delivery Errors

- when error in local address you may see:

mac0... User unknown: No such file or directory

- errors in remote addresses
- original message returned along with error message
- **Mail-Daemon**
- postmaster — contact for help, problems, questions
- **postmaster@julian.uwo.ca**
- **postmaster@uwo.ca**

Checking for Mail

- **/usr/spool/mail/user** is your mailbox — where mail is delivered
- notified of new mail with message:

  `You have new mail`

- for list of messages type:

  `mail`

- mail program prompt is **`&`**
- any new messages arriving while you are reading mail are NOT automatically incorporated — when you leave mail program you will be notified:

  `New mail has arrived.`

Message Headers

- message number
- sender's address
- date, time
- length (line/characters)
- Subject
- current message pointer
- message status (N, U, *****)
- list commands
    - **headers (h)**
    - **z**
    - **z –**

Getting Out of Mail

exit (ex) (x)	immediate return to shell without modifying mailbox. NO changes made.
quit (q)	terminate mail session and return to shell. normal method.

Reading Messages

Commands are:

- message number
- **print (p), Print (P)**
- **type (t), Type (T)**

At the **&** prompt, type:

message

`print` *message*

`type` *message*

`Print` *message*

`Type` *messsage*

Note Message Headers:

- To:
- From:
- Subject:

Long Messages

- messages longer than one screen will use **more** for display
- at bottom of screen you will see:

  --More--

- **space bar** will display the next screen of the message
- **return key** will display the next line of the message
- **q** will stop displaying the current message and return you to the **&** prompt

Responding to Messages

- send mail to sender and ALL recipients of original message
    - **respond (r)**
    - **reply (r)**
- send mail ONLY to originator and does not send to other recipeints of original message
    - **Reply (R)**
- inserts original subject prefixed with **Re:**

Storing Messages

- remain in **/usr/spool/mail/user** if not **delete**d
- can also store a copy of message in a file
- **save (s)**
- **copy (co)**
- **write (w)** — no headers
- **`*`** — status in header list after **save** or **write**

`save` ***message filename***

`save 15 my.let`

`write` ***message filename***

`write 7 newprog.f`

Message Lists

- allow mail commands to operate on many messages at once
- if no message list specified mail command uses **current message**

```
save 1 4 6 my.memos

save 1–5 my.memos

copy * my.memos

copy $ last.note

write 25 newprog.f

write newprog.f
```

Deleting Messages

- clean up
- **delete (d)**
- message will not longer appear in header list
- message will be deleted on **quit**ting
- **undelete (u)**
- restore message before **quit**ting
- message will appear in header list again

`delete` ***message_list***

`delete 12–18`

`undelete` ***message_list***

`undelete 14`

Mail Commands

Mail Commands	Used at the **&** prompt when inside Mail to read and handle messages. Commands are case sensitive and can be abbreviated.
Tilde Commands	Used when inside message. Begin with **~** (tilde) and allow you to escape from composition to enter command.
Set Variables	Used to customize Mail. Varaibles are enabled with **set** command and usually found in **.mailrc** file.

Tilde (~) Escape Commands

- inside composition mode
- begin with tilde (**~**)
- tilde at left margin
- followed by another character
- allow you to escape from composition
- will not appear as part of message
- message lists can be used

Some Tilde Commands

~p	print (display) current message
~e	create text with your choice of editor
~q	abort current message and write it to dead.letter file
~r filename	insert file into message
~f message	insert an existing message into current message
~m message	insert an existing message, right shifted one tab
~h	add information to message headers
~?	list all **~** commands

Customizing the Mail Program

- **/usr/lib/Mail.rc** system configuration file
- **set**
- can override with your own **.mailrc** file of **set variables**
- also a few mail commands in **.mailrc** file

```
& set

DEAD    ~/dead.letter
EDITOR  /usr/local/bin/emacs
PAGER   more
SHELL   /bin/csh
VISUAL  /usr/local/bin/emacs
alwaysignore
append
ask
askcc
asksub
autoprint
cmd     mp | lpr -PPostScript
crt     24
dot
hold
indentprefix    >
keepsave
metoo
prompt  Mail>
screen  24
toplines        10
```

Set Variables

- used at the **&** prompt
- usually placed in **.mailrc** file
- controlled with **set** and **unset**
- valued – set to values
- boolean – set to on or off
- can have several variables on 1 line

`set` ***variable_list***

`unset` ***variable_list***

```
set toplines=5
```

Aliases

- placed in **.mailrc** file
- define abbreviation or simple name
- represent several of users
- avoid typing long addresses
- use more meaningful names
- **alias**

`alias` ***alias_name address address***

`alias team swatson ted@uwo.ca`

`alias me sue@julian.uwo.ca`

Folders

- **/usr/spool/mail/user**
- **mbox**
- **.record**
- file of **save**d messages
- folder format
- mail commands at disposal to handle messages
- needs header information therefore CANNOT be created using **write** command

Looking at Folders

- From the **julian(1**:) prompt:

  `mail -f` ***foldername***

  `mail -f nov.folder`

  `mail -f`

  `mail`

- From inside mail program use **folder (fo)**, or **file (fi)** command. At the **&** prompt:

  `folder` ***foldername***

  `folder nov.folder`

  `file nov.folder`

  `folder %`

  `folder`

When You Are Away

- forward your mail automatically to another user
- **.forward** file in your home directory

`ted@uwovax.uwo.ca`

Files Used By Mail

- dead.letter
- /usr/spool/mail/user
- .mailrc
- .forward
- mbox
- .record

Registering with whois

- campus database of local email addresses
- voluntary
- only register from preferred system
- choose ***name***
- can register several
- one your full name (separated with period)

- **`Jason.Phillips`**

  `J.Phillips`

  `Phillips`

- check **whois** for duplicate

- **Name.Server@uwo.ca**
- **Subject: add**
- other operations: (**delete, query**)
- first line ***name***
- second line descriptive information

fname lname (dept,role,bldg,room,phone,ext)

- send
- informed by return mail
- 24 hours check **whois**
- reached with another address:

name**`@uwo.ca`**
`Phillips@uwo.ca`

EMACS

`emacs` ***filename***

```
Ctrl-x Ctrl-c
yes
```