

Assignment I

ให้เขียนรายงาน อย่างละเอียด และดีที่สุด (ให้ผู้อ่านเข้าใจง่ายที่สุด)

1. fc Command

คำอธิบาย : คำสั่ง fc เป็นคำสั่งที่ใช้ในการแสดงรายการที่ต้องการแก้ไขหรือต้องการดำเนินการกับคำสั่งก่อนหน้าอีกครั้ง กล่าวคือหากต้องการใช้คำสั่งก่อนหน้าเดิมอีกครั้ง ไม่จำเป็นต้องพิมพ์ใหม่ สามารถใช้คำสั่ง fc เพื่อเพิ่มคำสั่งก่อนหน้าได้ทันที เหมาะกับการเขียนคำสั่งยาวๆ และซ้ำเดิมโดยไม่ต้องเขียนใหม่ตั้งแต่ต้น

Syntax : [-e name] [-lnr] [first] [last]

คำอธิบายแต่ละ Argument

[-e name] คือ ชื่อของ editor ที่ต้องการใช้สำหรับแก้ไข

[-lnr] คือ ตัวเลือกที่แยกออกมาเพื่อใช้กับคำสั่ง fc

[first] และ *[last]* คือช่วงที่เราจะระบุค่าสำหรับนำคำสั่งเดิมมาใช้

ตัวอย่าง เช่น เราต้องการเช็คค่าที่ผ่านมาเราเขียนคำสั่งอะไรไปบ้างในโปรแกรม PuTTY ให้เราใช้คำสั่ง

fc -l ซึ่ง -l ย่อมาจาก list

```
debian@beaglebone:~$ fc -l
1      sudo passwd root
2      exit
3      clear
4      sudo passwd root
5      whoami
6      logout
7      sudo su
8      shutdown -r now
9      sudo su
10     help
11     help fc
12     help coproc
13     sudo su
14     sudo apt updtae && sudo apt upgrade
15     sudo apt update && sudo apt upgrade
debian@beaglebone:~$
```

หลังจากนั้น เราจะเห็นว่า ในบรรทัดที่ 14 มีคำสั่งที่เขียนผิด นั่นคือ upatae แต่ถ้าเราไม่มีเวลาพอเขียนใหม่ทั้งหมด ให้เราใช้คำสั่งต่อไปนี้

Fc 14

```

debian@beaglebone: ~
15      sudo apt update && sudo apt upgrade
debian@beaglebone:~$ fc 14
GNU nano 3.2 /tmp/bash-fc.Db8EYe
sudo apt updtae && sudo apt upgrade

```

ให้เราแก้จุดที่เราต้องการแก้

```

debian@beaglebone: ~
GNU nano 3.2 /tmp/bash-fc.Db8EYe
sudo apt update && sudo apt upgrade

```

จากนั้นให้เรากดปุ่ม F2 (หากสงสัยคำสั่งอะไรให้กด F1 ระบบจะมีคำอธิบายให้)

```

Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel

```

ระบบจะถามเราว่า ต้องการบันทึกที่เราแก้ไขใหม่นี้หรือไม่ ถ้าใช่ พิมพ์ Y แล้ว Enter

```

debian@beaglebone: ~
Err:3 http://deb.debian.org/debian buster-updates InRelease
  Temporary failure resolving 'deb.debian.org'
Err:4 http://deb.debian.org/debian-security buster/updates InRelease
  Temporary failure resolving 'deb.debian.org'
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
W: Failed to fetch http://deb.debian.org/debian/dists/buster/InRelease Temporary failure resolving 'deb.debian.org'
W: Failed to fetch http://deb.debian.org/debian/dists/buster-updates/InRelease Temporary failure resolving 'deb.debian.org'
W: Failed to fetch http://deb.debian.org/debian-security/dists/buster/updates/InRelease Temporary failure resolving 'deb.debian.org'
W: Failed to fetch http://repos.rcn-ee.com/debian/dists/buster/InRelease Temporary failure resolving 'repos.rcn-ee.com'
W: Some index files failed to download. They have been ignored, or old ones used instead.
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
debian@beaglebone:~$

```

จากนั้นให้เราทำการตรวจสอบว่า คำสั่งล่าสุดเราได้รับการแก้ไขแล้วหรือไม่

```

debian@beaglebone:~$ fc -l
3      clear
4      sudo passwd root
5      whoami
6      logout
7      sudo su
8      shutdown -r now
9      sudo su
10     help
11     help fc
12     help coproc
13     sudo su
14     sudo apt updtae && sudo apt upgrade
15     sudo apt update && sudo apt upgrade
16     fc -l
17     sudo apt updtae && sudo apt upgrade
18     sudo apt update && sudo apt upgrade
debian@beaglebone:~$

```

หรือหากเราต้องการคัดลอกข้อความจากบรรทัดที่ 14 จนถึง 18 ก็สามารถทำได้เช่นกัน โดยใช้คำสั่ง

fc 14 18 และถ้าไม่ต้องการบรรทัดที่ 16 สามารถลบได้ ทำการบันทึก จะได้ดังนี้

```

instead.
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
debian@beaglebone:~$ fc -l
8      shutdown -r now
9      sudo su
10     help
11     help fc
12     help coproc
13     sudo su
14     sudo apt updtae && sudo apt upgrade
15     sudo apt update && sudo apt upgrade
16     fc -l
17     sudo apt updtae && sudo apt upgrade
18     sudo apt update && sudo apt upgrade
19     fc -l
20     sudo apt updtae && sudo apt upgrade
21     sudo apt update && sudo apt upgrade
22     sudo apt updtae && sudo apt upgrade
23     sudo apt update && sudo apt upgrade
debian@beaglebone:~$

```

เราจะเห็นว่า ในบรรทัดที่ 20 – 23 เป็นการทำซ้ำจากบรรทัดที่ 14-18 (โดยไม่นับบรรทัดที่ 16) เหมาะกับการเขียนคำสั่งซ้ำๆนั่นเอง

2. set command

คำอธิบาย : เป็นคำสั่งที่ไว้สำหรับการตั้งค่า หรือกำหนดสิ่งต่างใน system

Syntax : set [options]

ตัวอย่างของ Options

- 1) -a จะทำเครื่องหมายตัวแปรที่แก้ไขหรือสร้างขึ้นเพื่อส่งออก
- 2) -b จะแจ้งเตือน Users ทันทีเมื่อการทำงานพื้นหลัง (งานที่ทำอัตโนมัติโดยที่เราไม่ต้องสั่ง) ถูกยกเลิก
- 3) -e จะสั่งให้ออกทันที หากพบคำสั่งล้มเหลว
- 4) -f ปิดการใช้งานการสร้างชื่อไฟล์
- 5) -h เปิดการใช้งานการจดจำคำสั่งหรือฟังก์ชันเมื่อมีการกำหนดฟังก์ชัน
- 6) -n เปิดการอ่านเฉพาะคำสั่ง แต่ไม่สามารถดำเนินการใดๆได้
- 7) -t ออกหลังจากอ่านและดำเนินการคำสั่งเดียว
- 8) -u ใช้เมื่อเจอตัวแปรที่ไม่ได้ตั้งค่าหรือถูกกำหนด ยกเว้นพารามิเตอร์พิเศษ เช่น * หรือ @
- 9) -v พิมพ์บรรทัดของอินพุตเซลล์ขณะที่ระบบกำลังประมวลผลอยู่
- 10) -x พิมพ์ Argument และ Command ระหว่างการดำเนินการ
- 11) - ล้างสิ่งที่ตั้งค่าทั้งหมด

ตัวอย่างของการใช้งานคำสั่ง

```
debian@beaglebone:~$ set red blue green black
debian@beaglebone:~$ echo $4
black
debian@beaglebone:~$ set --
debian@beaglebone:~$ echo $4
debian@beaglebone:~$
```

ในตอนแรกเราทำการ set คำไว้ 4 คำ ได้แก่ red blue green และ black และใช้คำสั่ง echo \$_ ตามด้วยหมายเลข แต่เมื่อเราใช้คำสั่ง set - และทำซ้ำคำสั่งก่อนหน้าอีกครั้งจะพบว่า คำที่ set ไว้ก่อนหน้านี้ได้หายไปหมดแล้ว เพราะถูกยกเลิกด้วยคำสั่ง set -

หรือการใช้คำสั่ง `set -u` เพื่อต้องการให้แจ้งเตือนเมื่อพบตัวแปรที่ไม่ได้ถูกกำหนดหรือตั้งค่าไว้ก่อนหน้า ดังที่แสดง จากตอนแรกที่เราพิมพ์ `echo $foo` ได้ เมื่อเราใช้คำสั่ง `set -u` ทำให้ระบบจะแจ้งเราว่ามีตัวแปรที่ไม่ทราบ

```
debian@beaglebone:~$ echo $foo
debian@beaglebone:~$ set -u
debian@beaglebone:~$ echo $foo
-bash: foo: unbound variable
debian@beaglebone:~$
```

3. umask command

คำอธิบาย : คำสั่ง `umask` เป็นคำสั่งที่ใช้สำหรับระบุค่าสิทธิ์ในการใช้งานไฟล์ที่สร้างขึ้นในระบบ Linux สามารถตรวจสอบค่า `umask` ได้ โดยใช้คำสั่ง `umask` โดยที่ default คือ 0022 โดยที่ไฟล์ข้อมูลจะมีค่า default เริ่มต้นที่ 0666 แต่ไฟล์ที่ระบบสามารถประมวลผลได้คือ 0777 วิธีแก้คือ

```
debian@beaglebone:~$ umask
0022
debian@beaglebone:~$
```

Syntax : `umask [-p] [-S] [mode]`

เราจะทราบได้อย่างไรว่าเรามีสิทธิ์ในการแก้ไขไฟล์ได้หรือไม่ ทำได้โดย

ถ้าเรามีไฟล์ 0066 ให้นำมา XOR กับ 0022 หรือค่า default ของระบบ ซึ่งจะได้ค่า 0644 เท่านั้น ตามที่ได้ระบุไว้ว่า หากเราต้องการให้ระบบประมวลผลได้ ไฟล์ต้องมี `umask 0777` ดังนั้นเจ้าของไฟล์จะเป็นผู้เดียวที่อ่านและแก้ไขไฟล์ได้ ในขณะที่ผู้ใช้งานคนอื่นๆจะทำได้

แต่ถ้าเรามีไฟล์ 0077 เมื่อนำมา XOR กับ 0022 จะได้ 0755 เจ้าของสามารถ `cd` ลงในไฟล์และแสดงรายการอ่าน แก้ไข สร้าง หรือลบไฟล์ใน Directories ผู้ใช้งานกลุ่มอื่นสามารถ `cd` และแสดงรายการและอ่านไฟล์ได้

เมื่อเราใช้คำสั่ง `umask 0777` จะเปลี่ยนค่าที่ถูก Default ไว้จาก 0022 เป็น 0777 เมื่อมีการแทรกไฟล์เข้ามาในระบบจะเป็นการชั่วคราวได้ว่าไฟล์ที่เพิ่มเข้ามานั้นสามารถประมวลผลได้นั่นเอง

```
debian@beaglebone:~$ umask 0777
debian@beaglebone:~$ umask
0777
debian@beaglebone:~$
```

4. bc command

คำอธิบาย : เป็นคำสั่งที่ใช้ในการคำนวณตัวเลข ด้วยสูตรทางคณิตศาสตร์

Syntax : bc [-hlwsqv] [long-options] [file...]

คำอธิบาย Options บางส่วน

- 1) -h , --help แสดงการใช้งานและจบการทำงาน
- 2) -l , --interactive เปิดใช้งานโหมดตอบโต้
- 3) -l , --matlib ค้นหา math library

ตัวอย่างการใช้งาน

เราสามารถพิมพ์คำสั่ง bc อย่างเดียวเพื่อคำนวณเลขอย่างง่ายได้ ดังที่แสดง

```
debian@beaglebone:~$ bc
bc 1.07.1
Copyright 1991-1994, 1997, 1998, 2000, 2004, 2006, 2008, 2012-2017 Free Software
Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
█
```

ซึ่งสามารถทำการบวก ลบ คูณ และหาร ได้ แต่การหารนั้นจะไม่สามารถได้ผลลัพธ์ที่เป็นทศนิยมได้ เช่น $\frac{1}{2}$ ควรจะได้ 0.5 แต่ผลลัพธ์ที่แสดงเป็น 0 หรือ $\frac{3}{2}$ ควรจะได้ 1.5 แต่กลับได้เพียง 1 เท่านั้น

```
1+1
2
1-1
0
1*1
1
1/1
1
1+2
3
1-2
-1
1*2
2
1/2
0
3/2
1
█
```

โดยคำสั่ง bc สามารถกำหนดทศนิยมผ่าน scale ได้ ดังนี้

```
debian@beaglebone:~$ echo 'scale=3; (3/2)'|bc
1.500
debian@beaglebone:~$
```

จากภาพจะเป็นว่า 3 / 2 ผลลัพธ์ที่ได้เป็น 1.500 ซึ่งตรงกับค่าจริงที่ควรจะได้แล้วนั่นเอง

5. id command

คำอธิบาย : คำสั่ง id ใช้เพื่อค้นหาชื่อ Users และ ชื่อ Group รวมทั้ง ID หรือ UID ของผู้ใช้งานในปัจจุบันหรือผู้ใช้งานรายอื่นในเซิร์ฟเวอร์ คำสั่งนี้มีประโยชน์อย่างมากในการค้นหาชื่อผู้ใช้งานจริง ค้นหา UID ผู้ใช้งานเฉพาะ และแสดงการรักษาความปลอดภัยของผู้ใช้งานปัจจุบัน

Syntax : id [Options] [User]

คำอธิบาย Options

- 1) -g แสดงเฉพาะ effective group id
- 2) -G แสดงทุกกลุ่ม id
- 3) -n แสดงชื่อแทนตัวเลข
- 4) -r แสดง ID จริงแทนตัวเลข
- 5) -u แสดงเฉพาะ effective User id
- 6) -help แสดงข้อความช่วยเหลือและออก
- 7) -version แสดงข้อมูลเวอร์ชันและออก

ตัวอย่าง

ถ้าเราพิมพ์คำสั่ง `id` เพียงอย่างเดียว จะปรากฏชื่อและ UID ของ User อื่นๆให้เราเห็น

```
debian@beaglebone:~$ id
uid=1000(debian) gid=1000(debian) groups=1000(debian),4(adm),15(kmem),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),100(users),101(systemd-journal),110(bluetooth),112(netdev),114(i2c),987(cloud9ide),988(xenomai),989(weston-launch),990(tiscd),991(docker),992(iio),993(spi),994(admin),996(remote-proc),997(egep),998(pwm),999(gpio)
debian@beaglebone:~$
```

ในส่วนของการคำสั่ง `id -G` จะแสดง ID ของทุกกลุ่ม แต่คำสั่ง `id -g` จะแสดงแค่กลุ่มเดียวที่มีประสิทธิภาพสูงสุด

```
debian@beaglebone:~$ id -G
1000 4 15 20 24 25 29 30 44 46 100 101 110 112 114 987 988 989 990 991 992 993 994 996 997 998 999
debian@beaglebone:~$ id -g
1000
debian@beaglebone:~$
```

6. who command

คำอธิบาย : คำสั่ง `who` ใช้แสดงข้อมูลของ user ที่ Login ในขณะนั้น

Syntax : `who [Options] [File | Arg1 Arg2]`

คำอธิบาย Options

- 1) -b ระบุเวลาครั้งสุดท้ายที่ระบบทำการ boot
- 2) -d แสดงกระบวนการที่ไม่สำเร็จ
- 3) -H แสดงเส้นของ Column headings
- 4) -t แสดงนาฬิกาครั้งสุดท้ายของระบบที่เปลี่ยนแปลง
- 5) -T เพิ่มสถานะของ User ว่าเป็น + , - หรือ ?

ตัวอย่าง

คำสั่ง `who`

```
debian@beaglebone:~$ who
debian pts/0 2020-04-06 19:22 (192.168.7.1)
debian@beaglebone:~$
```

User : Debian เข้าครั้งล่าสุด 19.22 นาฬิกา วันที่ 6 เดือน เมษายน ปี 2020 ไปยัง IP 192.168.7.1

คำสั่ง `who -T -H`

```
debian@beaglebone:~$ who -T -H
NAME      LINE      TIME      COMMENT
debian    + pts/0    2020-04-06 19:22 (192.168.7.1)
debian@beaglebone:~$
```

User : Debian สถานะ + เข้าครั้งล่าสุด 19.22 นาฬิกา วันที่ 6 เดือน เมษายน ปี 2020 ไปยัง IP 192.168.7.1