

Table of Contents

Basic Command Line Interface (CLI)	2
CLI Tools	2
Windows Tools	2
Linux Tools	3
MacOS Tools	4
Format CLI	5
CLI	5
man Command	5
System Category	6
>> uname	6
>> hostname	6
>> whoami	6
>> date	6
File Category	7
>> ls	7
>> pwd	7
>> mkdir	7
>> rm	8
>> cp	8
>> mv	9
>> cat / head / tail	9
>> tar	10
>> gzip	11
File Permission Category	12
>> chmod	12
Network Category	13
>> ping	13
>> wget	13
>> netstat	13
Search Category	14
>> grep	14
>> find	14
Remote Category	15
File transfer Category	15

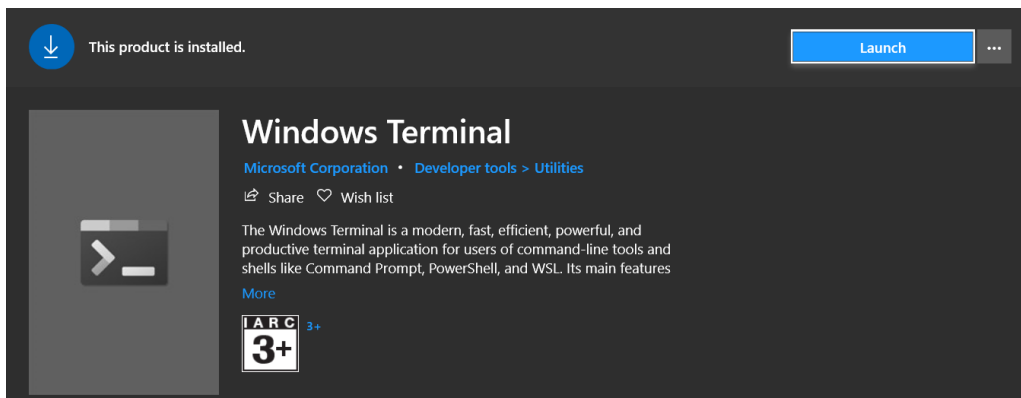
Basic Command Line Interface (CLI)

Command-line user interface คือ รูปแบบการแสดงผลแบบ Text Mode โดยหน้าโปรแกรมจะรับ Input เข้าไปที่ละบรรทัด ๆ จากนั้นจะหาความหมายของคำสั่งนั้น ๆ เพื่อนำไปสั่งให้ระบบปฏิบัติการ (OS) ทำตามคำสั่งนั้น ๆ ต่อไป โดยสมัยที่ยังเป็น MS-DOS จะเรียกอีกอย่างหนึ่งว่า GUI (Graphical User Interface)

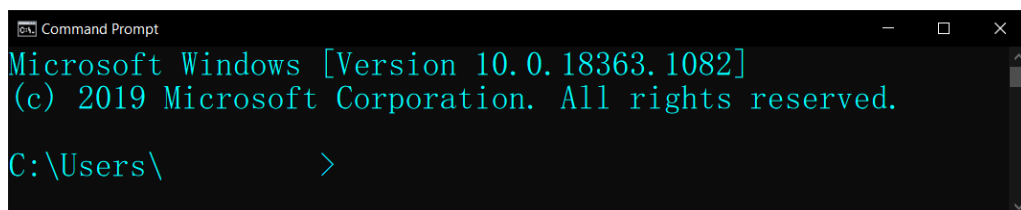
CLI Tools

Windows Tools

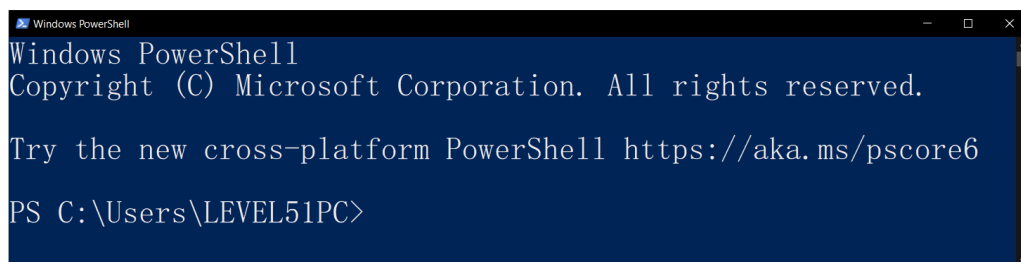
- **Windows Terminal*** : แอปพลิเคชันสำหรับผู้ใช้เครื่องมือคำสั่ง Command Line และ Shell เช่น Secure Shell, Command Prompt, PowerShell และ Windows Subsystem For Linux (WSL) โหลดได้จาก Microsoft Store ที่ติดมากับระบบปฏิบัติการ Windows 10



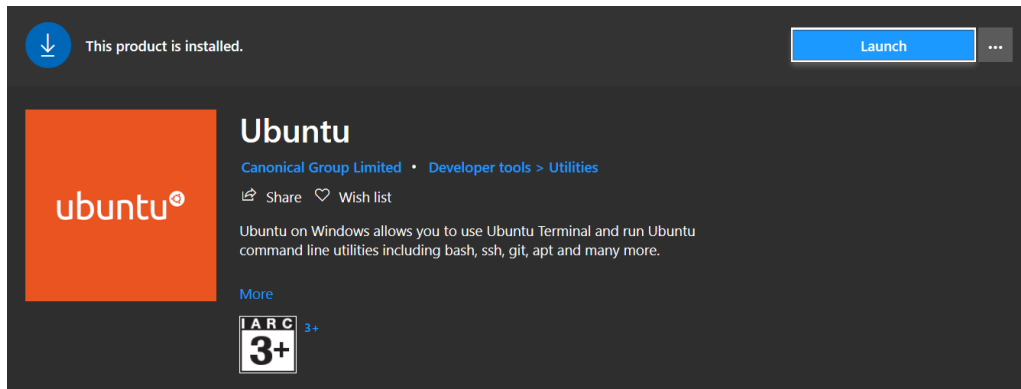
- **Command Prompt (CMD)** : เป็นเครื่องมือที่ติดมากับระบบปฏิบัติการ Windows ใช้สำหรับแปล Command Line ที่ใส่เข้าไปให้ทำงาน โดยสามารถใส่คำสั่งทีละบรรทัด หรือใช้สคริปต์รวมถึงใช้ Batch Files ก็ได้



- **Powershell** : ภาษา Shell Script ที่ Microsoft สร้างขึ้นมาให้ผู้ดูแลระบบงานบน Windows ใช้สั่งให้เครื่องทำงาน โดย PowerShell นี้ติดมากับ Windows ใช้ได้กับ Windows และอีกหลายโปรแกรม ซึ่งดีกว่าการใช้คำสั่งใน DOS โดยทำงานอยู่บน .Net Framework การไม่ต้องยึดติดกับ GUI ตลอดเวลา จะช่วยให้ทำงานเป็น Automate ได้มากขึ้น สะบาย ๆ ว่าอะไรที่ต้องซ้ำ ๆ ให้เขียนเป็น PowerShell Script เก็บไว้ใช้ได้ มันช่วยให้ทำงานได้เร็วขึ้น เป็นมาตรฐาน ลดความผิดพลาด



- **Ubuntu (Subsystem in Windows 10)** : เป็นฟีเจอร์ที่ทำให้ผู้ใช้งาน Windows 10 สามารถรัน Ubuntu ได้แบบ Native บน Windows 10 เช่น สามารถ Browse ไฟล์ของ Windows, ใช้ Network Interface ตัวเดียวกับ Windows (Mac Address และ IP เป็นตัวเดียวกับ Windows), Windows สามารถเห็นโปรเซสที่รันโดย Ubuntu และสามารถ Kill process เหล่านั้นได้ด้วย



Linux Tools

→ Terminal :

```

pi@eXit-PI: ~
File Edit Tabs Help
pi@eXit-PI:~ $ echo "YAYYYYYYYY"
YAYYYYYYYY
pi@eXit-PI:~ $ 

```

→ nmap : ใช้สำหรับ scan port ที่ server ใด ๆ

```

root@wks01:/home/vivek# nmap --top-ports 10 192.168.1.1

Starting Nmap 5.00 ( http://nmap.org ) at 2012-11-27 03:30 IST
Interesting ports on 192.168.1.1:
PORT      STATE SERVICE
21/tcp    closed ftp
22/tcp    open  ssh
23/tcp    closed telnet
25/tcp    closed smtp
80/tcp    open  http
110/tcp   closed pop3
139/tcp   closed netbios-ssn
443/tcp   closed https
445/tcp   closed microsoft-ds
3389/tcp  closed ms-term-serv
MAC Address: BC:AE:C5:C3:16:93 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.58 seconds

```

→ htop : มีลักษณะคล้าย Task Manager ในระบบปฏิบัติการ Windows

```

~/.projects/htop
1  [|||||] 34.3% Aug
2  [|||||] 55.0%
3  [|||||] 43.0%
4  [|||||] 47.0%
Mem[|||||] 1.16G/7.81G Tasks: 55, 165 thr: 3 running
Swap[|||||] OK/0K Load average: 0.64 0.38 0.29
Uptime: 05:19:59
Battery: 35.5% (Running on A/C)

PID USER   PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
5177 hishan  20   0 35020  5000 4592  S   0.0  0.1  0:00.00  gmain
5176 hishan  20   0 2952  2000 1976  S   0.0  0.0  0:00.05  /bin/dbus-daemon --conf ig-file=/System/Settings/at-spi2-ac
5175 hishan  20   0 35020  5000 4592  S   0.0  0.1  0:00.00  gibus
5168 root     20   0 34456  6224 5236  S   0.0  0.1  0:02.50  /usr/lib/upower/upowerd
5170 root     20   0 34456  6224 5236  S   0.0  0.1  0:00.00  gibus
5169 root     20   0 34456  6224 5236  S   0.0  0.1  0:00.00  gmain
5165 hishan  20   0 177M 12896 6764  S   0.0  0.2  0:47.75  /usr/bin/pulseaudio --start --log-target=syslog
5309 hishan  20   0 177M 12896 6764  S   0.0  0.2  0:00.00  alsa-source-ALC
5308 hishan  20   0 177M 12896 6764  S   0.0  0.2  0:00.00  alsa-sink-ALC36
5180 hishan  20   0 177M 12896 6764  S   0.0  0.2  0:00.01  alsa-source-ALC
5174 hishan  20   0 177M 12896 6764  S   0.0  0.2  0:45.67  alsa-sink-ALC36
5160 hishan  20   0 32288 11616 10624 S   0.7  0.1  0:00.67  xfsettingsd
5167 hishan  20   0 32288 11616 10624 S   0.0  0.1  0:00.53  gmain
5159 hishan  20   0 35076 17196 14320 S   0.0  0.2  0:01.17  xfce4-power-manager
5161 hishan  20   0 35076 17196 14320 S   0.0  0.2  0:00.00  gibus
5150 hishan  20   0 64348 31912 22820 S   0.0  0.4  0:00.68  nm-applet
5207 hishan  20   0 64348 31912 22820 S   0.0  0.4  0:00.00  gibus
5146 hishan  20   0 46952 22548 16712 S   0.0  0.3  0:01.52  xfdesktop
5211 hishan  20   0 46952 22548 16712 S   0.0  0.3  0:00.53  gmain
5144 hishan  20   0 33156 13072 12216 S   0.0  0.2  0:00.02  thunar-daemon
5153 hishan  20   0 33156 13072 12216 S   0.0  0.2  0:00.00  gmain
5142 hishan  20   0 39672 21724 17008 S   0.0  0.3  0:04.26  xfce4-panel
19006 hishan  20   0 18388 8600 7012  S   0.0  0.1  0:00.14  urxvt -cr green -fn *-lode-x -fb *-lode-x -fi *-lode-x -fb
19007 hishan  20   0 8788 5088 3280  S   0.0  0.1  0:00.09  zsh

F1 Help F2 Setup F3 Search F4 Filter F5 Sort F6 Collapse F7 Nice F8 Force F9 Kill F10 Quit

```

MacOS Tools

→ Terminal : มีลักษณะคล้าย Terminal ใน Linux

```

21:43:29 tsbest@Bmac.local ~ echo "Basic command line"
Basic command line
21:44:01 tsbest@Bmac.local ~

```

Format CLI

การใช้ CLI เช่น `ls -a`, `ping 192.168.1.1` เป็นต้น ซึ่งจะมีรูปแบบเป็นไปตามรูปแบบด้านล่างได้ดังนี้

```
<command> <flag> <input>
```

CLI

man Command

```
man <flag> <command>
```

เป็นคำสั่งใช้สำหรับอธิบายรายละเอียดของ Command ที่ต้องการ

```
pi@eXit~PI:~ $ man
What manual page do you want?
pi@eXit~PI:~ $
```

ตัวอย่างการใช้งาน man whoami

```

pi@eXit-Pl: ~
WHOAMI(1)                                User Commands                                WHOAMI(1)

NAME
    whoami - print effective userid

SYNOPSIS
    whoami [OPTION]...

DESCRIPTION
    Print the user name associated with the current effective user ID. Same as id -un.

    --help display this help and exit

    --version
        output version information and exit

AUTHOR
    Written by Richard Mlynarik.

REPORTING BUGS
    GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
    Report whoami translation bugs to <https://translationproject.org/team/>

COPYRIGHT
    Copyright © 2018 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later
    <https://gnu.org/licenses/gpl.html>.
    This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent per-
    mitted by law.

SEE ALSO
    Full documentation at: <https://www.gnu.org/software/coreutils/whoami>
    or available locally via: info '(coreutils) whoami invocation'

GNU coreutils 8.30                                February 2019                                WHOAMI(1)
~
~

```

System Category

>> uname

```
uname <flag>
```

เป็นคำสั่งสำหรับแสดงข้อมูลระบบของ Linux ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ uname -a
Linux eXit-PI 5.4.51-v8+ #1327 SMP PREEMPT Thu Jul 23 11:11:34 BST 2020 aarch64 GNU/Linux
```

>> hostname

```
hostname <flag>
```

เป็นคำสั่งสำหรับแสดงชื่อ Host ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ hostname
eXit~PI
```

>> whoami

```
whoami <flag>
```

เป็นคำสั่งสำหรับแสดงว่าล็อกอินเข้ามาด้วย User ใด ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ whoami
pi
```

>> date

```
date <flag>
```

เป็นคำสั่งสำหรับแสดงวัน และเวลาปัจจุบัน ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ date
Sun  1 Nov 10:01:27 GMT 2020
```

File Category

>> ls

```
ls <flag>
```

เป็นคำสั่งสำหรับแสดงข้อมูลของไฟล์ และไดเรกทอรี ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ ls
Arduino          Documents        Public           test.py
Bookshelf        Downloads        python-lab       thinclient_drives
brute-force.py    Music           Templates        Videos
Desktop          Pictures         test0002.cpp     vim
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
pi@eXit~PI:~ $ ls -la
total 252
drwxr-xr-x 25 pi pi 4096 Nov 1 15:41 .
drwxr-xr-x 3 root root 4096 May 27 16:53 ..
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 4 pi pi 4096 Aug 27 13:44 .arduino15
-rw----- 1 pi pi 8623 Nov 1 10:31 .bash_history
-rw-r--r-- 1 pi pi 220 May 27 16:53 .bash_logout
-rw-r--r-- 1 pi pi 3523 May 27 16:53 .bashrc
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
.
.
.
```

>> pwd

```
pwd <flag>
```

เป็นคำสั่งสำหรับแสดง Path ปัจจุบัน ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ pwd
/home/pi
```

>> mkdir

```
mkdir <flag> <name>
```

เป็นคำสั่งสำหรับสร้าง ไดเรกทอรี ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ mkdir test
pi@eXit~PI:~ $ ls -l
```

```
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
drwxr-xr-x 1 pi pi 4096 Nov 1 10:04 test
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

>> rm

```
rm <flag> <file/directory>
```

เป็นคำสั่งสำหรับลบไฟล์ หรือไดเรกทอรี ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ rm -r test
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

>> cp

```
cp <flag> <src. file/directory> <dst. file/directory>
```

เป็นคำสั่งสำหรับ Copy ไฟล์ หรือไดเรกทอรี ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ cp ./test.py ./Arduino
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
```



```
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test2.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

>> mv

```
mv <flag> <src. file/directory> <dst. file/directory>
```

เป็นคำสั่งสำหรับย้ายไฟล์ หรือไดเรกทอรี แต่สามารถใช้ในการเปลี่ยนชื่อไฟล์ หรือไดเรกทอรีได้เช่นกัน ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ mv ./test2.py ./test0002.py
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test0002.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

>> cat / head / tail

```
cat <flag> <file>
head <flag> <file>
tail <flag> <file>
```

เป็นคำสั่งสำหรับแสดงข้อมูลในไฟล์ ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ cat test0002.py
import time
while(True):
    printf("Hello world")
    time.sleep(2)
```

>> tar

```
tar <flag> <file> <file>
```

เป็นคำสั่งสำหรับบีบ หรือแตกไฟล์ ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ tar cf index.zip index.html
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
```

```

drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
-rw-r--r-- 1 pi pi 15970 Nov 1 10:14 index.html
-rw-r--r-- 1 pi pi 20480 Nov 1 10:18 index.zip
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test0002.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
pi@eXit~PI:~ $ tar xf index.zip
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
-rw-r--r-- 1 pi pi 122 Nov 1 10:38 brute-force.py
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
-rw-r--r-- 1 pi pi 15970 Nov 1 10:14 index.html
-rw-r--r-- 1 pi pi 20480 Nov 1 10:18 index.zip
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test0002.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim

```

>> gzip

```
gzip <flag> <file> <file>
```

เป็นคำสั่งสำหรับบีบไฟล์เป็นนามสกุล .gz ดังตัวอย่างการใช้งานด้านล่างนี้

```

pi@eXit~PI:~ $ gzip index.html
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
-rw-r--r-- 1 pi pi 122 Nov 1 10:38 brute-force.py
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
-rw-r--r-- 1 pi pi 6417 Nov 1 10:14 index.html
-rw-r--r-- 1 pi pi 15970 Nov 1 10:14 index.html.gz
-rw-r--r-- 1 pi pi 20480 Nov 1 10:18 index.zip
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public

```

```
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test0002.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

File Permission Category

>> chmod

```
chmod <flag> <number(option)> <file/directory>
```

Owner/Group/Public

7 7 5

(Read/Write/Execute)

0 คือ ไม่สามารถทำอะไรได้เลยกับ Files หรือ Folder นั้นๆได้เลย

1 คือ สามารถเรียกให้ Files หรือ Folder มาประมวลผลอย่างเดียว

2 คือ สามารถให้มีการเขียนใน Files หรือ Folder ได้

3 คือ สามารถให้มีการเขียนและประมวลผล ใน Files หรือ Folder ได้

4 คือ สามารถเรียกให้ Files หรือ Folder มาอ่านได้อย่างเดียว

5 คือ สามารถให้มีการอ่านและประมวลผล ใน Files หรือ Folder ได้

6 คือ สามารถให้มีการอ่านและเขียน ใน Files หรือ Folder ได้

7 คือ การกำหนดให้ Files หรือ Folder ทำได้ทุกอย่าง อ่าน เขียน และประมวลผล

โดยปกติทั่วไป เราจะกำหนดให้ไดเรกทอรีต่างๆมีสิทธิ์เป็น 644 (ไดเรกทอรีอ่านได้อย่างเดียว) และ ไฟล์ให้เป็น 775 (ไฟล์ไม่สามารถเขียนทับได้โดยผู้ใช้ภายนอก) จะเป็นการปลอดภัยที่สุด

เป็นคำสั่งสำหรับกำหนดสิทธิ์การเข้าถึงให้กับไฟล์ หรือไดเรกทอรี [สำหรับสิทธิ์ต่าง ๆ สามารถดูเพิ่มเติมได้ตามลิงค์นี้](#) ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ mv ./test2.py ./test0002.py
pi@eXit~PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
-rw-r--r-- 1 pi pi 122 Nov 1 10:38 brute-force.py
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rw-r--r-- 1 root root 68 Nov 1 10:26 test0002.py
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
pi@eXit~PI:~ $ chmod 777 test0002.cpp
pi@eXit~PI:~ $ ls -l
total 80
```

```
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
-rw-r--r-- 1 pi pi 122 Nov 1 10:38 brute-force.py
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
-rw-r--r-- 1 pi pi 15966 Nov 1 10:22 index.html
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rwxrwxrwx 1 pi pi 68 Nov 1 10:06 test0002.cpp
-rw-r--r-- 1 root root 68 Sep 14 10:26 test.py
drwxr-xr-t 2 pi pi 4096 Aug 13 17:03 thinclient\_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
```

Network Category

>> ping

```
ping <flag> <destination>
```

เป็นคำสั่งส่งแพ็คเก็ต ICMP Request เพื่อเช็คการเชื่อมต่อ ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ ping www.kmitl.ac.th
PING www.kmitl.ac.th (161.246.34.11) 56(84) bytes of data.
64 bytes from chaokhun.kmitl.ac.th (161.246.34.11): icmp_seq=1 ttl=58 time=7.00 ms
64 bytes from chaokhun.kmitl.ac.th (161.246.34.11): icmp_seq=2 ttl=58 time=1.95 ms
64 bytes from chaokhun.kmitl.ac.th (161.246.34.11): icmp_seq=3 ttl=58 time=1.71 ms
64 bytes from chaokhun.kmitl.ac.th (161.246.34.11): icmp_seq=4 ttl=58 time=1.93 ms
^C
--- www.kmitl.ac.th ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.710/3.151/7.007/2.228 ms
```

>> wget

```
wget <flag> <destination>
```

เป็นคำสั่งเพื่อดาวน์โหลดไฟล์ ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ wget http://google.com
--2020-11-02 00:47:17-- http://google.com/
Resolving google.com (google.com)... 172.217.174.174, 2404:6800:4001:807::200e
Connecting to google.com (google.com)|172.217.174.174|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.com/ [following]
--2020-11-02 00:47:17-- http://www.google.com/
Resolving www.google.com (www.google.com)... 172.217.31.68, 2404:6800:4001:804::2004
Connecting to www.google.com (www.google.com)|172.217.31.68|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html          [ <=>          ] 15.66K --.-KB/s   in 0.009s

2020-11-02 00:47:18 (1.77 MB/s) - 'index.html' saved [16033]
```

>> netstat

```
netstat <flag>
```

เป็นคำสั่งเพื่อดูสถานะการเชื่อมต่อ ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ netstat
Active Internet connections (w/o servers)

```

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	exit-Server:ssh	172.16.0.191:63284	ESTABLISHED
tcp	0	0	192.168.88.19:2222	192.168.88.10:62799	ESTABLISHED
tcp	0	0	192.168.88.27:ssh	192.168.88.10:60373	ESTABLISHED
tcp	0	300	exit-Server:ssh	10.10.10.10:7827	ESTABLISHED
tcp	0	0	192.168.88.19:ssh	192.168.88.10:62783	ESTABLISHED

```
tcp      0      0 192.168.88.28:ssh    192.168.88.10:60250  ESTABLISHED
```

Search Category

>> grep

```
grep <flag> "<keyword/pattern>" <file>
```

เป็นคำสั่งเพื่อค้นหารูปแบบหรือคีย์เวิร์ด ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~ $ grep "doc*" ./index.html
<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage"
lang="th"><head><meta
content="text/html; charset=UTF-8"http-equiv="Content-Type"><metacontent="/images/branding/g
ogle/1x/googleg_standard_color_128dp.png" itemprop="image"><title>Google</title><script
nonce="Ik4buhglWdsaq/XGT79gaA==">(function() {window.google=
.
.
.
```

>> find

```
find <path> <flag> "<file>"
```

เป็นคำสั่งเพื่อค้นหาชื่อไฟล์ ว่าตั้งอยู่ที่ใด ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~/home $ ls -l
total 4
Drwxr-xr-x 25 pi pi 4096 Nov  1 10:22 pi
pi@eXit~PI:~/home $ find ./pi/ -name "test.py"
./pi/python-lab/test.py
./pi/test.py

pi@eXit~PI:~/home $ find . -type f -name "*.txt"
.find: './exit/.gnupg': Permission denied
./exit/kippo/data/userdb.txt
./exit/kippo/data/lastlog.txt
find: './exit/.cache': Permission denied
find: './exit/.ssh': Permission denied
find: './exit/.config': Permission denied
./exit/kippo-master/data/userdb.txt
./exit/kippo-master/data/lastlog.txt

find + grep + pipe
pi@eXit~PI:~/home $ find ./pi/ -name "test.py" -print0 | xargs -0
grep "ALOHA"
./pi/test.py:# ALOHA: Hahaha
```

Remote Category

>> ssh

```
ssh <username>@<Destination_IP>
```

เป็นคำสั่งเพื่อเชื่อมต่อไปยังโฮสต์ผ่านช่องทางระยะไกลผ่าน SSH Port ดังตัวอย่างการใช้งานด้านล่างนี้

```
pi@eXit~PI:~/home $ ssh pi@192.168.88.14
pi@192.168.88.14's password:
Linux eXit-PI 5.4.51-v8+ #1327 SMP PREEMPT Thu Jul 23 11:11:34 BST 2020 aarch64

This programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
Individual files in /usr/share/doc/*/copyright.

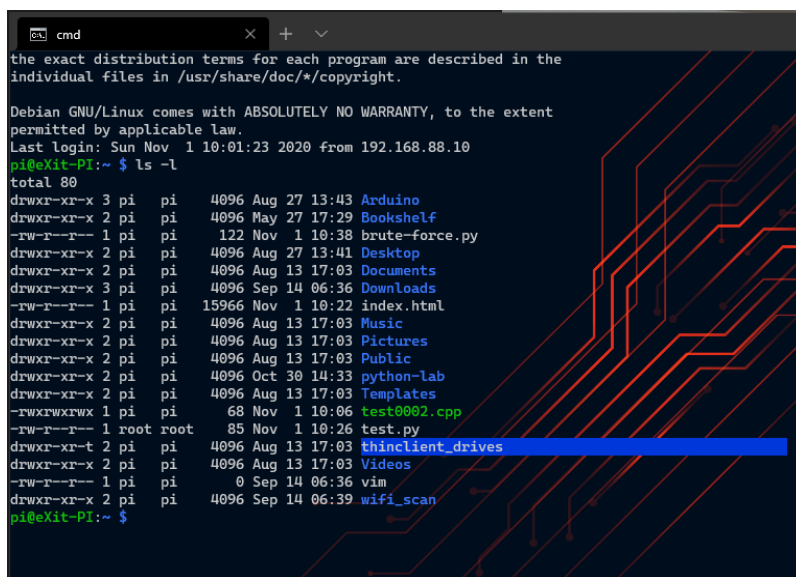
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
Permitted by applicable law.

Last login: Sun Nov  1 10:01:23 2020 from 192.168.88.10
```

* เมื่อก่อนนิยมใช้ telnet ซึ่งปัจจุบันไม่มีความปลอดภัยแล้ว

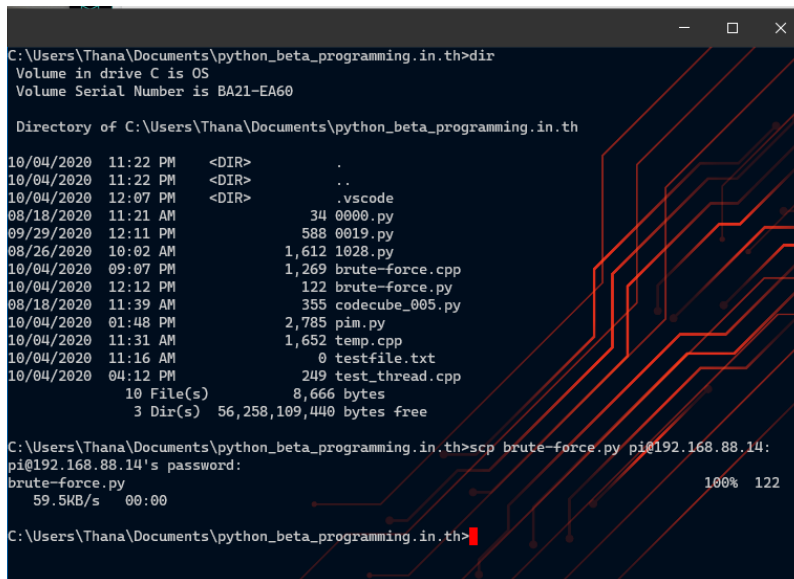
File Transfer Category

- scp: Secure copy file to remote



```
cmd
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Nov  1 10:01:23 2020 from 192.168.88.10
pi@eXit-PI:~ $ ls -l
total 80
drwxr-xr-x 3 pi pi 4096 Aug 27 13:43 Arduino
drwxr-xr-x 2 pi pi 4096 May 27 17:29 Bookshelf
-rw-r--r-- 1 pi pi 122 Nov  1 10:38 brute-force.py
drwxr-xr-x 2 pi pi 4096 Aug 27 13:41 Desktop
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Documents
drwxr-xr-x 3 pi pi 4096 Sep 14 06:36 Downloads
-rw-r--r-- 1 pi pi 15966 Nov  1 10:22 index.html
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Music
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Pictures
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Public
drwxr-xr-x 2 pi pi 4096 Oct 30 14:33 python-lab
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Templates
-rwxrwxrwx 1 pi pi 68 Nov  1 10:06 test0002.cpp
-rw-r--r-- 1 root root 85 Nov  1 10:26 test.py
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 thinclient_drives
drwxr-xr-x 2 pi pi 4096 Aug 13 17:03 Videos
-rw-r--r-- 1 pi pi 0 Sep 14 06:36 vim
drwxr-xr-x 2 pi pi 4096 Sep 14 06:39 wifi_scan
pi@eXit-PI:~ $
```


A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Users\Thana\Documents\python_beta_programming.in.th'. The prompt is 'C:\Users\Thana\Documents\python_beta_programming.in.th>dir'. The output shows the volume information for drive C (OS, BA21-EA60) and a directory listing of the current directory. The listing includes files like .vscode, 0000.py, 0019.py, 1028.py, brute-force.cpp, brute-force.py, codecube_005.py, pim.py, temp.cpp, testfile.txt, and test_thread.cpp, along with their sizes and timestamps. Below the listing, it shows '10 File(s) 8,666 bytes' and '3 Dir(s) 56,258,109,440 bytes free'. The prompt then changes to 'C:\Users\Thana\Documents\python_beta_programming.in.th>scp brute-force.py pi@192.168.88.14:'. The output shows the password prompt, the file being transferred (brute-force.py), the transfer speed (59.5KB/s), and the progress (100% 122). The prompt returns to 'C:\Users\Thana\Documents\python_beta_programming.in.th>'.

```
C:\Users\Thana\Documents\python_beta_programming.in.th>dir
Volume in drive C is OS
Volume Serial Number is BA21-EA60

Directory of C:\Users\Thana\Documents\python_beta_programming.in.th

10/04/2020  11:22 PM  <DIR>          .
10/04/2020  11:22 PM  <DIR>          ..
10/04/2020  12:07 PM  <DIR>          .vscode
08/18/2020  11:21 AM             34 0000.py
09/29/2020  12:11 PM             588 0019.py
08/26/2020  10:02 AM            1,612 1028.py
10/04/2020  09:07 PM            1,269 brute-force.cpp
10/04/2020  12:12 PM             122 brute-force.py
08/18/2020  11:39 AM             355 codecube_005.py
10/04/2020  01:48 PM             2,785 pim.py
10/04/2020  11:31 AM            1,652 temp.cpp
10/04/2020  11:16 AM              0 testfile.txt
10/04/2020  04:12 PM             249 test_thread.cpp
               10 File(s)      8,666 bytes
               3 Dir(s)  56,258,109,440 bytes free

C:\Users\Thana\Documents\python_beta_programming.in.th>scp brute-force.py pi@192.168.88.14:
pi@192.168.88.14's password:
brute-force.py
59.5KB/s  00:00                               100% 122

C:\Users\Thana\Documents\python_beta_programming.in.th>
```

Directory Traverse Category

- cd .. => To go up one level of the directory tree
- cd => to go to \$HOME directory
- cd /test => Change to /test directory

Other Category

- clear=> Clear screen

LINUX COMMANDS CHEAT SHEET

SYSTEM

uname -a ==> Display linux system information
 uname -r ==> Display kernel release information
 uptime ==> Show how long the system has been running + load
 hostname ==> Show system host name
 hostname -i ==> Display the IP address of the host
 last reboot ==> Show system reboot history
 date ==> Show the current date and time
 cal ==> Show this month calendar
 w ==> Display who is online
 whoami ==> Who you are logged in as
 finger user ==> Display information about user

HARDWARE

dmesg ==> Detected hardware and boot messages
 cat /proc/cpuinfo ==> CPU model
 cat /proc/meminfo ==> Hardware memory
 cat /proc/interrupts ==> Lists the number of interrupts per CPU per I/O device
 lshw ==> Displays information on hardware configuration of the system
 lsblk ==> Displays block device related information in Linux
 free -m ==> Used and free memory (-m for MB)
 lspci -tv ==> Show PCI devices
 lsusb -tv ==> Show USB devices
 dmidecode ==> Show hardware info from the BIOS
 hdparm -i /dev/sda ==> Show info about disk sda
 hdparm -tT /dev/sda ==> Do a read speed test on disk sda
 badblocks -s /dev/sda ==> Test for unreadable blocks on disk sda

USERS

id ==> Show the active user id with login and group
 last ==> Show last logins on the system
 who ==> Show who is logged on the system
 groupadd admin ==> Add group "admin"
 useradd -c "Sam Tomshi" ==> g admin -m sam #Create user "sam"
 userdel sam ==> Delete user sam
 adduser sam ==> Add user "sam"
 usermod ==> Modify user information

FILE COMMANDS

ls -al ==> Display all information about files/ directories
 pwd ==> Show the path of current directory
 mkdir directory-name ==> Create a directory
 rm file-name ==> Delete file
 rm -r directory-name ==> Delete directory recursively
 rm -f file-name ==> Forcefully remove file
 rm -rf directory-name ==> Forcefully remove directory recursively
 cp file1 file2 ==> Copy file1 to file2
 cp -r dir1 dir2 ==> Copy dir1 to dir2, create dir2 if it doesn't exist
 mv file1 file2 ==> Rename source to dest / move source to directory
 ln -s /path/to/file-name link-name ==> Create symbolic link to file-name
 touch file ==> Create or update file
 cat > file ==> Place standard input into file
 more file ==> Output contents of file
 head file ==> Output first 10 lines of file
 tail file ==> Output last 10 lines of file
 tail -f file ==> Output contents of file as it grows starting with the last 10 lines
 gpg -c file ==> Encrypt file
 gpg file.gpg ==> Decrypt file
 wc ==> print the number of bytes, words, and lines in files
 xargs ==> Execute command lines from standard input

PROCESS RELATED

ps ==> Display your currently active processes
 ps aux | grep 'telnet' ==> Find all process id related to telnet process
 pmap ==> Memory map of process
 top ==> Display all running processes
 kill pid ==> Kill process with mentioned pid
 killall proc ==> Kill all processes named proc
 pkill process-name ==> Send signal to a process with its name
 bg ==> Resumes suspended jobs without bringing them to foreground
 fg ==> Brings the most recent job to foreground
 fg n ==> Brings job n to the foreground

FILE PERMISSION RELATED

chmod octal file-name ==> Change the permissions of file to octal
 Example
 chmod 777 /data/test.c ==> Set rwx permission for owner,group,world
 chmod 755 /data/test.c ==> Set rwx permission for owner,rx for group and world
 chown owner-user file ==> Change owner of the file
 chown owner-user:owner-group file-name ==> Change owner and group owner of the file
 chown owner-user:owner-group directory ==> Change owner and group owner of the directory

NETWORK

ip addr show ==> Display all network interfaces and ip address (a iproute2 command, powerful than ifconfig)
 ip address add 192.168.0.1 dev eth0 ==> Set ip address
 ethtool eth0 ==> Linux tool to show ethernet status
 mii-tool eth0 ==> Linux tool to show ethernet status
 ping host ==> Send echo request to test connection
 whois domain ==> Get who is information for domain
 dig domain ==> Get DNS information for domain
 dig -x host ==> Reverse lookup host
 host google.com ==> Lookup DNS ip address for the name
 hostname -i ==> Lookup local ip address
 wget file ==> Download file
 netstat -tupl ==> Listing all active listening ports

COMPRESSION / ARCHIVES

tar cf home.tar home ==> Create tar named home.tar containing home/
 tar xf file.tar ==> Extract the files from file.tar
 tar czf file.tar.gz files ==> Create a tar with gzip compression
 gzip file ==> Compress file and renames it to file.gz

INSTALL PACKAGE

rpm -i pkgname.rpm ==> Install rpm based package
 rpm -e pkgname ==> Remove package

INSTALL FROM SOURCE

./configure
 make
 make install

SEARCH

grep pattern files ==> Search for pattern in files
 grep -r pattern dir ==> Search recursively for pattern in dir
 locate file ==> Find all instances of file
 find /home/tom -name "index*" ==> Find files names that start with "index"
 find /home -size +10000k ==> Find files larger than 10000k in /home

LOGIN (SSH AND TELNET)

ssh user@host ==> Connect to host as user
 ssh -p port user@host ==> Connect to host using specific port
 telnet host ==> Connect to the system using telnet port

FILE TRANSFER

scp ==> Secure copy
 scp file.txt server2:/tmp ==> Secure copy file.txt to remote host /tmp folder
 rsync -a /home/apps/backup/ ==> Synchronize source to destination

DISK USAGE

df -h ==> Show free space on mounted filesystems
 df -i ==> Show free inodes on mounted filesystems
 fdisk -l ==> Show disks partitions sizes and types
 du -ah ==> Display disk usage in human readable form
 du -sh ==> Display total disk usage on the current directory
 findmnt ==> Displays target mount point for all filesystem
 mount device-path mount-point ==> Mount a device

DIRECTORY TRAVERSE

cd .. ==> To go up one level of the directory tree
 cd ==> Go to \$HOME directory
 cd /test ==> Change to /test directory



MORE DETAILED : [HTTP://LINUXIDE.COM/GUIDE/LINUX-COMMAND-SHELF.HTML](http://LINUXIDE.COM/GUIDE/LINUX-COMMAND-SHELF.HTML)

https://github.com/ISAG-Lab/Chamber_Command