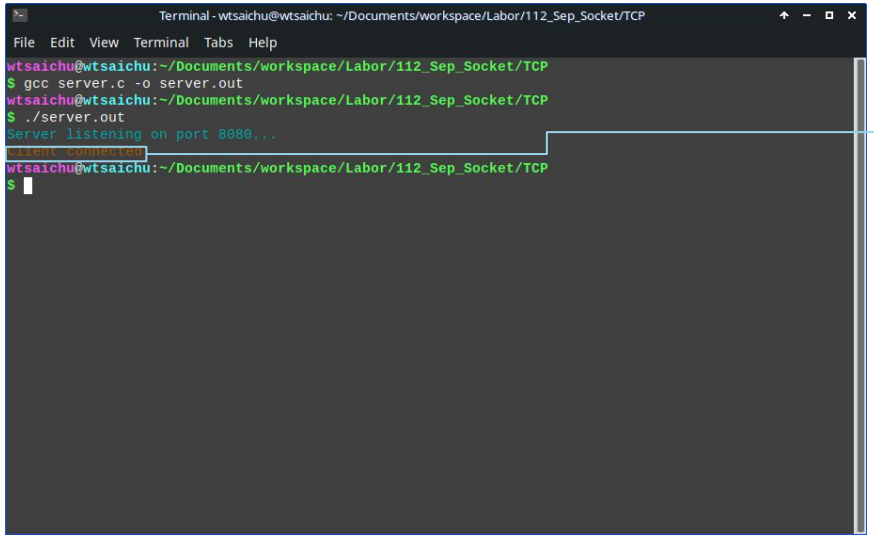
Socket

Date : 13/9/2023 9:30
Presenter : Chi-Chuan Ho

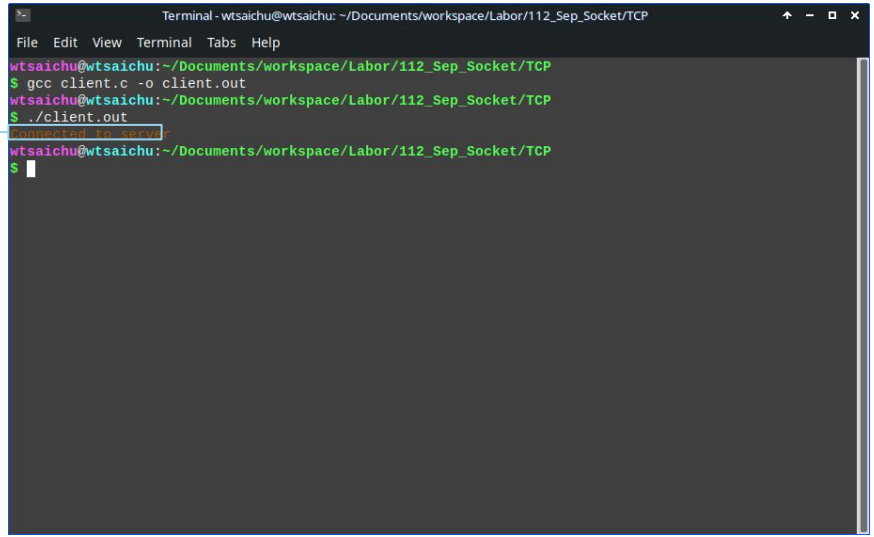
TCP Socket

Consequence

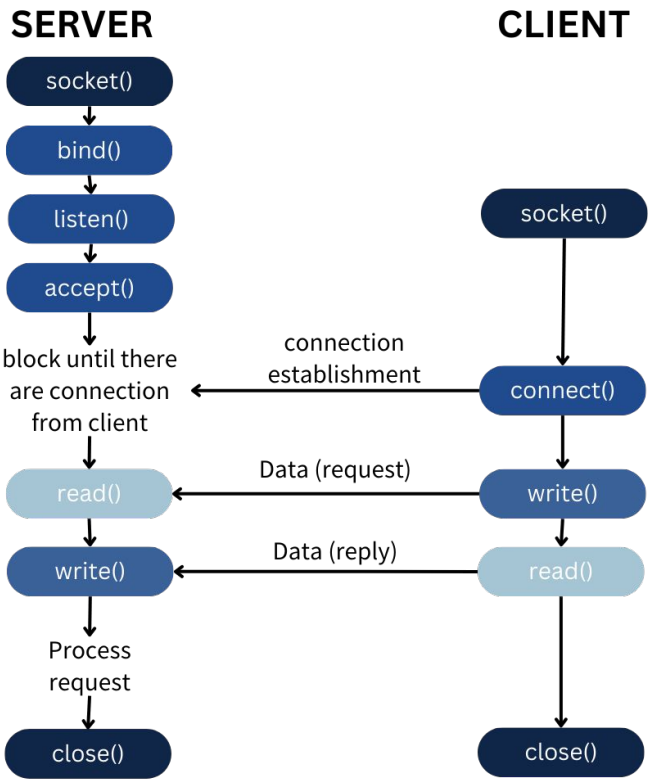
TCP Socket [Server]



TCP Socket [Client]



Frame


SERVER
CLIENT
socket()
bind()
listen()
accept()
block until there are connection from client
connection establishment
socket()
connect()
Data (request)
read()
write()
write()
Data (reply)
read()
Process request
close()
close()

Key code : Server

socket

```
server_socket = socket(AF_INET, SOCK_STREAM, 0);
```

domain : AF_INET, IPv4
type : **SOCK_DGRAM**
protocol : 0, for ALL

bind

```
bind(server_socket, (struct sockaddr*)&server_addr, sizeof(server_addr))
```

my_addr : 綁定的 IP 和 port

accept

```
accept(server_socket, (struct sockaddr*)&client_addr, &client_addr_len)
```

backlogaddr : client 的 IP 和 port
addrten : size of backlogaddr

Key code : Client

socket

```
client_socket = socket(AF_INET, SOCK_STREAM, 0);
```

domain : AF_INET, IPv4
type : **SOCK_STREAM**
protocol : 0, for ALL

connect

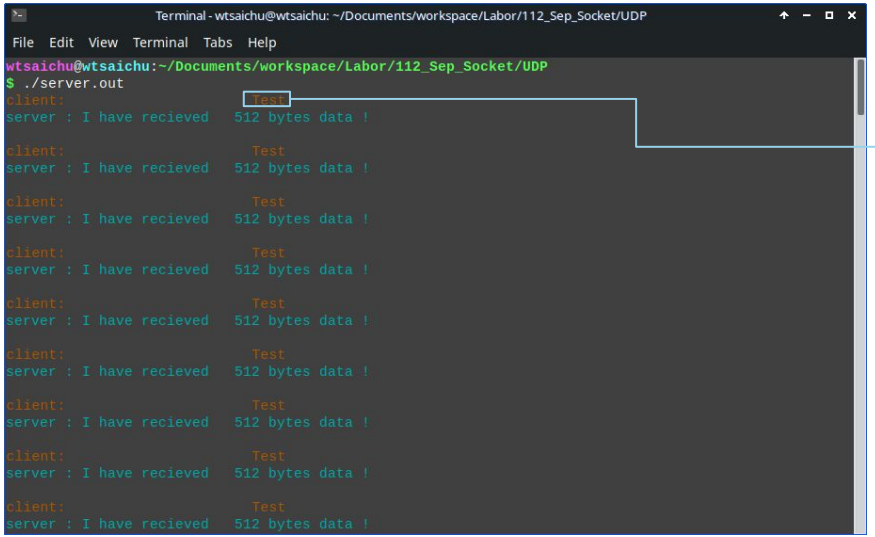
```
connect(client_socket, (struct sockaddr*)&server_addr, sizeof(server_addr))
```

serv_addr : server IP 和 port
addrlen : size of serv_addr

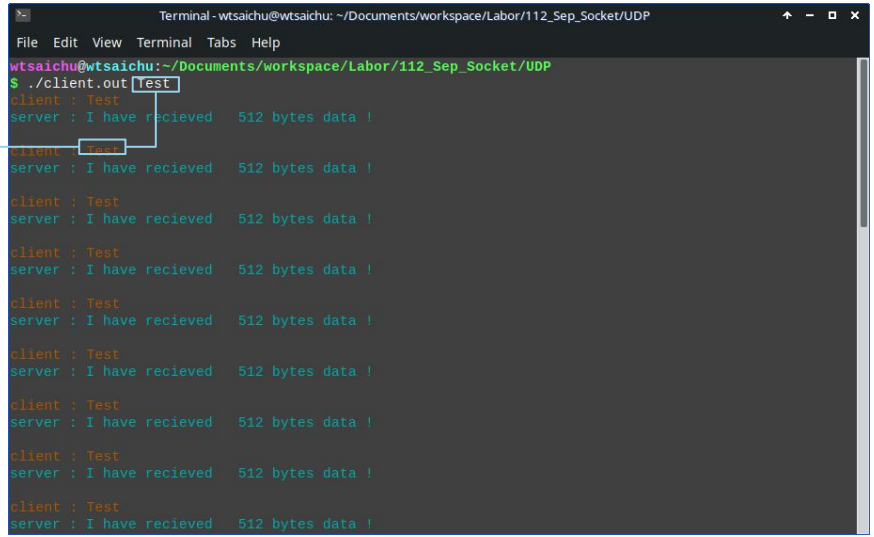
UDP Socket

Consequence

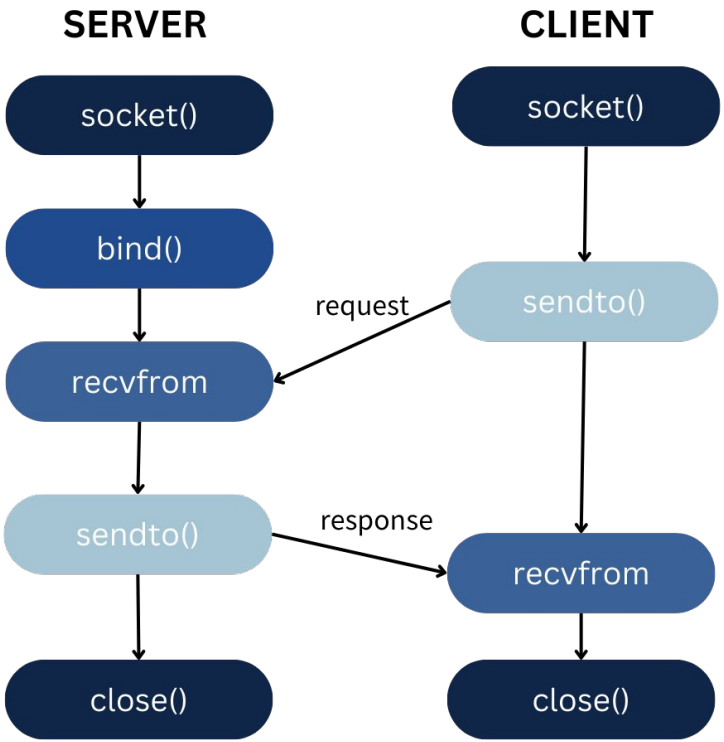
UDP Socket [Server]



UDP Socket [Client]



Frame


SERVER
CLIENT
socket()
socket()
bind()
request
sendto()
recvfrom
sendto()
response
recvfrom
close()
close()

Key code : Server

socket

```
server_dentifier = socket(AF_INET, SOCK_DGRAM, 0);
```

domain : AF_INET, IPv4
type : **SOCK_DGRAM**
protocol : 0, for ALL

bind

```
ret = bind(server_dentifier, (struct sockaddr*)&server_addr, sizeof(server_addr));
```

my_addr : 綁定的 IP 和 port

recvfrom

```
num = recvfrom(fd, buffer, BUFF_LEN, 0, (struct sockaddr*)&client_addr, &length);
```

buf : 已被 memset 清空的 1024 字元陣列
flags : 0
from : 來源主機的 IP 和 port

sendto

```
sendto(fd, buffer, BUFF_LEN, 0, (struct sockaddr*)&client_addr, length);
```

同上

Key code : Client

socket

```
client_dentifier = socket(AF_INET, SOCK_DGRAM, 0);
```

domain : AF_INET, IPv4
type : **SOCK_DGRAM**
protocol : 0, for ALL

sendto

```
sendto(fd, buffer, BUFF_LEN, 0, dst_address, length);
```

buf : 緩衝區 **len** : size of buf
flags : 0, 等價於 write()
to : 目地主機的IP 和 port **tolen** : size of to

recvfrom

```
recvfrom(fd, buffer, BUFF_LEN, 0, (struct sockaddr*)&src_address, &length);
```

buf : 已被 memset 清空的 1024 字元陣列 **len** : size of buf
flags : 0
to : 來源主機的IP 和 port **tolen** : size of tot

Features

不與通訊方**建立連接**，而是直接把要發的數據報發給通訊方

特性

1. 不在乎可靠性
2. 比使用 TCP 快很多
3. 適用於即時或是小數據傳輸

應用

1. SNMP
2. DNS

Raw Socket

Consequence

A ICMP echo request

```
Terminal - wtsaichu@wtsaichu: ~/Documents/workspace/Labtor/Socket/Raw
File  Edit  View  Terminal  Tabs  Help
wtsaichu@wtsaichu:~/Documents/workspace/Labtor/Socket/Raw
$ gcc ICMP.c -o ICMP.out
wtsaichu@wtsaichu:~/Documents/workspace/Labtor/Socket/Raw
$ sudo ./ICMP.out 8.8.8.8  -c 5
We have sended an ICMP packet to 8.8.8.8
The host 8.8.8.8 is alive!
The ICMP          Type is     0
The ICMP          Code is     0
The ICMP      Checksum is     0
The ICMP           TTL is    54
The ICMP  Header length is    5
We have sended an ICMP packet to 8.8.8.8
The host 8.8.8.8 is alive!
The ICMP          Type is     0
The ICMP          Code is     0
The ICMP      Checksum is     0
The ICMP           TTL is    54
The ICMP  Header length is    5
We have sended an ICMP packet to 8.8.8.8
The host 8.8.8.8 is alive!
The ICMP          Type is     0
The ICMP          Code is     0
The ICMP      Checksum is     0
The ICMP           TTL is    54
The ICMP  Header length is    5
We have sended an ICMP packet to 8.8.8.8
The host 8.8.8.8 is alive!
The ICMP          Type is     0
The ICMP          Code is     0
The ICMP      Checksum is     0
The ICMP           TTL is    54
The ICMP  Header length is    5
We have sended an ICMP packet to 8.8.8.8
The host 8.8.8.8 is alive!
The ICMP          Type is     0
The ICMP          Code is     0
The ICMP      Checksum is     0
The ICMP           TTL is    54
The ICMP  Header length is    5
wtsaichu@wtsaichu:~/Documents/workspace/Labtor/Socket/Raw
$
```

Field	Introduction
type	ICMP 消息類型
code	消息類型的代碼
Checksum	校驗和
ttl	Time to live
header length	

Key code

socket

```
socket_dentifier = socket(PF_INET, SOCK_RAW, IPPROTO_ICMP);
```

domain : PF_INET, IPv4
type : **SOCK_RAW**
protocol : ICMP

send

```
num = sendto(socket_dentifier, (char*)&hdr, sizeof(hdr), 0, (struct sockaddr*)&addr, sizeof(addr));
```

msg : hdr ,ICMP header **len** : size of hdr
flags : 0, 等價於 write()
to : destination IP address **tolen** : size of to

recieve

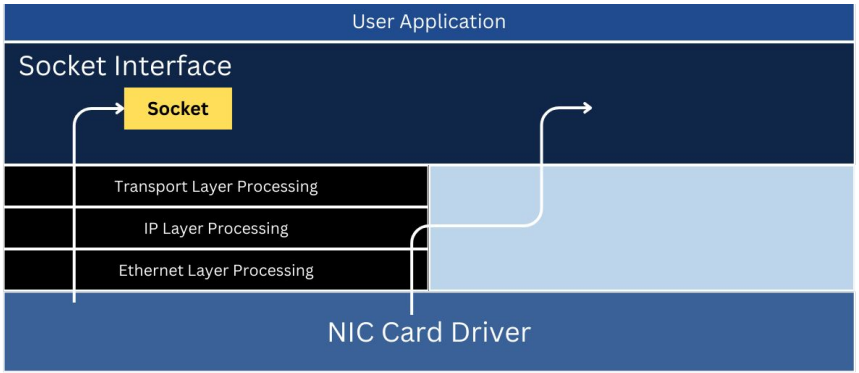
```
num = recv(socket_dentifier, buffer, sizeof(buffer), 0);
```

buf : 已被 memset 清空的 1024 字元陣列
len : size of buf
flags : 0, 等價於 open() 的 O_CLOEXEC

Features

允許**直接**傳送/接收IP協定封包而不需要任何傳輸層協定格式

TCP/UDP Socket : 只能處理數據載荷



Raw Socket : 可以處理 header