Automated event notification

Tutor: Lưu Thanh Trà

Email: lttra@hoasen.edu.vn

Outline

- Notification with emails
- Scheduling utilities

mail and mailx

- mail -s “This is the subject” $MAILOUT_LIST < $MAIL_FILE

OR

- cat $MAIL_FILE | mail -s “This is the subject” $MAILOUT_LIST
- mailx -s “This is the subject” $MAILOUT_LIST < $MAIL_FILE

OR

- cat $MAIL_FILE | mailx -s “This is the subject” $MAILOUT_LIST

- $MAILOUT_LIST: list of receivers
- $MAIL_FILE : email content

Example with mail/mailx

```
MAIL_FILE=/tmp/mailfile.out
cat /dev/null > $MAIL_FILE
MAIL_LIST="randy@my.domain.com
    1234567890@mypage_somebody.net"

check_filesystems
# This function checks the filesystems percentage

if [ -s $MAIL_FILE ]
then
mail -s "Filesystem Full" $MAIL_LIST < $MAIL_FILE
fi
```

Note: -v for debug

```
check_filesystems ()
{
    FS=/var
    PERCENT=98
    THISHOST=$(uname -n)
    echo “$THISHOST: $FS is $PERCENT” | tee -
      a $MAIL_FILE
}
```

Note: ps -eo pid,pcpu,comm|egrep -v 0.0|sort -r -k2

sendmail

1. function send_notification
2. {
3. **if [-s $MAIL_FILE -a “$MAILOUT” = “TRUE”];**
4. then
5. **case $(uname) in**
6. AIX|HP-UX|Linux) SENDMAIL=”/usr/sbin/sendmail”
7. ;;
8. SunOS) SENDMAIL=”/usr/lib/sendmail”
9. ;;
10. esac
11. echo “\nSending email notification”
12. **$SENDMAIL -f randy@$THISHOST $MAIL_LIST < $MAIL_FILE**
13. fi
14. }

sendmail ⇔ postfix

- Sendmail, postfix, qmail can be changed alternatively
  - To display which MTA alternative is in use:
    - alternatives --display mta
  - To choose from the available MTA alternatives from the command line :
    - alternatives --config mta
  - To setup Postfix as the default mail system:
    - alternatives --set mta

Scheduling Utilities

- Crontab (= CRON Table)
  - run a job based on a schedule
  - job is executed on a periodic basis
- at
  - run a job some time in the future
- batch
  - run a job when system load is low

Periodic Execution: crontab

- crontab is based on control file
- crontab file has 6 columns:

```
minute  hour day  monthweekday
  command
```

- meaning:
  1. **minute** **0-59**
  2. **hour** **0-23**
  3. **day** **1-31**
  4. **month** **1-12**
  5. **weekday** **1-7 (1=Mon,2=Tue, … ,7=Sun)**
  6. **command** **Any UNIX command**
- "*" means any value

Example: crontab file

```
0   8  *  *  1  echo Happy Monday Morning
30 14  *  *  1  echo Meeting at 3pm
0  17  *  *  5  $HOME/bin/cleanup.sh
```

ISC (“Vixie”) Cron

- Replacement for standard cron daemon
- **`/etc/crontab`** – System crontab file
  - Inserted “run-as” field (6)
  - **`run-parts`**
- Scripts put into **`/etc/cron.<period>`**
  - **`cron.hourly`** (run 1 minute after every hour)
  - **`cron.daily`** (run 4:02 daily)
  - **`cron.weekly`** (run 4:22 every Sunday)
  - **`cron.monthy`** (run 4:42 first of every month)

crontab command

options:

-e	to edit the control file
-l	to list the control file
-r	to remove the control file

- for superuser

-u	to edit another user's control file

One Time Execution: at utility

- Use ‘at’ to run a command or list of commands at a later time
- Must specify on the command the time and date on which your command to be executed
- Do not have to be logged in when the commands are scheduled to run

Syntax:

```
% at timeDate command
```

at utility

- Can give as much of date as desired
- If date/time has passed, command will run instantly
  - In case system was down when it was supposed to run

Examples:

```
% at 13:45 Wed
% at 01:45 pm Sep 18
% at 09:25 Sep 18 2010
% at 09:25 Sep 18, 2010
% at 11:00 pm tomorrow
% at teatime                    # 4:00 pm
```

Example: at utility

- End input list with cntl-D = end-of-text
- Remember command runs in batch
  - No output on terminal
  - May get email
  - Or just look for side effects

```
% at 08:55 am tomorrow [Return]
at> enscript -Pcsl csci330.csh
at> <EOT>
commands will be executed using /bin/sh
job 1 at Thu Dec  4 08:55:00 2008
```

at utilities

- atq

  lists user's scheduled jobs
- atrm

  removes specified job from at queue

batch command

- batch

  schedules job to be performed when system load is *low*
  same program as “at”

Syntax:

```
% batch command
```

on turing: *low* means load is less than 1.5

Login Accounting

- **`wtmp`** – DB of all logins and logouts
  - Time
  - User/TTY
  - Where
- **`utmp`** – DB of currently logged in users
- Reports
  - **`who/w`** – Lists currently logged in users
  - **`last`** – Lists all login sessions
  - **`lastlog`** – List last time users logged in