



# **Basic Linux3**

**Ubuntu**

---

# การ Login Ubuntu

เครื่องจะขึ้น login prompt เพื่อเตรียมพร้อมรับการทำงาน  
เมื่อท่านต้องการเข้าใช้งานเครื่อง ต้องใส่ชื่อ login และ  
password

ป้อน user ==> **test** แล้วกดแป้น Enter

ป้อน password ==> **123456789** แล้วกดแป้น Enter

# คำสั่ง touch (change file timestamps)

เป็นคำสั่งที่มีไว้เพื่อสร้างแฟ้มข้อมูลเปล่าขึ้นมา

รูปแบบคำสั่ง

touch [ชื่อแฟ้มข้อมูล]

ตัวอย่างการใช้งาน

```
$ touch /tmp/whoistheowner
```

# คำสั่ง touch

Work shop สร้างแฟ้มเปล่า

```
$ mkdir lab3
```

```
$ touch lab3/EX3_1
```

```
$ touch lab3/EX3_2 lab3/EX3_3
```

```
$ ls -l lab3
```

# Work shop

```
sukhothai@ubuntu:~$ mkdir lab3
sukhothai@ubuntu:~$ touch lab3/EX3_1
sukhothai@ubuntu:~$ touch lab3/EX3_2 lab3/EX3_3
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:51 EX3_1
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:51 EX3_2
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:51 EX3_3
sukhothai@ubuntu:~$ _
```

# File Permission

ในระบบปฏิบัติการ Linux ไฟล์ทุกไฟล์ และไดเรกทอรีทุกไดเรกทอรี จะมีเจ้าของ และมีการกำหนดสิทธิในการจัดการ สามารถตรวจสอบ permission ของไฟล์ได้ด้วยคำสั่ง

```
$ ls -l
```

ผลลัพธ์จากคำสั่ง ได้ออกมาจะเป็นประมาณนี้

```
rw-r--r-- 4 test user 4096 Jan 31 2013 EX3_1
```

# File Permission

```
-rwxr-xr-x 2 test user 4096 Oct 20 10:50 start
```

File Permission

ชื่อแฟ้มข้อมูล

วันที่สร้างแฟ้มข้อมูล

ขนาดของแฟ้มข้อมูล

กลุ่มของแฟ้มข้อมูล

เจ้าของแฟ้มข้อมูล

จำนวน link หรือแฟ้มข้อมูลที่อยู่ใน directory

# File Permission

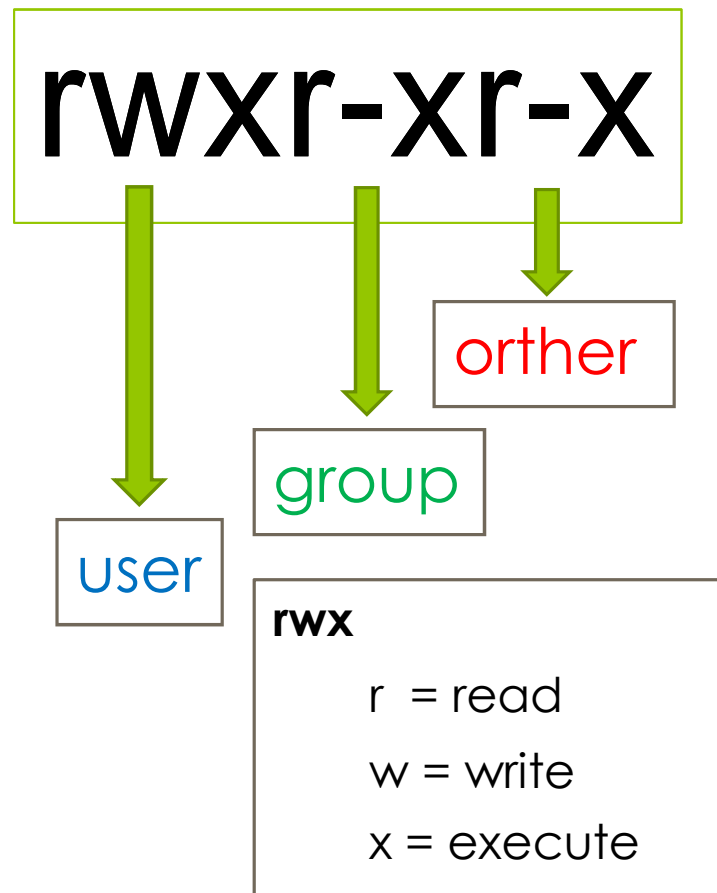
`-rwxr-xr-x`

- เพิ่มข้อมูลปกติ
- d เพิ่มข้อมูลประเภท directory
- l เพิ่มข้อมูลประเภท soft link
- b block device (HDD, CD-ROM)
- c character device (modem)
- s socket เชื่อมต่อแบบ network
- p เพิ่มข้อมูลประเภท pipe



# File Permission

สิทธิการเข้าถึงแฟ้มข้อมูล  
แบ่งออกเป็น 3 กลุ่ม คือ  
เจ้าของ (user : u) ผู้ใช้อื่นใน  
กลุ่ม (group : g) และคน  
อื่นๆ (other : o) โดยการ  
เข้าถึงแฟ้มข้อมูลมี 3 แบบคือ  
การอ่าน (read : r) การเขียน  
(write : w) และการเรียกใช้  
งาน (execute : x)



# File Permission

ตัวอย่าง กำหนดสิทธิ์การเข้าถึง แยกพิจารณาเป็นกลุ่มย่อย

--- : ไม่มีสิทธิ์อะไรเลย (เลขที่ใช้คือ 0)

--X : ประมวลผลได้อย่างเดียว (เลขที่ใช้คือ 1)

R-- : อ่านได้อย่างเดียว (เลขที่ใช้คือ 4)

RW- : อ่าน และเขียนได้ (เลขที่ใช้คือ 6)

R-X : อ่าน และประมวลผลได้ (เลขที่ใช้คือ 5)

RWX : อ่าน เขียน และประมวลผลได้ (เลขที่ใช้คือ 7)

# File Permission

ตัวอย่าง กำหนดสิทธิการเข้าถึงทั้ง 3 กลุ่ม

RWX----- : เจ้าของเท่านั้นที่มีสิทธิทุกอย่าง (เลขที่ใช้คือ 700)

RWXRWX--- : เจ้าของ และสมาชิกกลุ่มเดียวกันมีสิทธิทุกอย่าง (เลขที่ใช้คือ 770)

RWXR-XR-X : เจ้าของทำได้หมด ส่วนกลุ่มและคนอื่นอ่านและ  
ประมวลผลได้ (เลขที่ใช้คือ 755)

R--R--R-- : ทุกคนอ่านได้อย่างเดียว (เลขที่ใช้คือ 444)

# File Permission

การปรับสิทธิการเข้าถึงแฟ้มข้อมูล สามารถกำหนดเป็นตัวเลขก็ได้

ตัวอย่าง

**เลขฐานสิบ**

ถ้าต้องการปรับสิทธิการเข้าถึงแฟ้มข้อมูล myfile.txt ให้

ผู้ใช้ มีสิทธิทุกอย่างคือ อ่าน เขียน และเรียกใช้งาน (4+2+1)

กลุ่ม มีสิทธิในการการอ่าน (4+0+0)

คนอื่นๆ ไม่มีสิทธิใดๆ ในแฟ้มข้อมูลนี้เลย (0+0+0)

`chmod 740 myfile.txt`

**rwX**  
↓ ↓ ↓  
4 2 1

# File Permission

การปรับสิทธิการเข้าถึงแฟ้มข้อมูล สามารถกำหนดเป็นตัวเลขก็ได้

ตัวอย่าง

**rwxr-xr-x**



**111101101**

**เลขฐานสอง**

รหัสเลขฐาน 111 101 101 = 755 คือเจ้าของแฟ้มใช้ได้ครบ

Group เดียวกัน และคนอื่นๆ อ่านและเรียกใช้งานได้

โดยปกติเมื่อมีการสร้างไฟล์ขึ้นมาเจ้าของไฟล์ก็จะเป็นชื่อ user

และ group ก็จะเป็น group ที่ user นั้นอยู่

# คำสั่ง chmod (change file mode bits)

ใช้สำหรับแก้ไขสิทธิการเข้าถึงแฟ้มข้อมูล สำหรับเจ้าของ (Owner), คนในกลุ่ม (Group) และคนอื่น (Other) ให้สามารถทำอะไรกับแฟ้มข้อมูลได้บ้าง

รูปแบบคำสั่ง

```
$ chmod permission ชื่อไฟล์หรือไดเรกทอรี
```

# คำสั่ง chmod

ตัวอย่างการใช้คำสั่ง

สั่งให้แฟ้มมี Permission เป็น rwxr-xr-x ใช้คำสั่ง

```
$ chmod 755 first.txt
```

เจ้าของแฟ้มสามารถ อ่าน เขียน และรันได้ ก็จะได้  $4(\text{read}) + 2(\text{write}) + 1(\text{execute}) = 7$  ส่วนคนใน group เดียวกันกับ other รันและอ่านได้ก็จะเป็น  $4(\text{read}) + 1(\text{execute}) = 5$

# คำสั่ง chmod

ตัวอย่างการใช้คำสั่ง

หากกำหนด permission เขียนและอ่านให้กับเจ้าของแฟ้ม  
ส่วนคนอื่นๆ ให้อ่านได้อย่างเดียว ใช้คำสั่ง

```
$ chmod 644 lab1
```

หากใช้หลักการเลขฐานสิบ 6 ได้มาจาก 4(read) + 2(write)  
ส่วน group และ other อ่านได้อย่างเดียว เป็น 4



# คำสั่ง chmod

ตัวอย่างการใช้คำสั่ง

```
$ chmod a+x filename
```

เป็นการกำหนดความสามารถ execute ให้กับทุกๆ คน ไม่ว่าจะ  
จะเป็น user group และ other

```
$ chmod o-wx filename
```

ปลดความสามารถในการอ่าน write และ execute ออกจาก  
other

# คำสั่ง chmod

ตัวอย่างการใช้คำสั่ง

```
$ chmod ug+w filename
```

เป็นการเพิ่ม write ให้กับ user และ group

```
$ chmod 755 -R directoryname
```

เมื่อต้องการเปลี่ยน permission ให้กับทุกไฟล์ในไดเรกทอรี

ใส่ -R เข้าไปหมายถึง recursive

# คำสั่ง chmod

ตัวอย่างการใช้คำสั่ง

\$ **chmod 777 index.php** :: ทำให้แฟ้มนี้ อ่าน เขียน และ  
ประมวลผล โดยทั้ง 3 กลุ่ม

\$ **chmod 755 \* -Rf** :: ทำให้ทุกแฟ้ม ทุก directory ในห้อง  
ปัจจุบัน เปลี่ยนตามที่กำหนด

# คำสั่ง chmod

Work shop เปลี่ยน permission ของแฟ้ม

```
$ ls -l lab3
```

```
$ chmod 744 lab3/EX3_1
```

```
$ chmod a-r+w lab3/EX3_2
```

```
$ chmod ugo+x lab3/EX3_3
```

```
$ ls -l lab3
```

# Work shop

```
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_2
-rw-r--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ chmod 744 lab3/EX3_1
sukhothai@ubuntu:~$ chmod a-r+w lab3/EX3_2
sukhothai@ubuntu:~$ chmod ugo+x lab3/EX3_3
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rwxr--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
--w--w--w- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_2
-rwxr-xr-x 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ _
```

# คำสั่ง chown (change file owner and group)

ใช้ในการกำหนดความเป็นเจ้าของแฟ้มข้อมูล

รูปแบบคำสั่ง

```
$ chown newuser filename
```

ใช้เปลี่ยน group ของแฟ้มข้อมูล

รูปแบบคำสั่ง

```
$ chown newuser:newgroup filename
```

# คำสั่ง chown

ตัวอย่างการเปลี่ยน Owner ของแฟ้มข้อมูล

```
$ chown test file1.txt
```

ตัวอย่างการเปลี่ยนทั้ง Owner และ Group ในคราวเดียว

```
$ chown root.root file1.txt
```

```
$ chown test:users x :: เปลี่ยนเจ้าของของแฟ้ม x ให้เป็น  
test และอยู่ในกลุ่มของ users
```

# คำสั่ง chown

ตัวอย่างการใช้งาน

\$ **chown user1 filename** คือเปลี่ยนเจ้าของไฟล์ชื่อ  
filename เป็น user1

\$ **chown -R user1.root dirname** คือเปลี่ยนทั้งเจ้าของไฟล์  
และกลุ่มไปพร้อมกันทุกไฟล์ใน sub dirname



# คำสั่ง chown

ตัวอย่างการใช้งาน

\$ **chown user1 somedir** คือเปลี่ยนเจ้าของไดเรคทอรี somedir เป็น user1

# คำสั่ง chown

Work shop เปลี่ยนเจ้าของแฟ้มข้อมูล

```
$ ls -l lab3
```

```
$ sudo chown root:root lab3/EX3_2
```

```
$ ls -l lab3
```

# Work shop

```
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rwxr--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
--w--w--w- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_2
-rwxr-xr-x 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ sudo chown root:root lab3/EX3_2
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rwxr--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
--w--w--w- 1 root      root      0 2013-07-21 03:58 EX3_2
-rwxr-xr-x 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ _
```

# คำสั่ง chgrp (change group ownership)

ใช้เปลี่ยน group ของแฟ้มข้อมูล

รูปแบบคำสั่ง

```
$ chgrp newgroup filename
```

# คำสั่ง chgrp

ตัวอย่างการเปลี่ยน Group ของไฟล์

```
$ chgrp test file1.txt
```

```
$ chgrp users y :: เปลี่ยนกลุ่มของแฟ้ม x ให้เป็น users
```

```
$ chgrp root /root/* :: เปลี่ยน group ให้กับไฟล์ทุกไฟล์ใน  
ไดเรกทอรี /root ให้เป็น group root
```

# คำสั่ง chgrp

Work shop เปลี่ยน group ของแฟ้ม

```
$ ls -l lab3
```

```
$ sudo chgrp root lab3/EX3_3
```

```
$ ls -l lab3
```

# Work shop

```
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rwxr--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
--w--w--w- 1 root      root      0 2013-07-21 03:58 EX3_2
-rwxr-xr-x 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ sudo chgrp root lab3/EX3_3
sukhothai@ubuntu:~$ ls -l lab3
total 0
-rwxr--r-- 1 sukhothai sukhothai 0 2013-07-21 03:58 EX3_1
--w--w--w- 1 root      root      0 2013-07-21 03:58 EX3_2
-rwxr-xr-x 1 sukhothai root      0 2013-07-21 03:58 EX3_3
sukhothai@ubuntu:~$ _
```

# คำสั่ง man

(an interface to the on-line reference manuals) เป็นคำสั่งที่ช่วยอธิบายหน้าที่ของคำสั่ง พร้อมกับแสดง parameter ในการใช้งานทั้งหมดของคำสั่งนั้น และยังมีตัวอย่างการใช้ และคำสั่งที่เกี่ยวข้อง

รูปแบบคำสั่ง

```
$ man [command]
```

หมายเหตุ เมื่อต้องการออก กดแป้น q; กดแป้น [Spacebar] เพื่อเลื่อนหน้าถัดไป; ใช้ลูกศรขึ้นดูหน้าที่ผ่านมา



# คำสั่ง man

ตัวอย่างคำสั่ง และการใช้งาน

\$ **man man** :: เพื่ออธิบายคำสั่ง man เอง ว่าตัวคำสั่งนี้ใช้  
อย่างไร

\$ **man ls** :: เพื่ออธิบายคำสั่ง ls ว่าใช้อย่างไร

\$ **man useradd** :: เพื่ออธิบายคำสั่ง useradd ว่าใช้อย่างไร

# คำสั่ง man

หากทราบแค่ keyword ยังไม่ทราบกระทั่งชื่อคำสั่ง เช่น  
อยากรู้คำสั่งที่ใช้ในการทำการ copy ให้พิมพ์ว่า man -k  
copy จะแสดงรายชื่อของคำสั่งที่เกี่ยวข้องกับการ copy  
หากทราบคำสั่ง แต่ไม่ทราบรูปแบบการใช้งานให้ลองพิมพ์  
คำสั่ง แล้วตามด้วย -- help ก็จะมีตัวช่วยอย่างย่อขึ้นมาให้  
เช่น # ls --help

# คำสั่ง whereis

(locate the binary, source, and manual page files for a command) เป็นคำสั่งค้นหาแฟ้มที่ต้องการว่าอยู่ที่ส่วนใด แต่ค้นได้เฉพาะที่กำหนดไว้ใน PATH เท่านั้น หากต้องการค้นทั้งเครื่องต้องใช้คำสั่ง find

รูปแบบคำสั่ง

```
$ whereis [command]
```

# คำสั่ง whereis

ตัวอย่างคำสั่ง และการใช้งาน

\$ **whereis usermod** :: แสดงห้องที่เก็บคำสั่ง usermod ทำให้สามารถสั่ง run จากห้องที่เก็บคำสั่งโดยตรงได้

\$ **whereis ifconfig** :: แสดงห้องที่เก็บคำสั่ง ifconfig ทำให้ใช้คำสั่งเช่น /sbin/ifconfig ได้โดยตรง

# คำสั่ง whereis

Work shop ค้นหาแฟ้มที่กำหนดไว้ใน PATH

```
$ whereis cp
```

```
$ whereis date
```

```
$ whereis mkdir
```

# Work shop

```
sukhothai@ubuntu:~$ whereis cp
cp: /bin/cp /usr/share/man/man1/cp.1.gz
sukhothai@ubuntu:~$ whereis date
date: /bin/date /usr/share/man/man1/date.1.gz
sukhothai@ubuntu:~$ whereis mkdir
mkdir: /bin/mkdir /usr/share/man/man1/mkdir.1.gz
sukhothai@ubuntu:~$ _
```

# คำสั่ง find

(search for files in a directory hierarchy) เป็นคำสั่งที่ใช้ค้นหาไฟล์ที่ต้องการว่าอยู่ใน directory ของเครื่องเราหรือไม่

รูปแบบคำสั่ง

```
$ find [path] [parameter] [filename]
```

# คำสั่ง find

ตัวอย่างคำสั่ง และการใช้งาน

\$ **find / -name hello.pl** :: ใช้ค้นหาแฟ้ม hello.pl ในทุก directory

\$ **find / -name hello\*** :: ใช้ค้นหาแฟ้มที่ขึ้นต้นด้วยคำว่า hello

\$ **find /bin -size 626188c** :: ใช้ค้นหาแฟ้มที่มีขนาด 626188 ถ้าเป็น RH8 จะพบแฟ้ม bash



# คำสั่ง find

Work shop ค้นหาเพิ่มข้อมูล

```
$ find ~ -name EX3*
```

```
$ find ~/lab* -name lab*
```

# Work shop

```
sukhothai@ubuntu:~$ find ~ -name EX3*  
/home/sukhothai/lab3/EX3_2  
/home/sukhothai/lab3/EX3_3  
/home/sukhothai/lab3/EX3_1  
sukhothai@ubuntu:~$ find ~/lab* -name lab*  
/home/sukhothai/lab3  
sukhothai@ubuntu:~$ _
```

# คำสั่ง clear (clear the terminal screen)

เป็นคำสั่งให้มีการลบข้อความบนจอภาพ

รูปแบบคำสั่ง

```
$ clear
```

# คำสั่ง echo (display a line of text)

เป็นคำสั่งให้แสดงข้อความออกทางจอภาพ

รูปแบบคำสั่ง

```
$ echo (ข้อความที่ต้องการให้แสดงผล)
```

ตัวอย่าง

```
$ echo my name is user1
```

```
$ echo Hello > /dev/tty2 ส่งข้อความ Hello ไปออกจอ  
เทอร์มินอลที่ 2
```

# คำสั่ง echo

Work shop แสดงข้อความออกทางจอภาพ

```
$ echo my name is Sukhothai
```

```
$ echo good day > /dev/tty2
```

# Work shop

```
sukhothai@ubuntu:~$ echo My name is Sukhothai
My name is Sukhothai
sukhothai@ubuntu:~$ echo good DAY > /dev/tty1
good DAY
sukhothai@ubuntu:~$ _
```

## การทดลองที่ 3

1. ให้สร้างแฟ้มข้อมูลว่าง lab3\_1 และ lab3\_2 ในไดเรกทอรี lab3
2. ให้ปรับสิทธิแฟ้มข้อมูล lab3\_1 ให้ผู้ใช้มีสิทธิทุกอย่าง กลุ่มและคนอื่นๆ สามารถเรียกใช้งานได้อย่างเดียว
3. ให้ปรับสิทธิแฟ้มข้อมูล lab3\_2 ให้ผู้ใช้ มีสิทธิทุกอย่าง กลุ่ม มีสิทธิทุกอย่าง คนอื่นๆ สามารถอ่านและเรียกใช้งานได้
4. เปลี่ยนเจ้าของแฟ้มข้อมูลและกลุ่มของแฟ้ม lab3\_1 เป็น root
5. เปลี่ยน group ของแฟ้ม lab3\_2 เป็น root
6. ให้ค้นหาแฟ้มข้อมูลที่ชื่อขึ้นต้นด้วย lab3\_
7. เขียนสรุปคำสั่งในแต่ละขั้นตอนของแต่ละข้อ ลงบนกระดาษ