Simple Linux

Network programming

2021/09/27

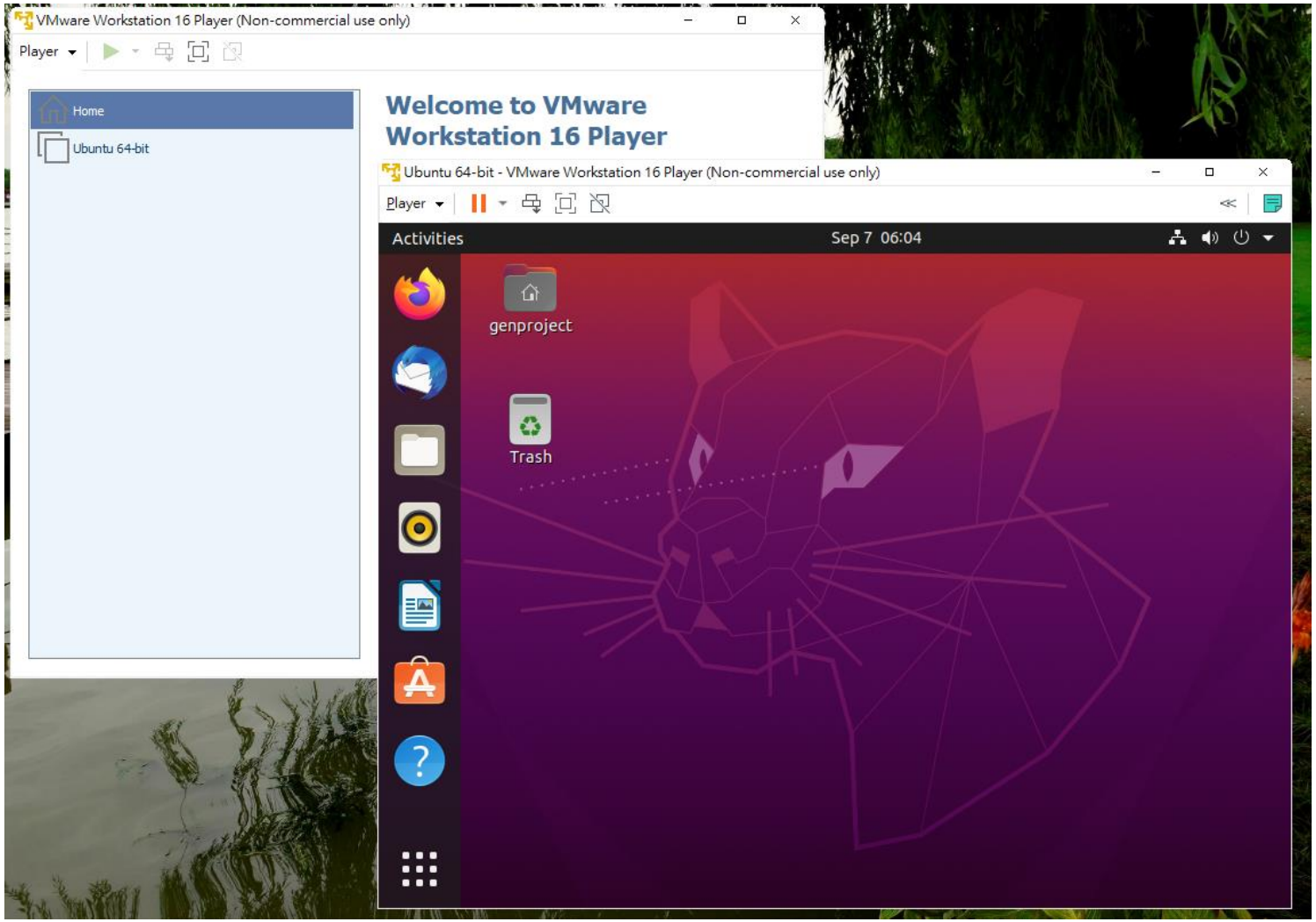
Outline

- Introduction of virtual machines and an installation tutorial
- How to connect to a remote machine ?
- Basic Linux commands
- Basic Linux programming
- Networking tools on Linux

Outline

- Introduction of virtual machines and an installation tutorial
- How to connect to a remote machine ?
- Basic Linux commands
- Basic Linux programming
- Networking tools on Linux

Virtual Machines



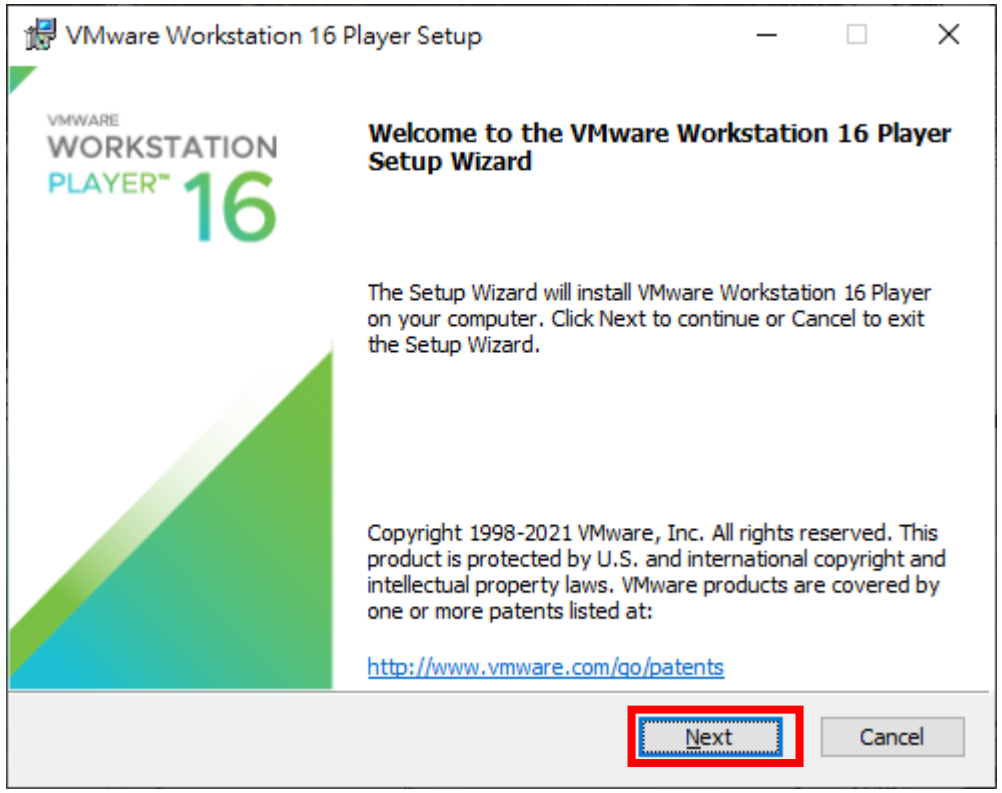
Install VMware Workstation

1. Download VMware workstation
   - https://www.vmware.com/products/player/playerpro-evaluation.html
   - Or google VMware workstation player

Install VMware Workstation

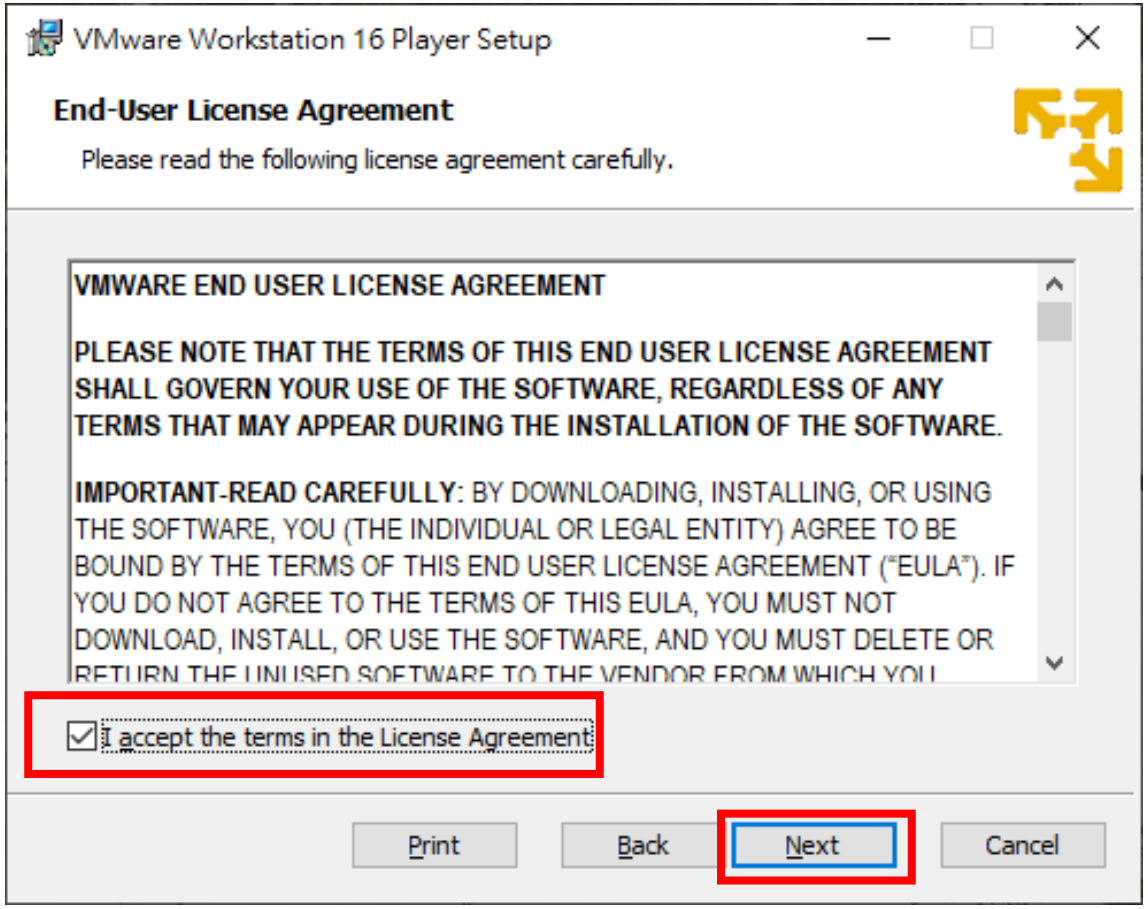
2. Click the program you have downloaded

2. Click “Next” to continue installation



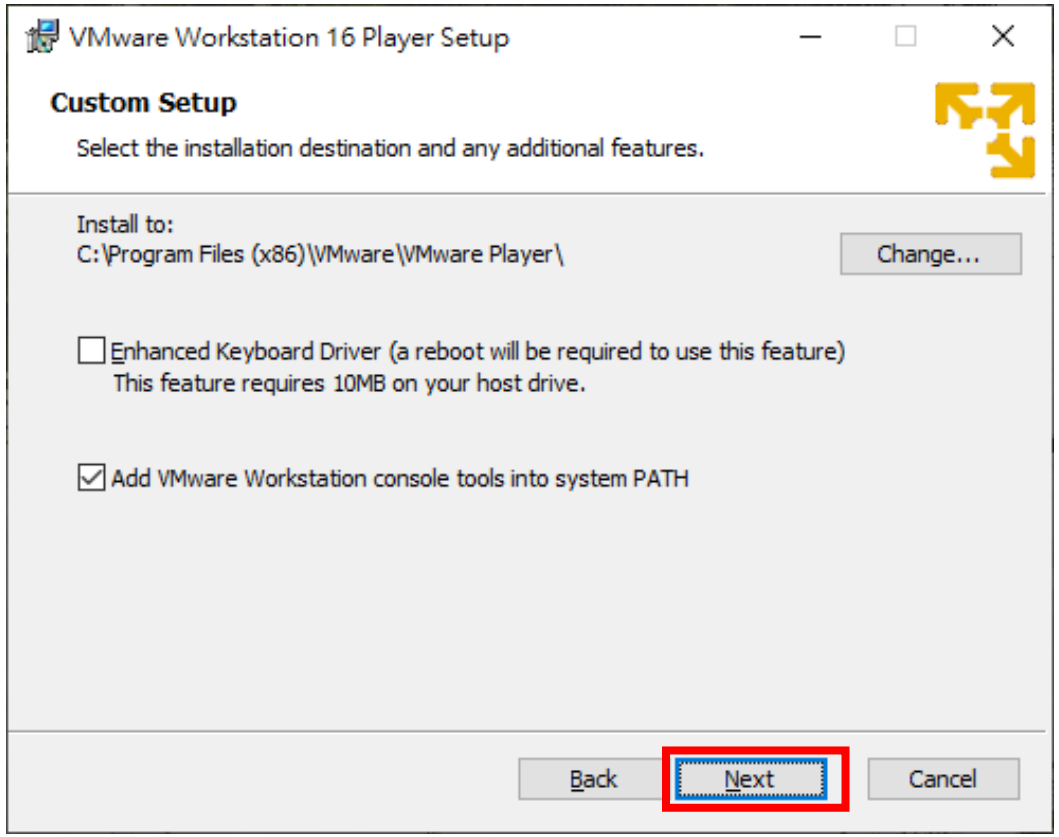
Install VMware Workstation

4. Click “I accept the terms in the license Agreement” and then “Next”



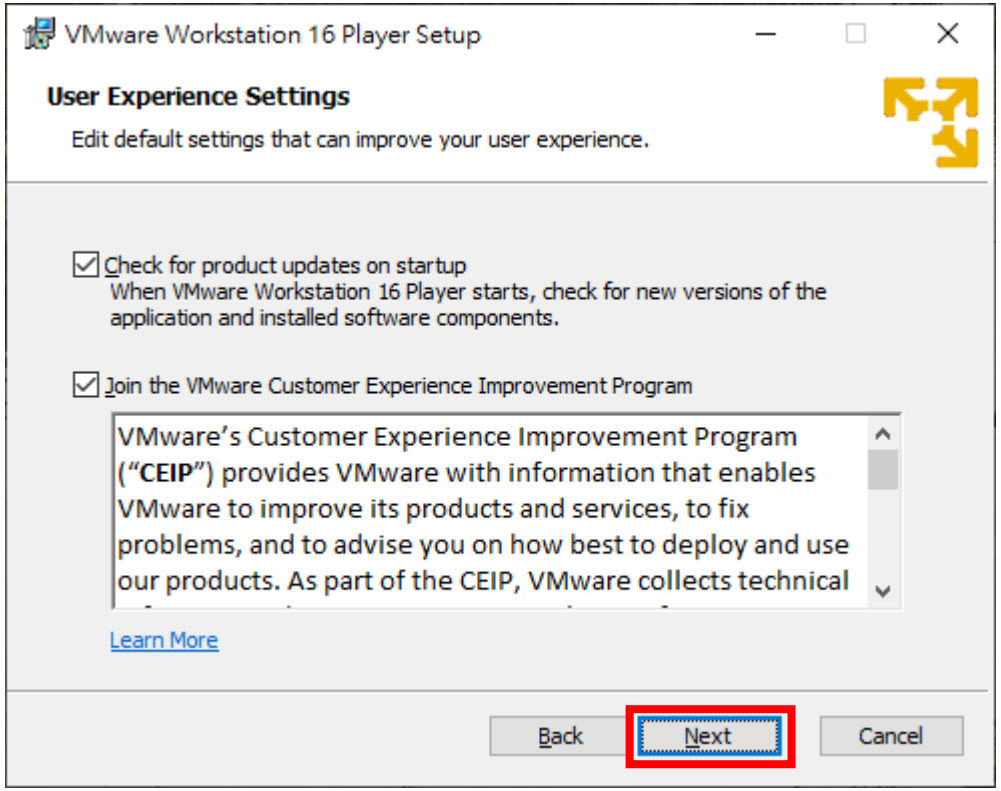
Install VMware Workstation

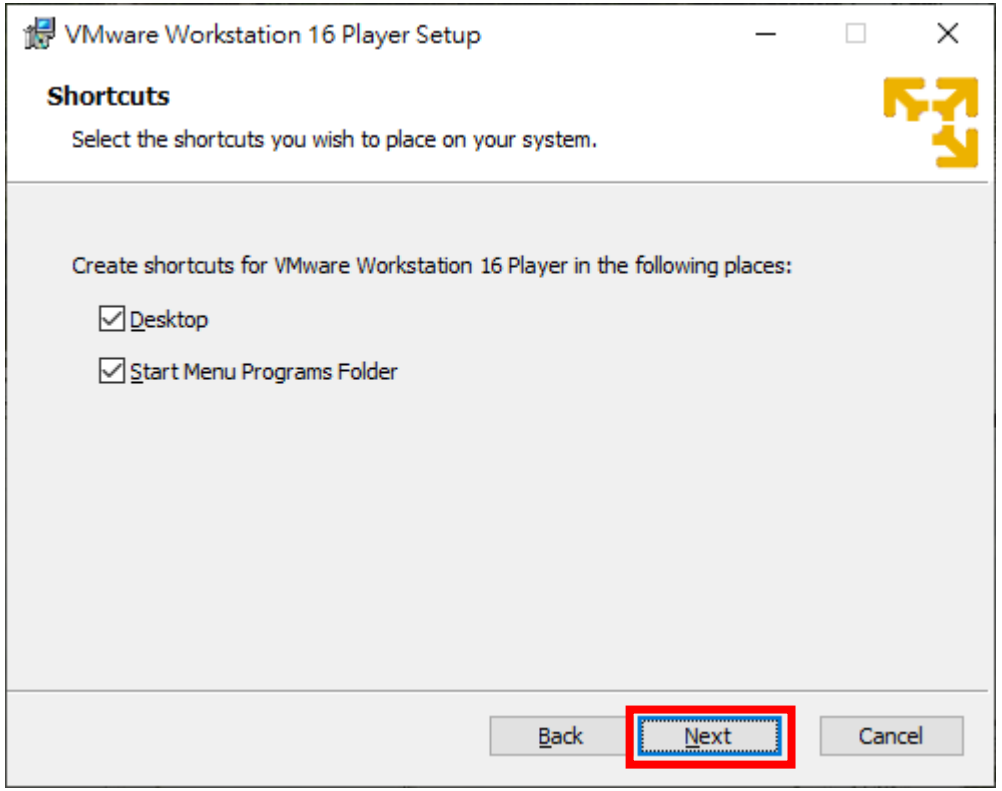
5. Click “Next” if you accept the default setup



Install VMware Workstation

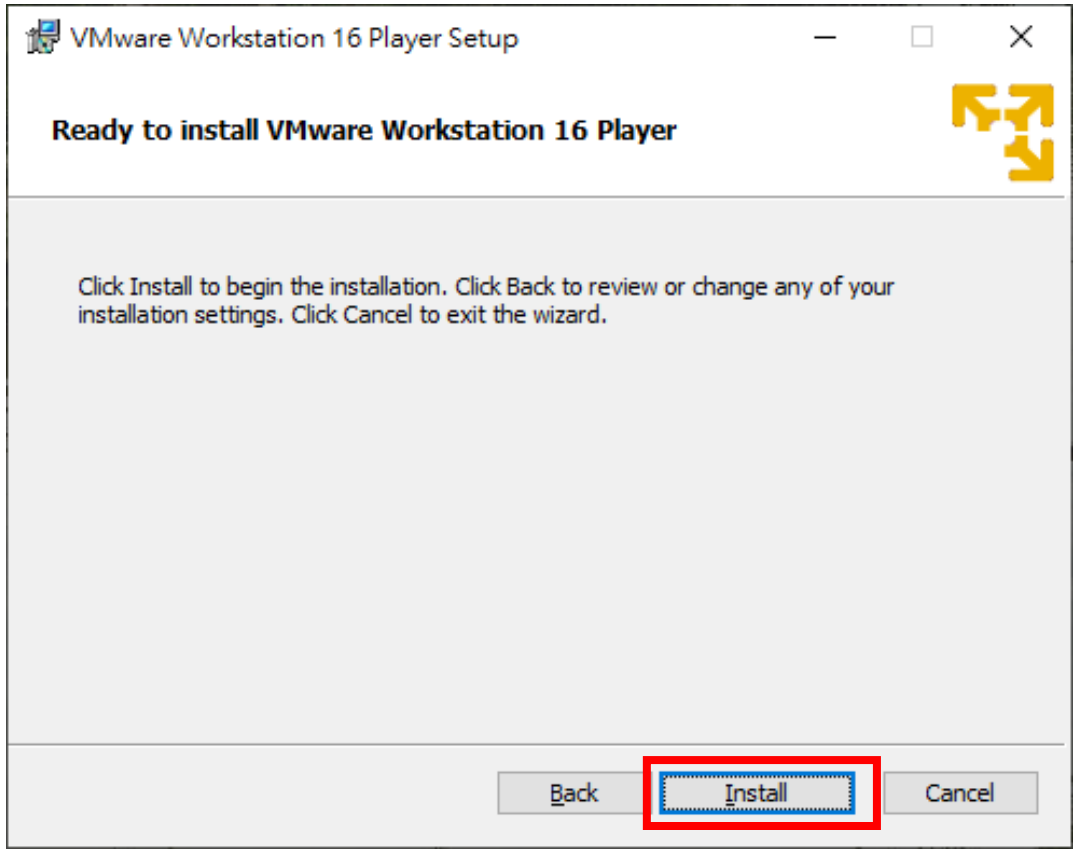
6. Click “Next” if you accept the default setup





Install VMware Workstation

7. Click “Install” to start the installation



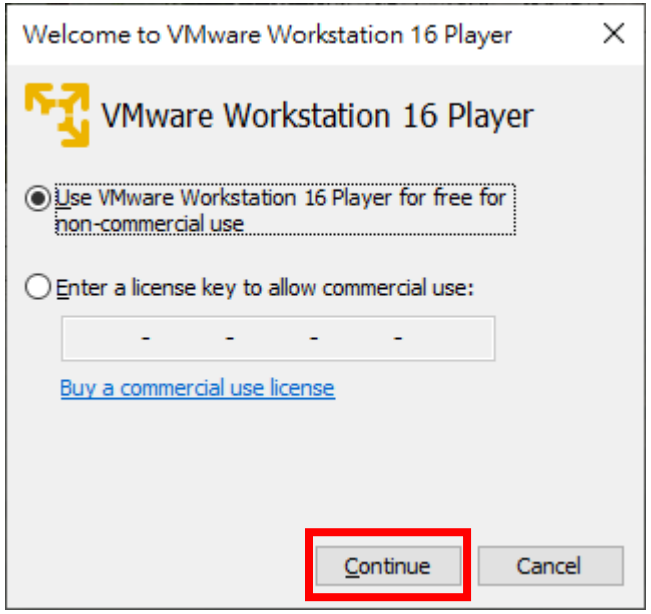
Install the Guest OS

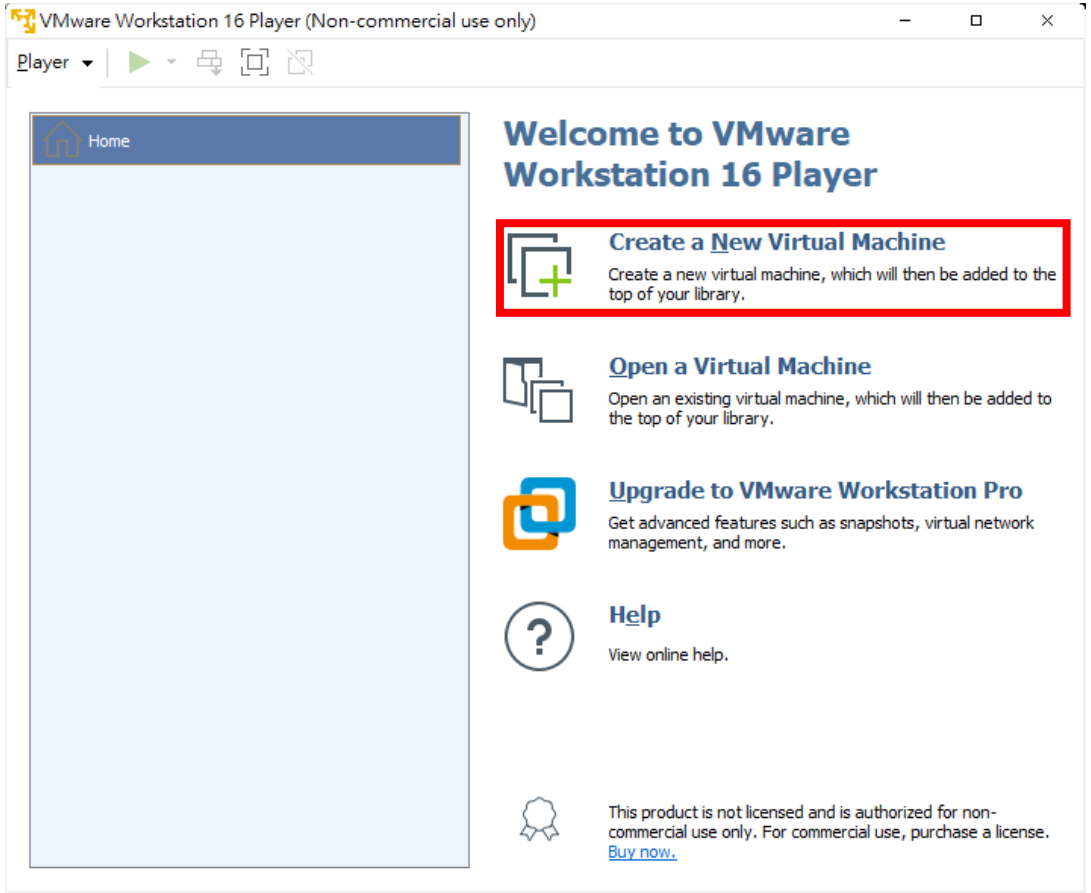
8. Install the guest OS
   1) Download the Linux distribution you like
   2) In this tutorial, I will use Ubuntu 20.04 LTS
   3) Download Ubuntu from here: http://ubuntu.cs.nctu.edu.tw/ubuntu-release/20.04/

ubuntu-20.04.3-desktop-amd64.iso	19-Aug-2021 11:06	3071934464
ubuntu-20.04.3-desktop-amd64.iso.torrent	26-Aug-2021 09:42	234738
ubuntu-20.04.3-desktop-amd64.iso.zsync	26-Aug-2021 09:42	6000109
ubuntu-20.04.3-desktop-amd64.list	19-Aug-2021 11:06	29476
ubuntu-20.04.3-desktop-amd64.manifest	19-Aug-2021 10:56	59485

Install the Guest OS

a. Back to VMware workstation player
b. Free for non-commercial use
c. Click “Create a New Virtual Machine”

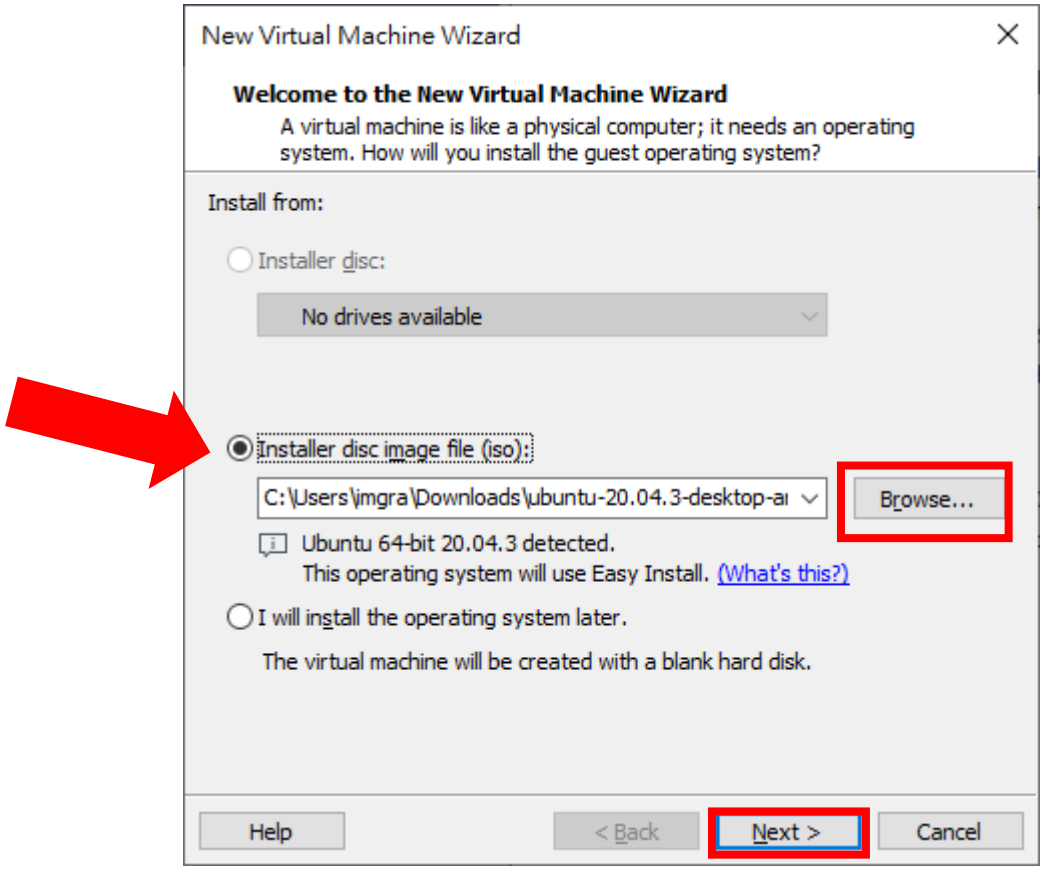




Install the Guest OS

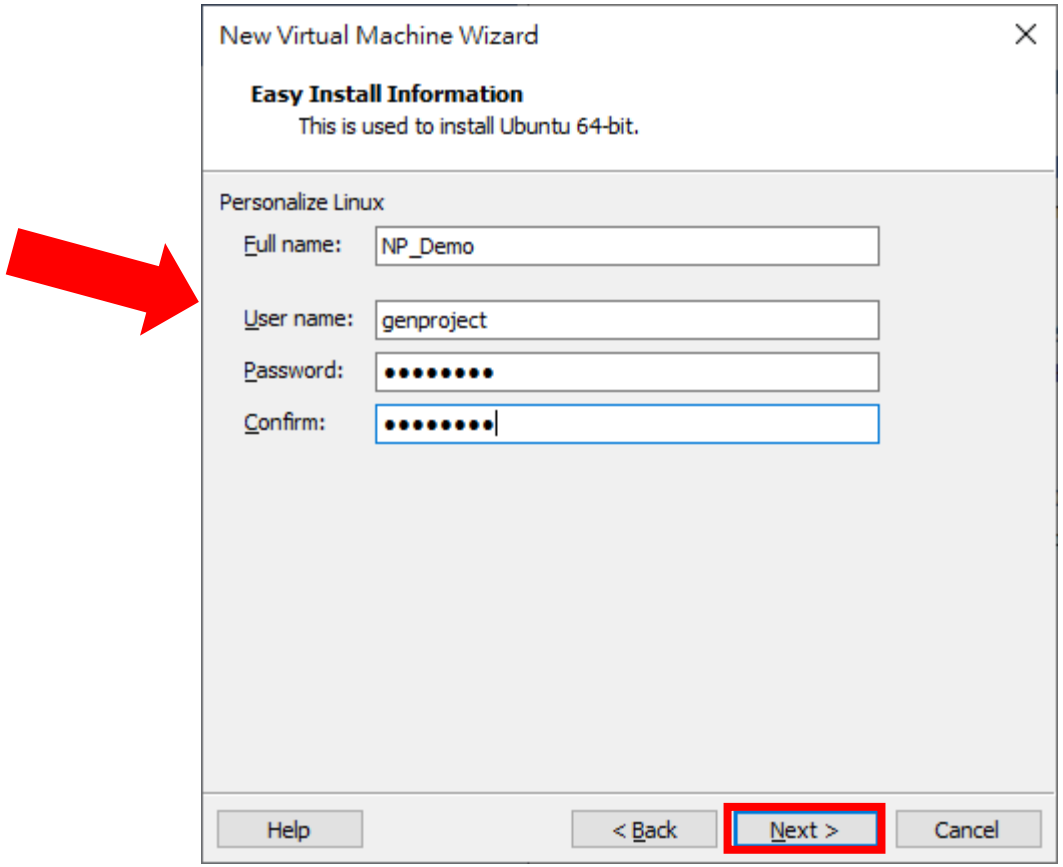
d. Choose “Installer disc image file(iso)” and press “Browse” to find the ISO file you download from website

e. Press “Next” to continue



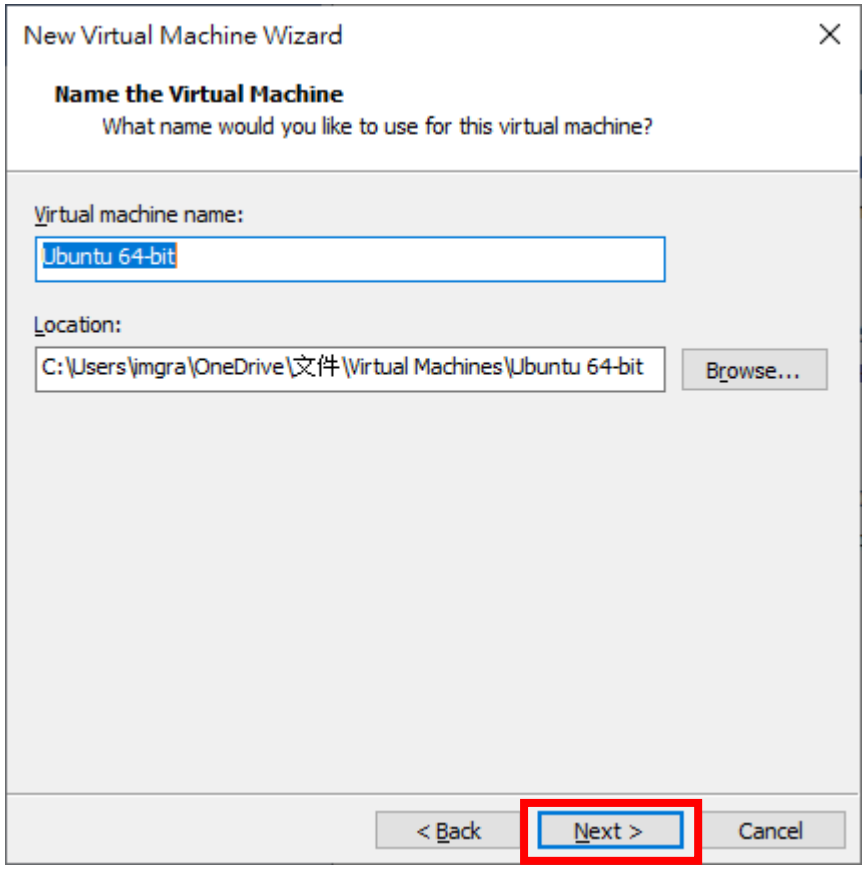
Install the Guest OS

f. Enter the user name and the password for your guest machine
g. Press “Next”



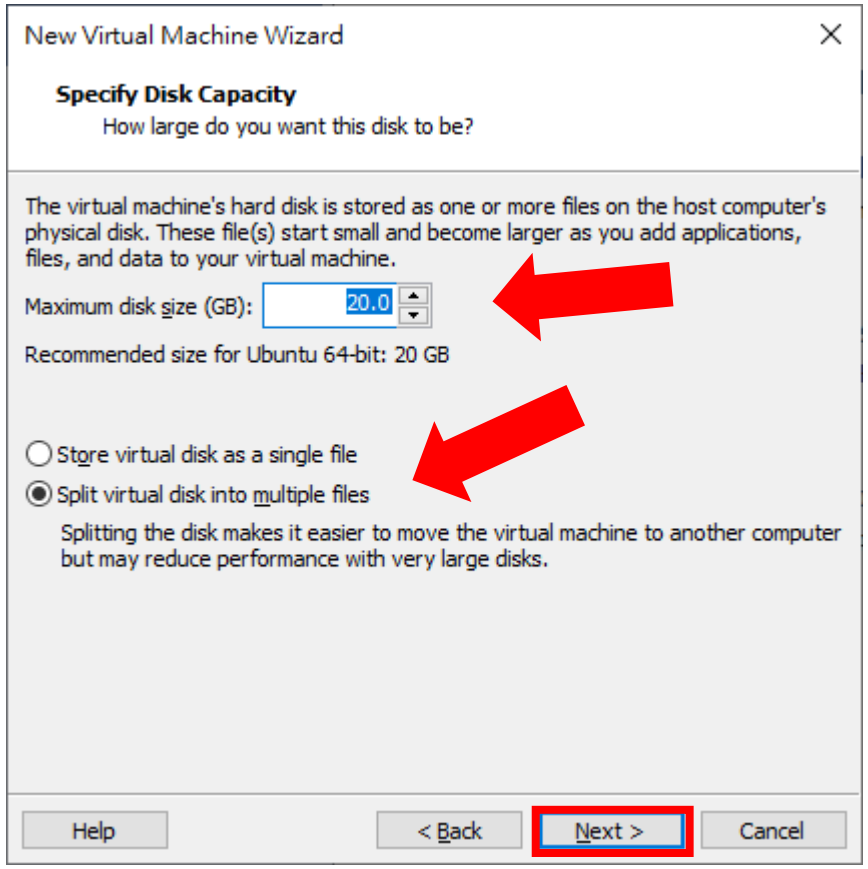
Install the Guest OS

h. Click “Next” if you accept default setup



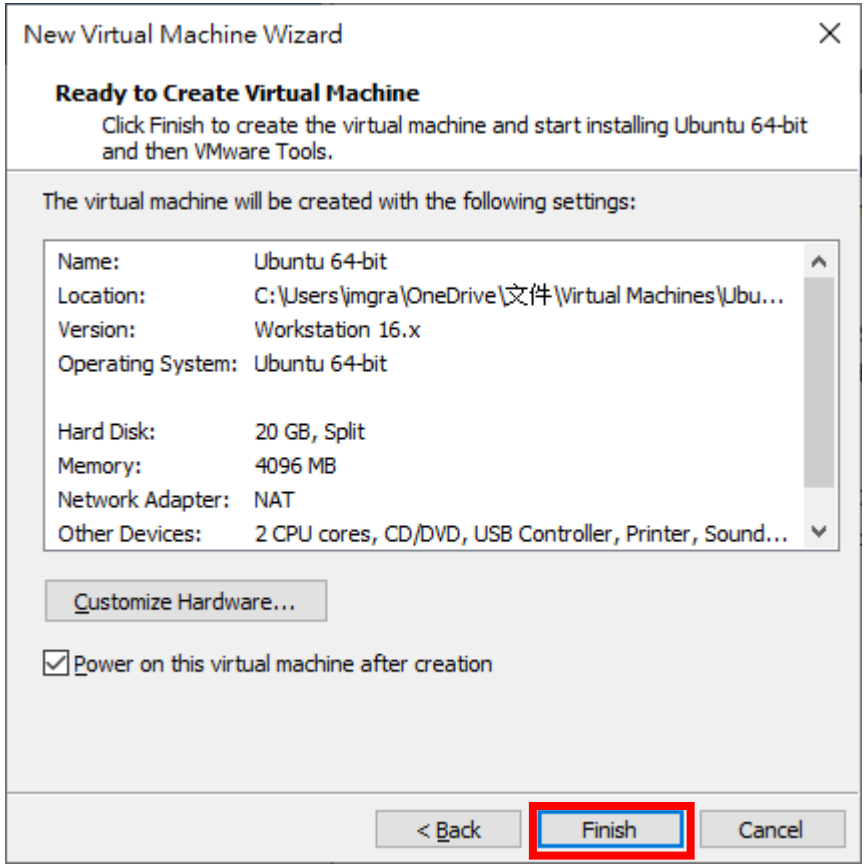
Install the Guest OS

i. Set the maximum disk size of your guest OS
j. Choose “Split virtual disk into multiple files”
k. Click “Next”



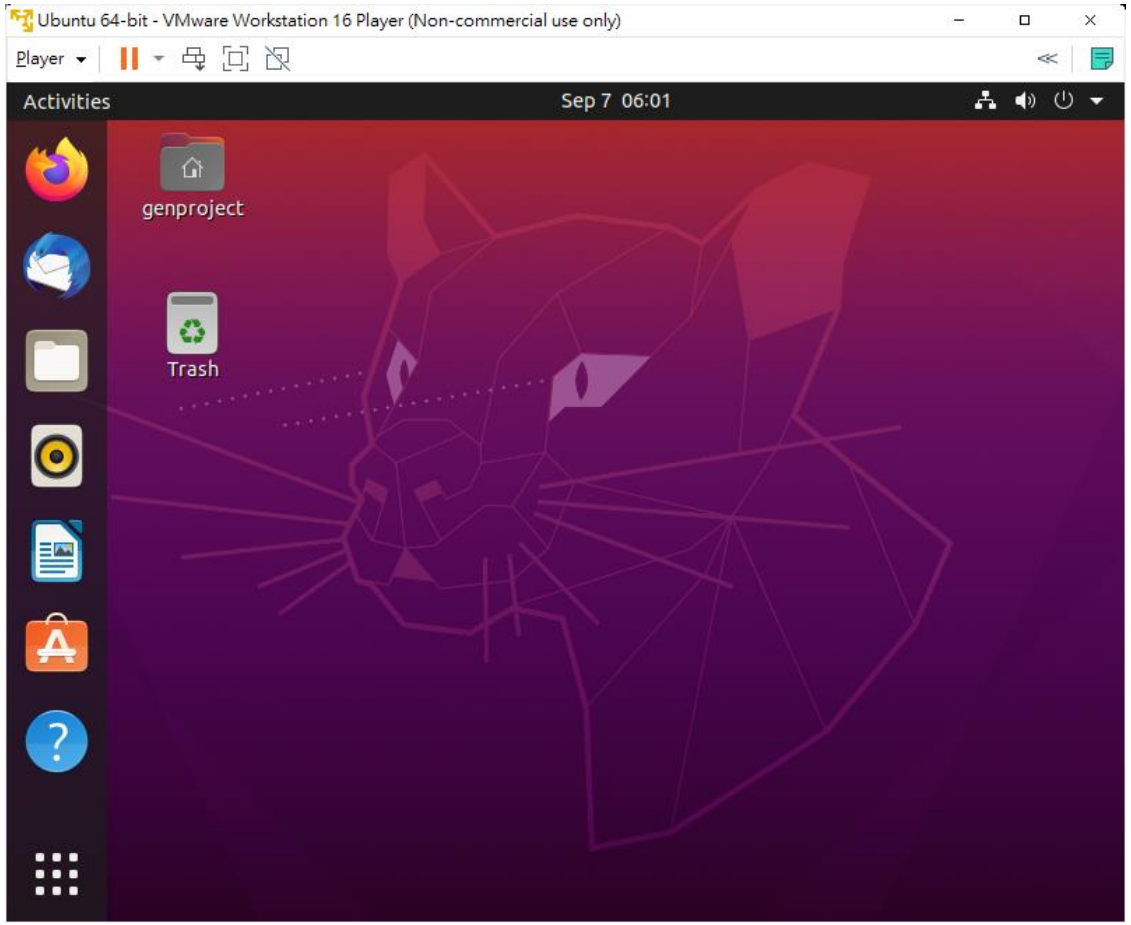
Install the Guest OS

l. Press “Finish” if you accept default setup
m. The guest OS will be powered on automatically.



Install the Guest OS

n. After waiting for a while, you can get a whole new guest OS on your computer.



Outline

- Introduction of virtual machines and an installation tutorial
- How to connect to a remote machine ?
- Basic Linux commands
- Basic Linux programming
- Networking tools on Linux

To start with...

- Download putty.exe
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
  - Or google putty
- Install

Package files

You probably want one of these. They include versions of all the PuTTY utilities.

(Not sure whether you want the 32-bit or the 64-bit version? Read the FAQ entry.)

MSI ('Windows Installer')

64-bit x86:	putty-64bit-0.76-installer.msi	(or by FTP)	(signature)
64-bit Arm:	putty-arm64-0.76-installer.msi	(or by FTP)	(signature)
32-bit x86:	putty-0.76-installer.msi	(or by FTP)	(signature)

Unix source archive

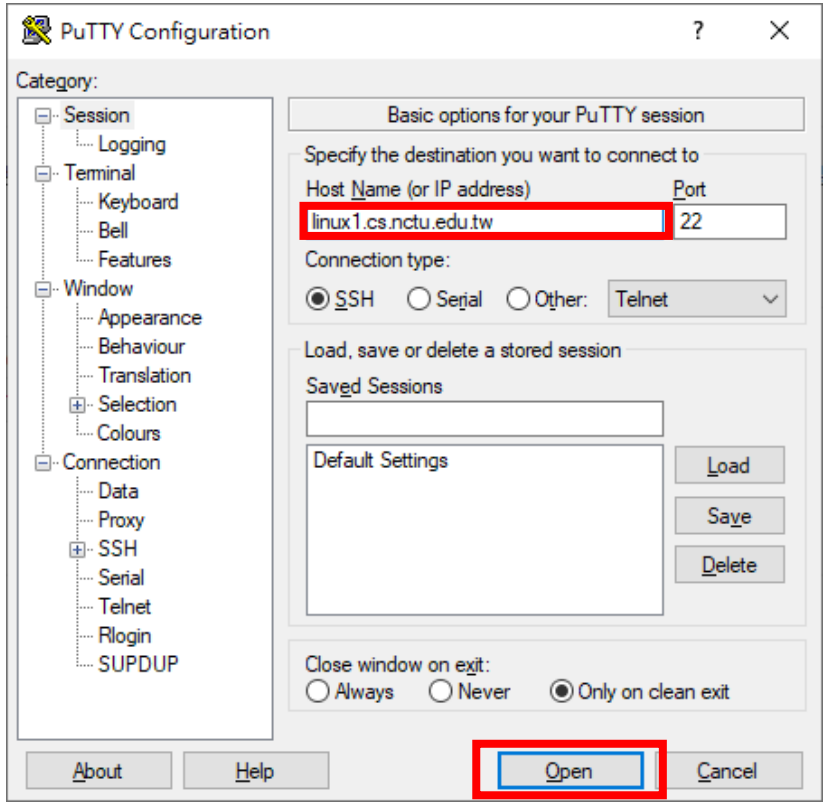
.tar.gz:	putty-0.76.tar.gz	(or by FTP)	(signature)

To start with... (Continued)

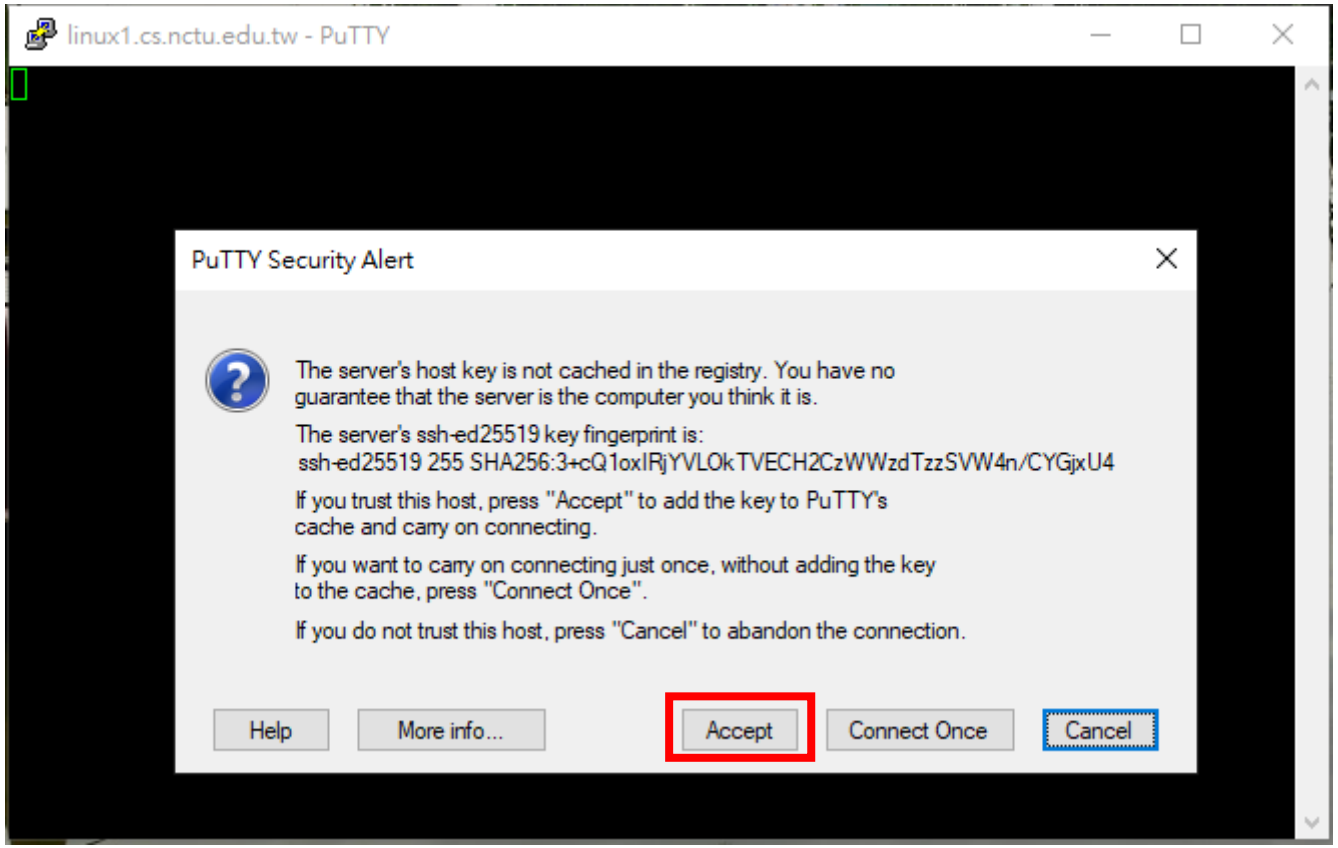
- 參考交大資工系工作站
  https://it.cs.nycu.edu.tw/workstation-guide

連線位址

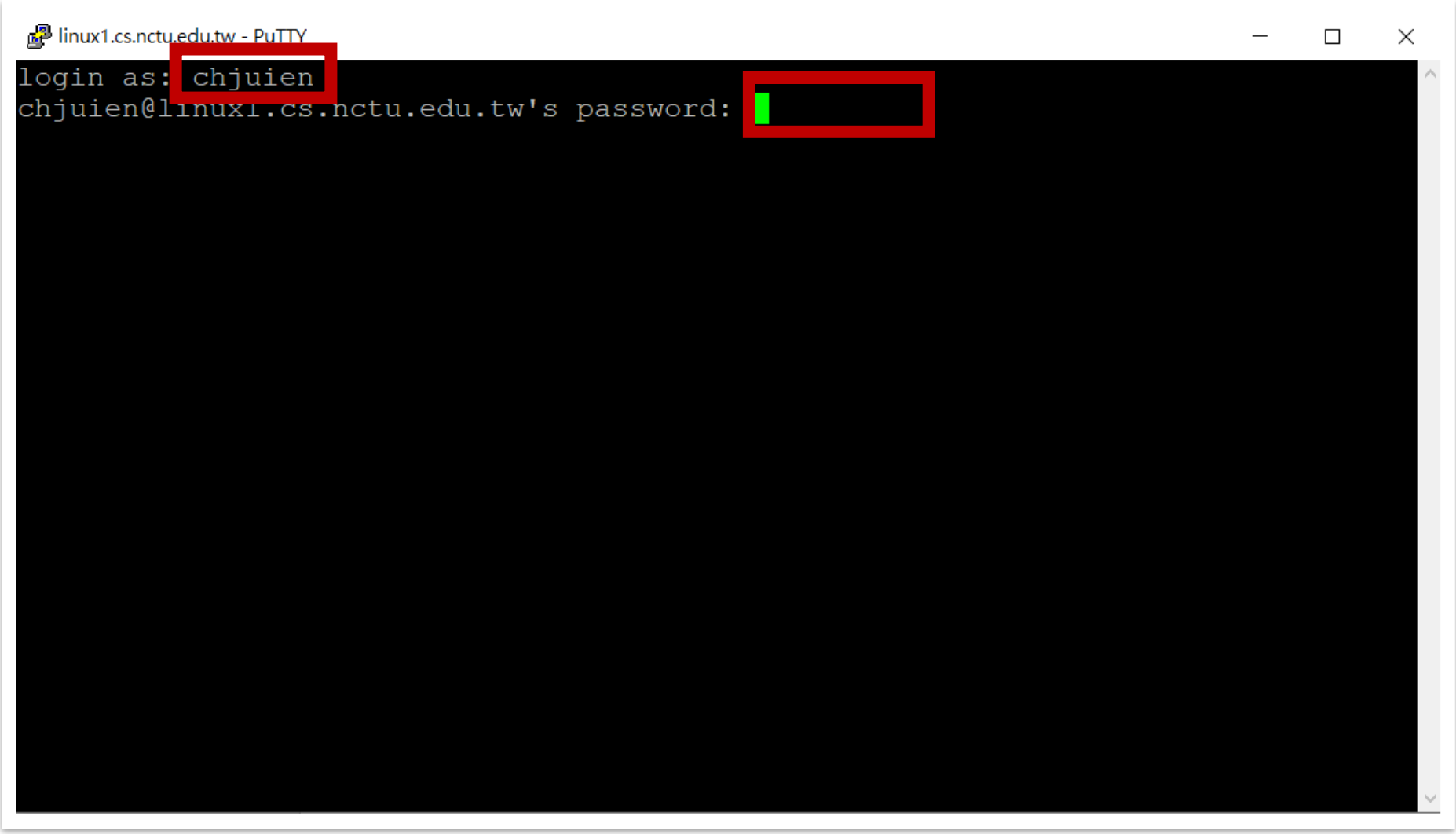
- Linux 工作站: `linux{1,2,3,4}.cs.nctu.edu.tw`



To start with... (Continued)



To start with... (Continued)



To start with... (Continued)

chjuien@linux1:~

```
 3. For rights of other users, please don't occupy /tmp as yours,
    please use (re)nice/taskset/cpuset to lower the priority of high-loading pro
cesses,
    and please use ipcrm to clear shared memory after using it.
= Disk Usage ==================================================================
 Mail:                                                  0% 0/256000 KB

 Home:                                                  1% 8.21/500 MB

= Process =====================================================================
  PID TTY          TIME CMD
 6635 pts/47   00:00:00 tcsh
 6643 pts/47   00:00:00 csShell
 6739 pts/47   00:00:00 ps
= Information =================================================================
 Current Time: Sat Aug 31 10:53:53 CST 2019
 Online Users: 11
= CSCC Announce ===============================================================
2019-08-29 資工系計算機中心人才招募
https://cscc.cs.nctu.edu.tw/news/202

                                CS Computer Center <help@cs.nctu.edu.tw>
[chjuien@linux1 ~]$
```

Outline

- Introduction of virtual machines and an installation tutorial
- How to connect to a remote machine ?
- Basic Linux commands
- Basic Linux programming
- Networking tools on Linux

Basic Linux commands

command	explanation	中文解釋
ls	List directory contents	把現在資料夾下的檔案顯示出來
pwd	Print name from working directory	看現在在哪個資料夾底下
mkdir <dir>	Make directories	建立資料夾
cd <dir>	Change directory	換資料夾
mv <src file> <dst file>	Move (rename) file	移動檔案 (重新命名) 檔案
cp <src file> <dst file>	Copy file	複製檔案
rm <file>	Remove file	刪除檔案
clear	Clean up the screen	把畫面清乾淨

Directory Listing

```
[chjuien@linux1 ~]$
```

account@server hostname [current directory]

```
[chjuien@linux1 ~]$ ls
```

- You can type “ls”, which means “list “, to show all the files in the home directory.

command	explanation
ls ~	**List contents in user home directory.**
ls .	**List contents in current directory.**
ls ..	**List contents in parent directory.**

The ‘man’ Command

- Use command ‘man’ to discover more usages.

```
[chjuien@linux1 ~]$ man ls
```

- man ls
  - ls –a : show all files, including files starting with . , which are hidden files.
  - ls -l : show long information of contents.
  - ls –al : show all the files and their long information.
  - ls <path> : show the content of given path
- You can also man the function you may use in the program.
  - man printf
  - man strtok

The ‘man’ Command

chjuien@linux1:~

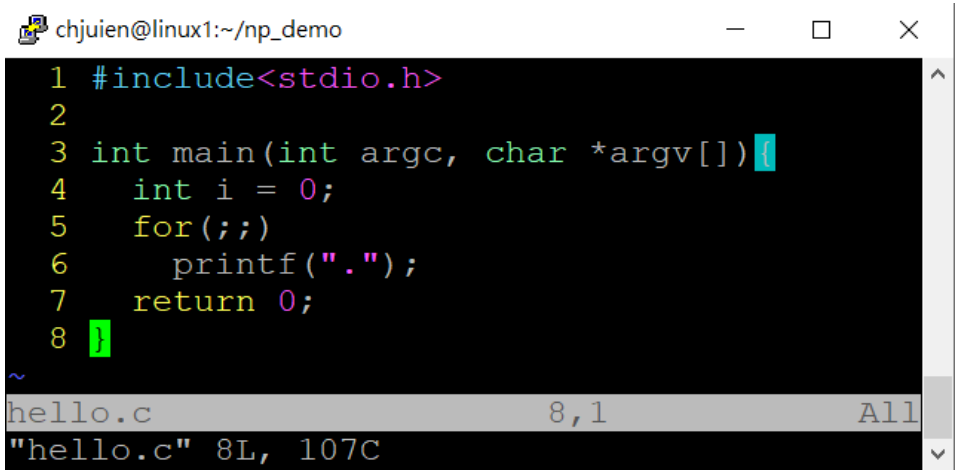
```
LS(1)                          User Commands                          LS(1)



NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about  the FILEs (the current directory by default).
       Sort entries alphabetically if none of -cftuvSUX nor --sort  is  speci-
       fied.

       Mandatory  arguments  to  long  options are mandatory for short options
       too.

       -a, --all
              do not ignore entries starting with .

       -A, --almost-all
              do not list implied . and ..

       --author
              with -l, print the author of each file
 Manual page ls(1) line 1 (press h for help or q to quit)
```

Outline

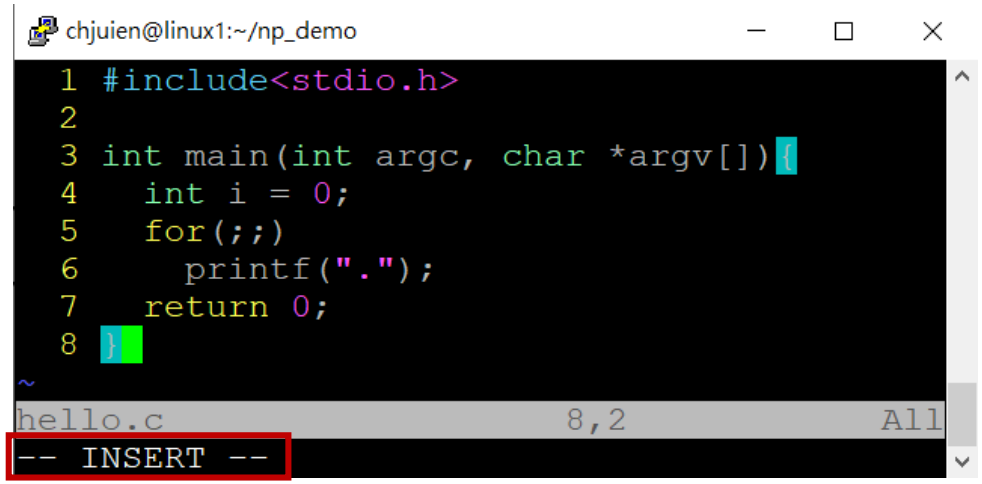
- Introduction of virtual machine and installation tutorial
- How to connect to a remote machine ?
- Basic Linux commands
- Basic Linux programming
- Networking tools on Linux

The VIM Text Editor

- vim is a text editor
- vim hello.c

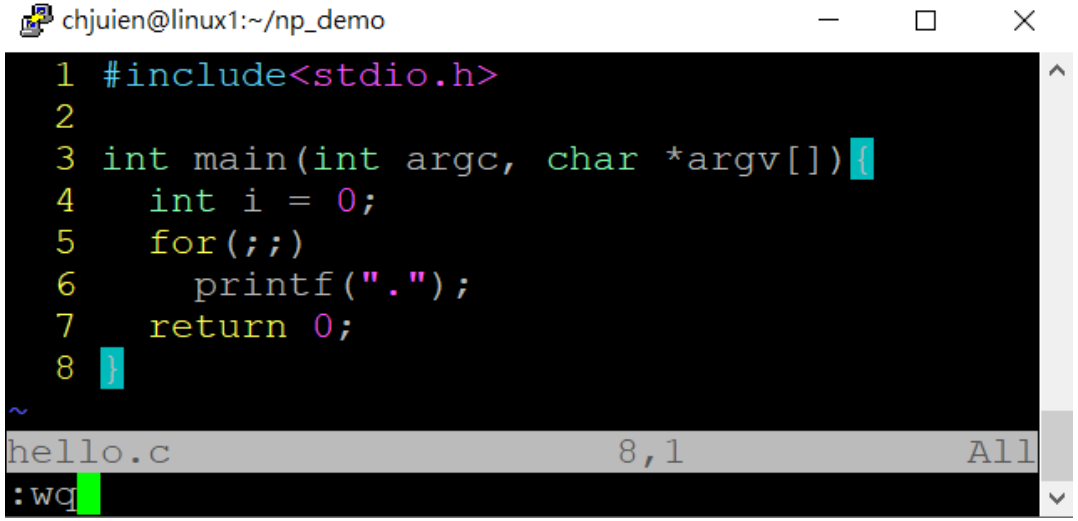


- Press ”insert” or “i” to enter insert mode, or you can type nothing.



The VIM Text Editor

- Press “Esc” to enter back to normal mode, where you can type some commands, including **“:wq”** for saving and leaving, **“:q!”** for leaving without saving, **“/int”** for searching the string “int”.
- You can pick up other usage by your self.



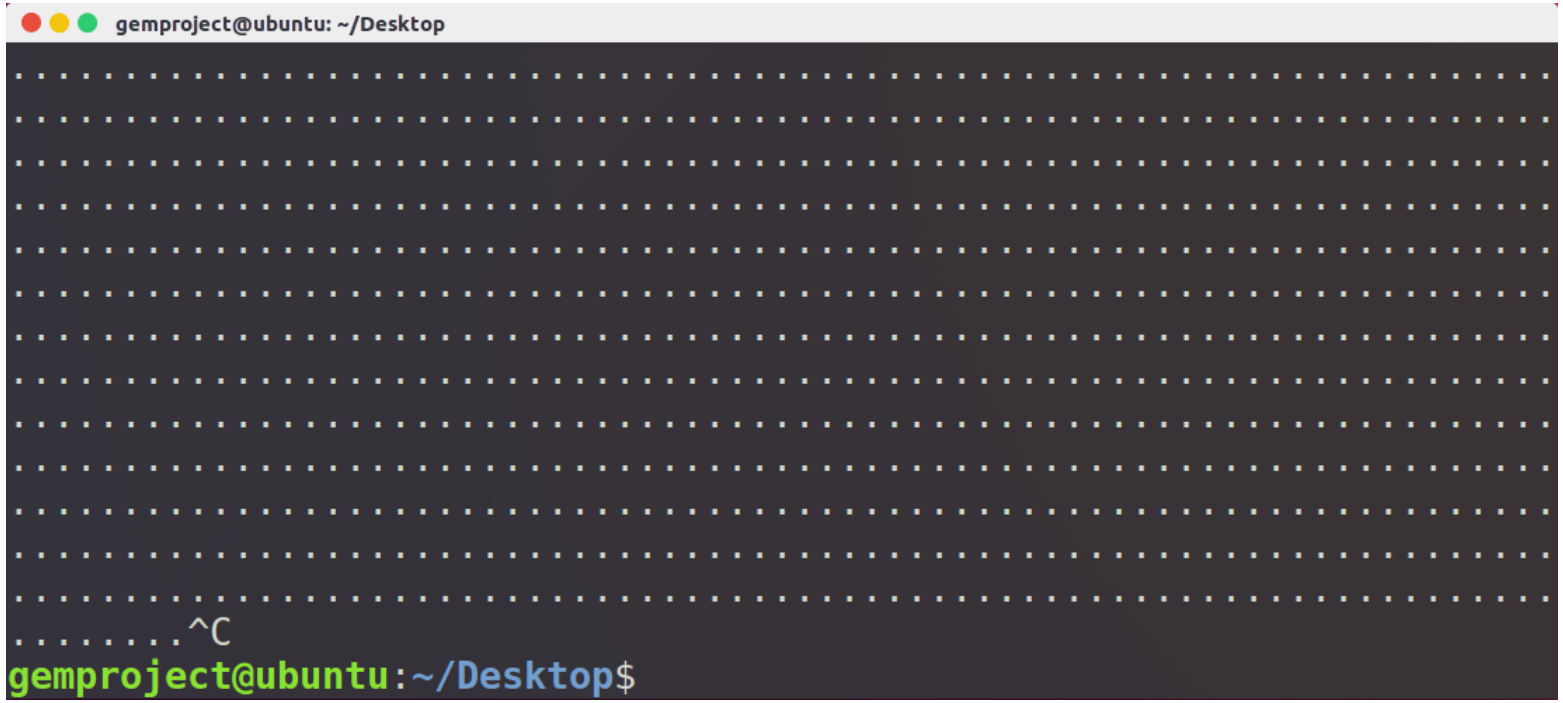
The GCC/G++ Compilers

- gcc/g++ –o <output> <filename>
  - -o
    - The program produced will be the same name as assigned.
    - If –o is not provide, a.out will be the default output file name.

```
gemproject@ubuntu: ~/Desktop
gemproject@ubuntu:~/Desktop$ ls
hello.c
gemproject@ubuntu:~/Desktop$ gcc hello.c
gemproject@ubuntu:~/Desktop$ ls
a.out  hello.c
gemproject@ubuntu:~/Desktop$ gcc -o hello hello.c
gemproject@ubuntu:~/Desktop$ ls
a.out  hello  hello.c
gemproject@ubuntu:~/Desktop$
```

Execute and Forced Terminate a Program

- Execute
  - ./a.out
  - ./hello
  - ~/*/a.out
  - ~/*/hello
- Forced Terminate
  - Ctrl+c



More explanation

Argument Variables

gemproject@ubuntu: ~/Desktop

```
1  #include<stdio.h>
2
3  int main int argc  char *argv[]){
4
5    printf("argument counts: %d\n", argc);
6
7    for(int i = 0; i < argc; i++)
8      printf("argv[%d] : %s\n", i, argv[i]);
9
10   return 0;
11 }
~
~
~
~
~
                                10,11          All
```

Argument count

Argument vector

Argument Variables

- int main (int argc, char *argv[])
- argc
  - Integer
  - The sum of given arguments count, including the program name.

```
printf("argument counts: %d\n", argc);
```

- argv
  - A string array that holds arguments.

```
for(int i = 0; i < argc; i++)
  printf("argv[%d] : %s\n", i, argv[i]);
```

Argument Variables

```
gemproject@ubuntu: ~/Desktop
gemproject@ubuntu:~/Desktop$ ./argument this is a simple demo program
argument counts: 7
argv[0] : ./argument
argv[1] : this
argv[2] : is
argv[3] : a
argv[4] : simple
argv[5] : demo
argv[6] : program
gemproject@ubuntu:~/Desktop$ ./a.out this is a simple demo program
argument counts: 7
argv[0] : ./a.out
argv[1] : this
argv[2] : is
argv[3] : a
argv[4] : simple
argv[5] : demo
argv[6] : program
gemproject@ubuntu:~/Desktop$
```

Process Management Commands

- ps
  - Report a snapshot of the current processes.
  - ps u
    - List all of your process in difference login session.
    - Get process ID
- kill
  - Terminates or send signals to a process
  - When a process doesn’t go your way or out of control, use this command to terminate it .
  - kill <pid>

Run a Program in Linux

- Every command is an executable program
  - Ls -> /bin/ls
  - vim -> /usr/bin/vim
- Environment variables
  - env
    - PATH=/usr/local/bin:/usr/bin:/bin:/opt/bin
    - When user commit a command, if it wasn‘t a shell‘s built-in command, OS will try to put each environment variable in front of the command and find that program for execute.
- That means we can‘t just simply type a.out and hope to execute the program we compiled
  - You have to determine the file path of the output program.
    - ./a.out
    - ~/demo/a.out

File I/O

- Write file

gemproject@ubuntu: ~/Desktop

```c
#include <stdio.h>
#include <stdlib.h>

int main(){
  char buffer[20];
  snprintf(buffer, 20, "Hello world!");

  FILE *fp = fopen("write.txt", "w");   //open a file, set mode as write

  if(fp == NULL){
    printf("The file cannot be opened.");
    exit(1);
  }else {
    fprintf(fp, "%s", buffer);          //write buffer to file
  }

  fclose(fp);     // close file

  return 0;
}
~
~
~
"write.c" 20L, 385C                                    1,1           All
```

File I/O

- Read file

gemproject@ubuntu: ~/Desktop

```c
#include <stdio.h>
#include <stdlib.h>

int main(){
  char buffer[20];
  snprintf(buffer, 20, "Hello world!");

  FILE *fp = fopen("write.txt", "r");   //open a file, set mode as read

  if(fp == NULL){
    printf("The file cannot be opened.");
    exit(1);
  }else {
    printf("Read from ""write.txt"" :\n");

    while(fscanf(fp, "%s", buffer) == 1)     //read string from file
      printf("%s ", buffer);

    printf("\n");
  }

  fclose(fp);     // close file

  return 0;
}
```

Outline

- Introduction of virtual machine and installation tutorial
- How to connect to a remote machine ?
- Basic Linux command
- Basic Linux programming
- Networking tools on Linux

Network Related Commands

- ifconfig
  - Show current system NIC
    - eth* as intel Ethernet card
  - Placed in /sbin/ifconfig
- ping <host IP>
  - Send ICMP packet to host
  - Simply test if a server is alive
    - Not working if firewall is set.
      - Windows 7 default
- telnet <host IP> <host Port>
  - User interface to the TELNET protocol
    - A simple but solid text-based network client
    - Default port is 23

Capture and Analyze Packets

- Command line interface:
  - tcpdump

- Graphical User Interface:
  - Wireshark

What is tcpdump?

- Packet analyzer runs under the command line.
- Works on most unix-like OS.
- Support common protocols not just TCP.

How to Install tcpdump

- Ubuntu

  $ sudo apt-get update

  $ sudo apt-get install tcpdump

tcpdump

- tcpdump [-AbdDefhHIJKlLnNOpqStuUvxX#]
  [-i 網路介面] [-c 數量] [-w 檔案名(*.pcap)] [-r 檔案名(*.pcap)]
  - -A : Print each packet in ASCII. Handy for capturing web pages
  - -e : Print the link-level header (MAC) on each dump line.
  - -n : Print data with IP and port number instead of host name
  - -X : print the data of each packet in hex and ASCII
  - -i : Listen on interface, e.g.: eth0, lo, ppp0
  - -c : Exit after receiving count packets, if without the parameter, tcpdump will continue capture packets until press [ctrl]-c
  - -w : Write the raw packets to file rather than parsing and printing them out.
  - -r : Read packets from file (which was created with the –w that write pcap files)

Networking tools on Linux - tcpdump

$ sudo tcpdump –i <interface>

```
gemproject@ubuntu: ~/Desktop
gemproject@ubuntu:~/Desktop$ sudo tcpdump -i ens33
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
02:37:16.984801 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:16.986548 IP 192.168.47.128.35585 > 192.168.47.2.domain: 37155+ PTR? 255.47.168.192.in-addr.arpa. (45)
02:37:16.992068 IP 192.168.47.2.domain > 192.168.47.128.35585: 37155 NXDomain 0/0/0 (45)
02:37:16.993719 IP 192.168.47.128.35585 > 192.168.47.2.domain: 10604+ PTR? 1.47.168.192.in-addr.arpa. (43)
02:37:16.997292 IP 192.168.47.2.domain > 192.168.47.128.35585: 10604 NXDomain 0/0/0 (43)
02:37:17.000107 IP 192.168.47.128.35585 > 192.168.47.2.domain: 8791+ PTR? 2.47.168.192.in-addr.arpa. (43)
02:37:17.003932 IP 192.168.47.2.domain > 192.168.47.128.35585: 8791 NXDomain 0/0/0 (43)
02:37:17.004769 IP 192.168.47.128.35585 > 192.168.47.2.domain: 10533+ PTR? 128.47.168.192.in-addr.arpa. (45)
02:37:18.007601 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:19.008920 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:20.004588 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:20.998749 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:21.996189 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:23.003143 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:23.425760 IP 192.168.47.128.39976 > 91.108.56.110.https: Flags [P.], seq 2834803280:2834803369, ack 324399646, win 64480, length 8
9
02:37:23.426648 IP 91.108.56.110.https > 192.168.47.128.39976: Flags [.], ack 89, win 64240, length 0
02:37:23.427385 IP 192.168.47.128.35585 > 192.168.47.2.domain: 31279+ PTR? 110.56.108.91.in-addr.arpa. (44)
02:37:23.432877 IP 192.168.47.2.domain > 192.168.47.128.35585: 31279 NXDomain 0/0/0 (44)
02:37:23.509553 IP 91.108.56.110.https > 192.168.47.128.39976: Flags [P.], seq 1:90, ack 89, win 64240, length 89
02:37:23.509648 IP 192.168.47.128.39976 > 91.108.56.110.https: Flags [.], ack 90, win 64480, length 0
02:37:24.005875 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:25.006263 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
02:37:26.000085 IP 192.168.47.1.54915 > 192.168.47.255.54915: UDP, length 263
```

What is Wireshark?

“

Wireshark is the world's foremost network protocol analyzer. It lets you see what's happening on your network at a microscopic level. It is the de facto (and often de jure) standard across many industries and educational institutions.

”

How to Install Wireshark

- Ubuntu

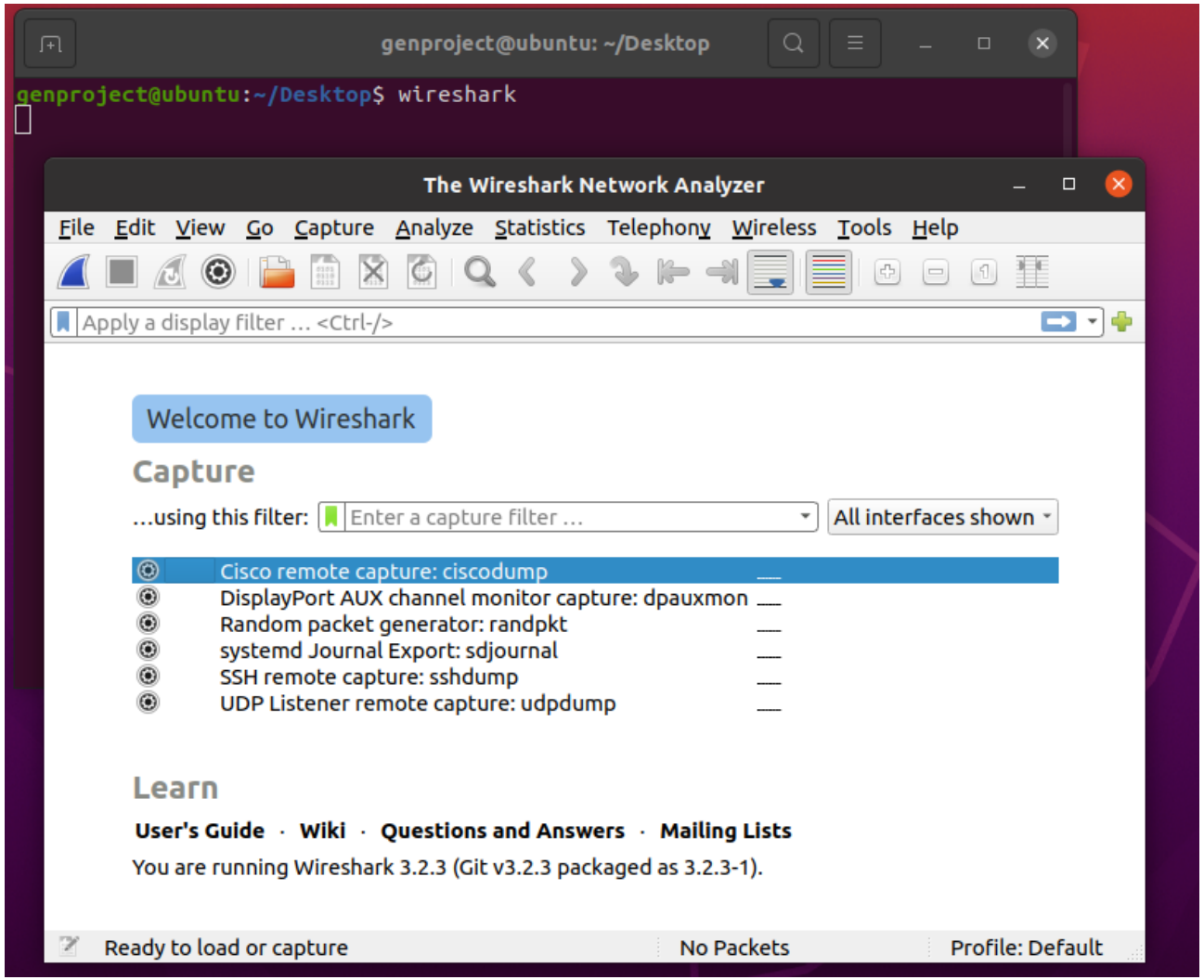
  $ sudo apt-get update

  $ sudo apt-get install wireshark

Start Wireshark (I)

$ wireshark

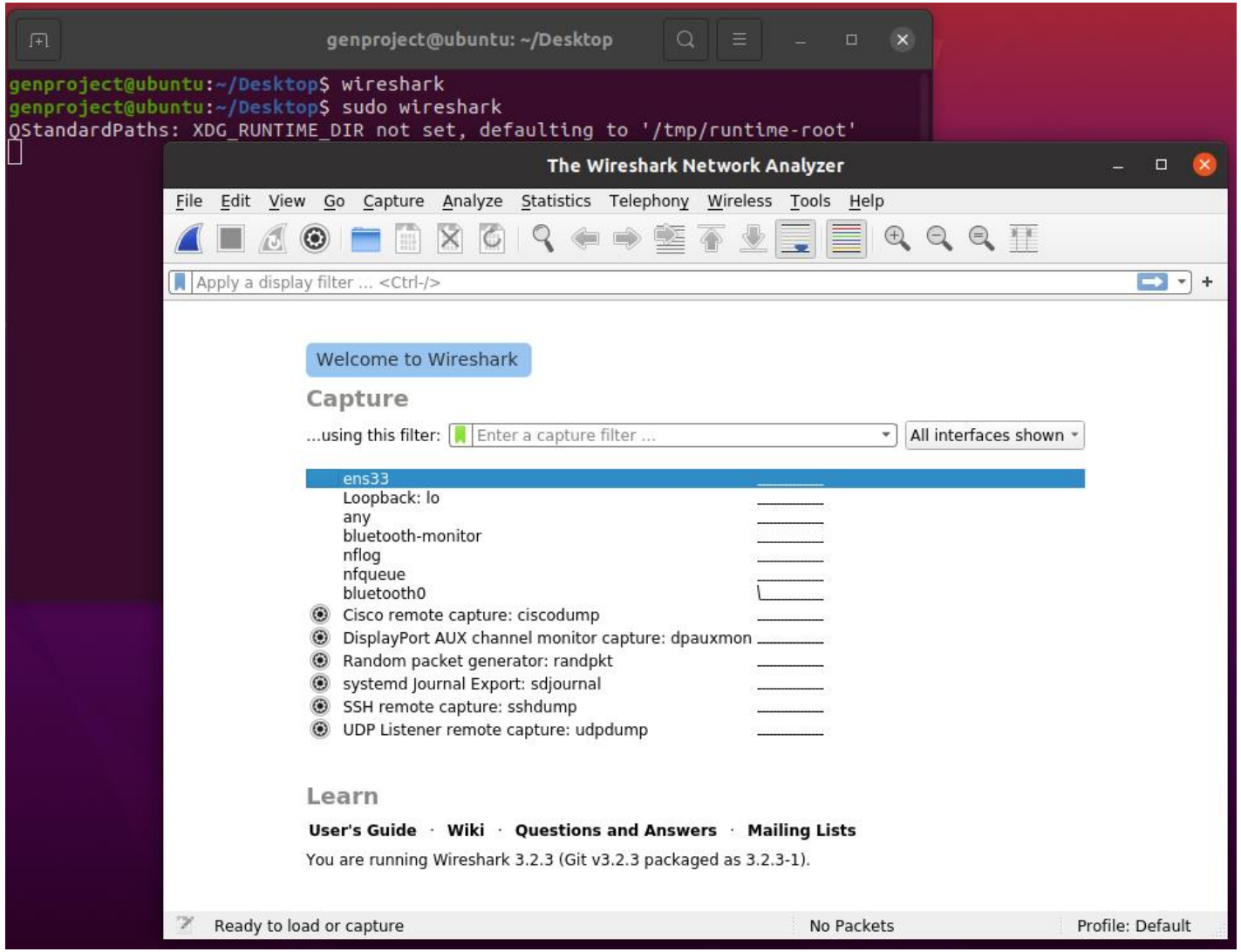
can’t find any interface!!!



Start Wireshark (II)

$ sudo wireshark

Run as super user



Start Wireshark (III)

- Solution

  $ sudo chgrp YOUR_USERNAME /usr/bin/dumpcap

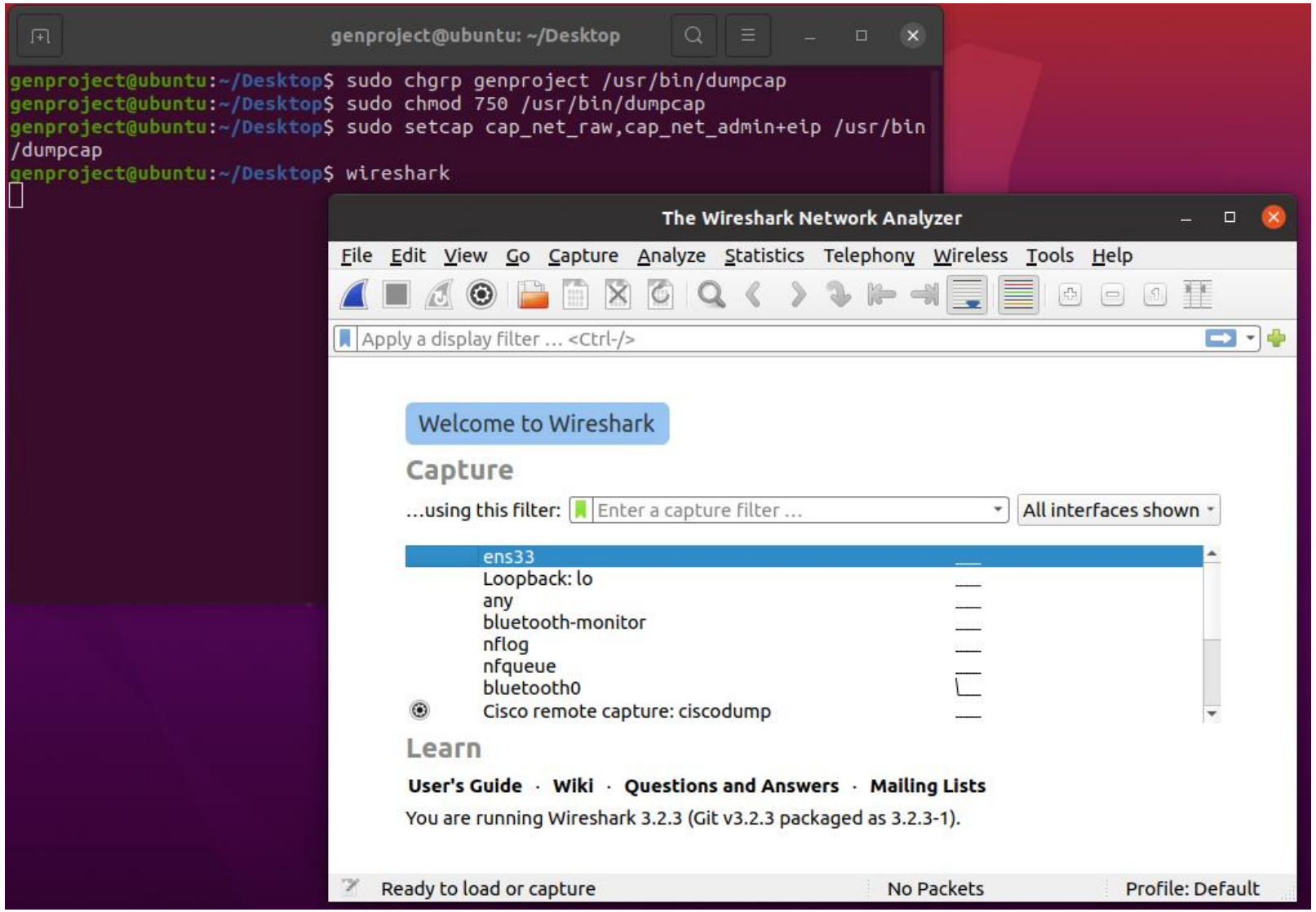
  $ sudo chmod 750 /usr/bin/dumpcap

  $ sudo setcap cap_net_raw,cap_net_admin+eip /usr/bin/dumpcap

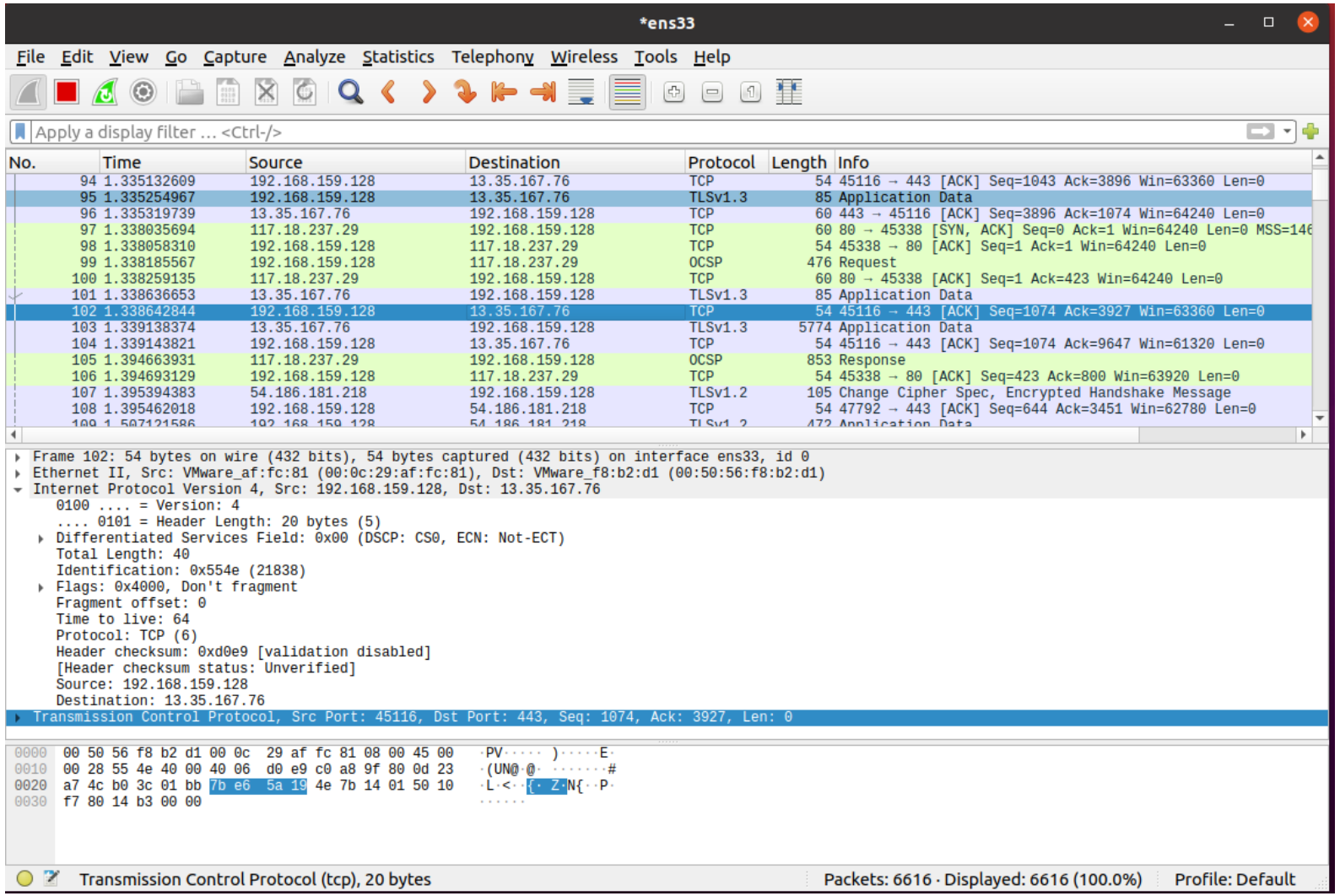
- Why
  - https://blog.wireshark.org/2010/02/running-wireshark-as-you/

Start Wireshark (IV)

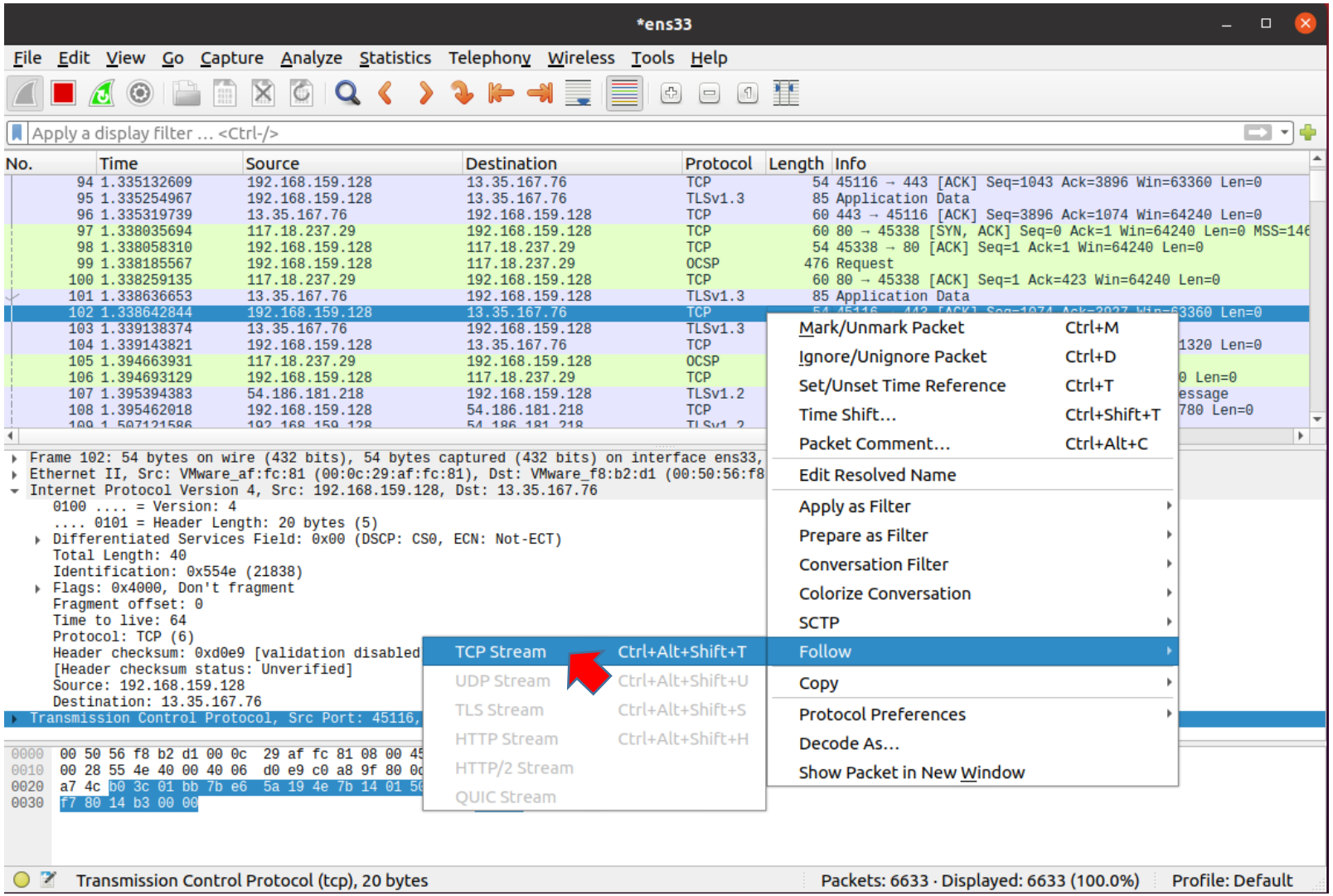
$ wireshark



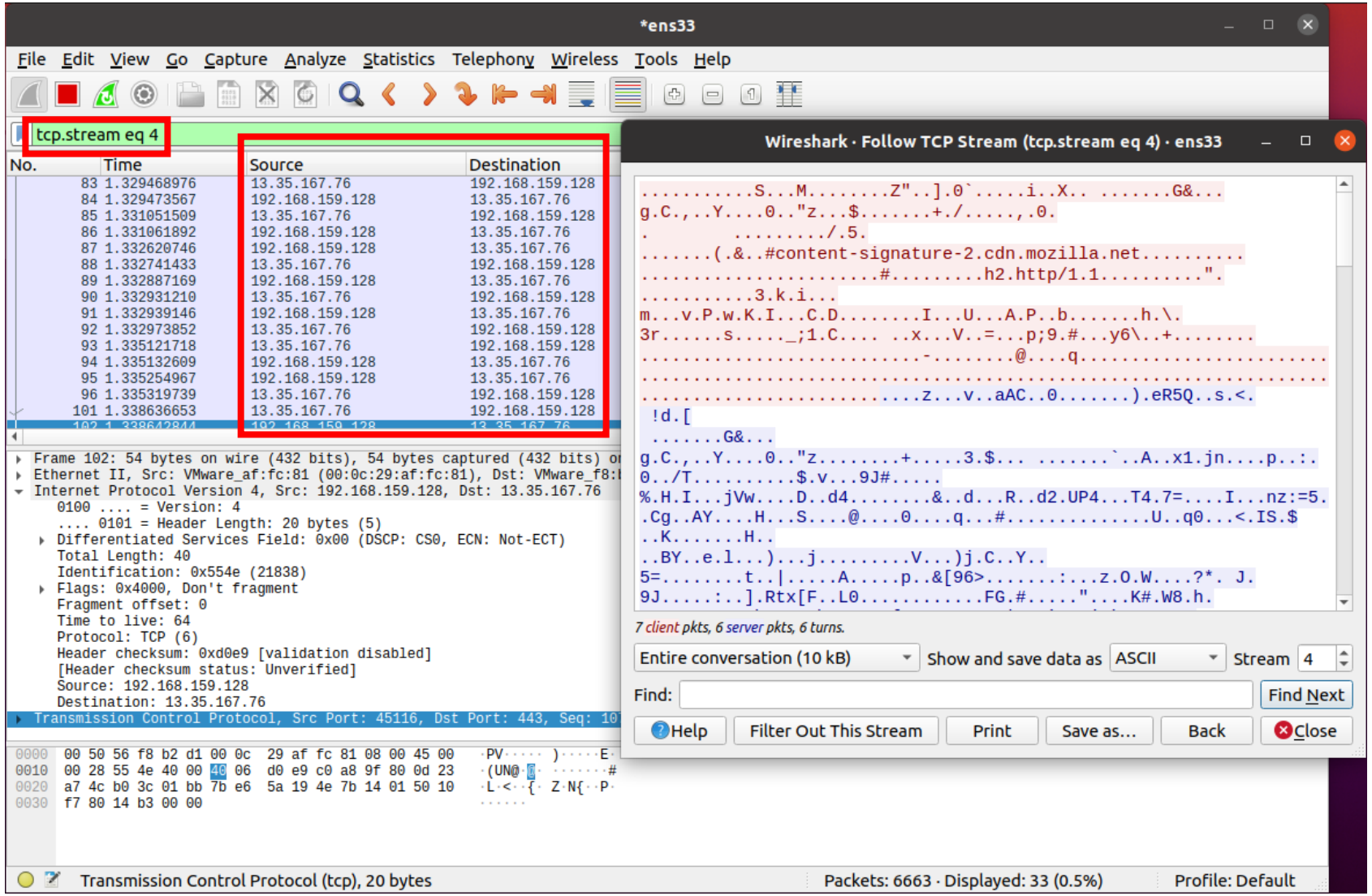
Basic Functions (I)



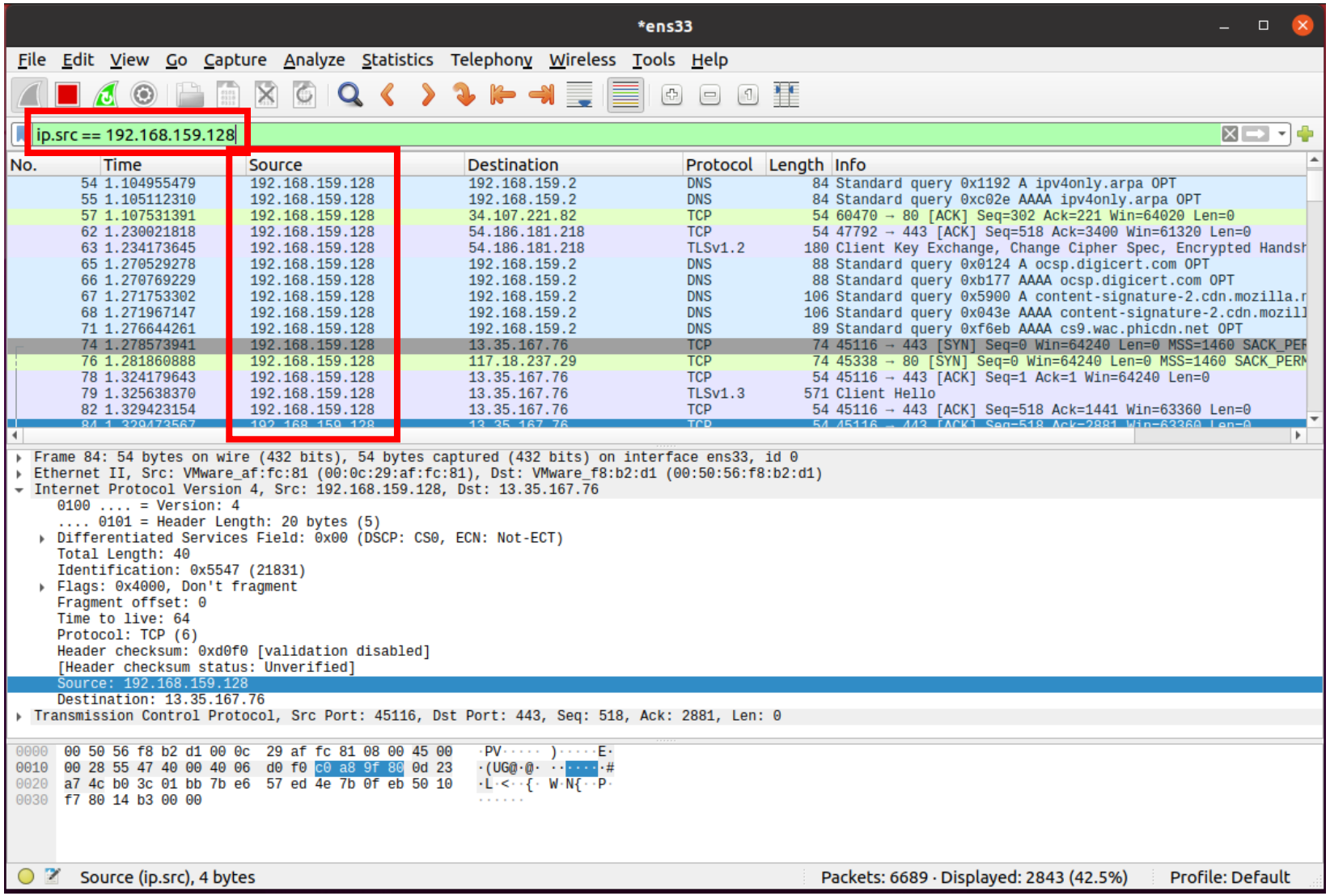
Basic Functions (II)



Basic Functions (III)



Filtering (I)


*ens33
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.src == 192.168.159.128
No. Time Source Destination Protocol Length Info
54 1.104955479 192.168.159.128 192.168.159.2 DNS 84 Standard query 0x1192 A ipv4only.arpa OPT
55 1.105112310 192.168.159.128 192.168.159.2 DNS 84 Standard query 0xc02e AAAA ipv4only.arpa OPT
57 1.107531391 192.168.159.128 34.107.221.82 TCP 54 60470 → 80 [ACK] Seq=302 Ack=221 Win=64020 Len=0
62 1.230021818 192.168.159.128 54.186.181.218 TCP 54 47792 → 443 [ACK] Seq=518 Ack=3400 Win=61320 Len=0
63 1.234173645 192.168.159.128 54.186.181.218 TLSv1.2 180 Client Key Exchange, Change Cipher Spec, Encrypted Handsh
65 1.270529278 192.168.159.128 192.168.159.2 DNS 88 Standard query 0x0124 A ocsp.digicert.com OPT
66 1.270769229 192.168.159.128 192.168.159.2 DNS 88 Standard query 0xb177 AAAA ocsp.digicert.com OPT
67 1.271753302 192.168.159.128 192.168.159.2 DNS 106 Standard query 0x5900 A content-signature-2.cdn.mozilla.r
68 1.271967147 192.168.159.128 192.168.159.2 DNS 106 Standard query 0x043e AAAA content-signature-2.cdn.mozill
71 1.276644261 192.168.159.128 192.168.159.2 DNS 89 Standard query 0xf6eb AAAA cs9.wac.phicdn.net OPT
74 1.278573941 192.168.159.128 13.35.167.76 TCP 74 45116 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PER
76 1.281860888 192.168.159.128 117.18.237.29 TCP 74 45338 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
78 1.324179643 192.168.159.128 13.35.167.76 TCP 54 45116 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
79 1.325638370 192.168.159.128 13.35.167.76 TLSv1.3 571 Client Hello
82 1.329423154 192.168.159.128 13.35.167.76 TCP 54 45116 → 443 [ACK] Seq=518 Ack=1441 Win=63360 Len=0
Frame 84: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface ens33, id 0
Ethernet II, Src: VMware_af:fc:81 (00:0c:29:af:fc:81), Dst: VMware_f8:b2:d1 (00:50:56:f8:b2:d1)
Internet Protocol Version 4, Src: 192.168.159.128, Dst: 13.35.167.76
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 40
Identification: 0x5547 (21831)
Flags: 0x4000, Don't fragment
Fragment offset: 0
Time to live: 64
Protocol: TCP (6)
Header checksum: 0xd0f0 [validation disabled]
[Header checksum status: Unverified]
Source: 192.168.159.128
Destination: 13.35.167.76
Transmission Control Protocol, Src Port: 45116, Dst Port: 443, Seq: 518, Ack: 2881, Len: 0
0000 00 50 56 f8 b2 d1 00 0c 29 af fc 81 08 00 45 00
0010 00 28 55 47 40 00 40 06 d0 f0 c0 a8 9f 80 0d 23
0020 a7 4c b0 3c 01 bb 7b e6 57 ed 4e 7b 0f eb 50 10
0030 f7 80 14 b3 00 00
Source (ip.src), 4 bytes
Packets: 6689 · Displayed: 2843 (42.5%)
Profile: Default

Filtering (II)

- Common expression
  - ip.src / ip.dst / ip.addr
  - tcp / udp / arp
  - tcp.port / udp.port

Homework

String processing

Overview

- Warm-up mini program (10%)
- Due date **10/14 23:59**
  - if you can’t submit it on time, your score will *0.75 every additional day.
- In this mini program, you will write a program (C or C++) with three commands:
  1. Reverse the string that you type.
  2. Split the string with specific character.
  3. Terminate itself by the command.
- You have to submit your code to new **e3** system, compress your code and Makefile with **zip** file, and renamed as your **student ID** (e.g. 0616057.zip). https://e3.nycu.edu.tw/
- If you have any questions, please email TAs.

String - strtok

char * strtok (char * str, const char * delimiters);

CODE

```
int main()
{
    char str[] = "I Love NP";
    char *pch = strtok(str, " ");
    while(pch != NULL)
    {
        printf("%s\n", pch);
        pch = strtok(NULL, " ");
    }
}
```

OUTPUT

```
13:28 changht@linux1 [~/np2013] >./strtok
I
Love
NP
13:28 changht@linux1 [~/np2013] >
```

String - strcmp

int strcmp (const char * str1, const char * str2);

CODE

```
int main()
{
    char str1[]="NP";
    char str2[]="haha";
    char str3[] = "NP";
    if(strcmp(str1,str2)==0)
        printf("str1 is the same as str2\n");
    else
        printf("str1 is different from str2\n");
    if(strcmp(str1,str3)==0)
        printf("str1 is the same as str3\n");
    else
        printf("str1 is different from str3\n");

    return 0;
}
```

OUTPUT

```
13:30 changht@linux1 [~/np2013] >./strcmp
str1 is different from str2
str1 is the same as str3
13:30 changht@linux1 [~/np2013] >
```

String - strcpy

char * strcpy (char * destination, const char * source);

CODE

```c
int main()
{
    char str1[]="I Love NP";
    char str2[20];
    strcpy(str2,str1);
    printf("str1: %s\nstr2: %s\n",str1,str2);
    return 0;
}
```

OUTPUT

```
13:31 changht@linux1 [~/np2013] >./strcpy
str1: I Love NP
str2: I Love NP
13:31 changht@linux1 [~/np2013] >
```

DEMO

```
reverse abcdefg
split abdecfg
~
~
~
~
```

c text file

```
ubuntu2@ubuntu:/mnt/hgfs/Ubuntu2/TA_course_WARM_UP$ ./Warmup_sample
usage: ./Warmup_sample [input file path] [split token]
ubuntu2@ubuntu:/mnt/hgfs/Ubuntu2/TA_course_WARM_UP$ ./Warmup_sample example.txt c
------------------------Input file example.txt------------------------
reverse abcdefg
gfedcba
split abdecfg
abde fg
------------------------End of input file example.txt------------------------
****************************User input****************************
reverse 11223344
44332211
split aabbcckk
aabb kk
exit
ubuntu2@ubuntu:/mnt/hgfs/Ubuntu2/TA_course_WARM_UP$
```

Q & A