

分散式系統

Lab: Networking

請務必填寫學號系級姓名，以免成績登錄錯誤。

學號系級姓名:

112753207 資碩計一 張詠軒

請依問題與提示在指定區域回答問題，並依規定時間內上傳至 moodle。

第一部份 UDP

1. 請依據課程投影片中的範例，寫作符合下列要求的「UDP Client 與 Server」，其中，Server 請以 Node.js 寫作，Client 請以 Python 寫作。
2. Server 的需求:
 - 以 Node.js 寫作
 - 傾聽本地端(127.0.0.1)的 port 20213，接收 Client 傳來的訊息
 - 在 console 中印出所接收到的訊息
 - 將收到的訊息的最前面加上「XXXX:」再送回給 Client，其中，XXXX 是 Client 的 port number。(client 的 port number 請由 node.js 的 API 取得)
https://nodejs.org/docs/latest/api/dgram.html#dgram_event_message
提示: 使用 server.send 方法要嵌入在 on message 的 call back function 中，才會在收到訊息之後執行。
 - 印出後立即關閉連線(提示: server.close 方法要嵌入在 server.send 方法中)
3. Client 的需求:
 - 以 Python 寫作
 - 連接到本地端的 UDP Server，port 為 20213
 - 送出訊息: 「This is a test from python client」，記得要先將訊息轉為 binary: b"This is....."
 - 在 console 中印出所接收到的回應訊息 (記得用 message.decode('utf-8') 將 binary 轉回文字，正常的話應該會印出 xxxxx:This is a test from python client，而不是 b'xxxxx:This is a test form python client)
 - 收到後關閉連線

請將 Server 的程式碼則在下面:

答:

```

//使用 'dgram' 模組建立一個 UDP Socket
//使用 createSocket() 方法，並傳入 'udp4' 參數來建立一個 IPv4 的 Socket
const dgram = require('node:dgram');
const server = dgram.createSocket('udp4');

//處理錯誤
//印出錯誤訊息，然後關閉伺服器
server.on('error', (err) => {
    console.error(`server error:\n${err.stack}`);
    server.close();
});

//處理與回應訊息
//設定 message 事件的處理程序
//當收到客戶端發送的訊息時，會印出該訊息、客戶端的 IP 位址以及 port number
server.on('message', (msg, rinfo) => {
    console.log(`server got: ${msg} from
${rinfo.address}:${rinfo.port}`);

    //const responseMessage =
JSON.parse(`${rinfo.port}:${msg.toString}`);
    const responseMessage = Buffer.from(`${rinfo.port}:${msg}`);

    server.send(responseMessage, rinfo.port, rinfo.address, (err) => {
        server.close();
    });
});

server.on('listening', () => {
    const address = server.address();
    console.log(`server listening ${address.address}:${address.port}`);
});

server.bind(("127.0.0.1", 20213));

```

```

PS C:\Users\rick\Desktop\DS分散式系統\HW1> node Server.js
server listening 0.0.0.0:20213
server got: This is a test from python client from 127.0.0.1:51427
PS C:\Users\rick\Desktop\DS分散式系統\HW1>

```

請將 Client 的程式碼貼在下面:

答:

```
import socket

server_address = '127.0.0.1'
server_port = 20213
message = b"This is a test from python client"
sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

#UDP 是無連結協定，因此 connect()方法不是必須的，但可將 socket「連接」到特定的遠端位址，即可使用 send()和 recv()而不是 sendto()和 recvfrom()，使程式碼簡潔。
#並不意味著建立了一個真正的連接，因為 UDP 本質上仍然是無連接的。
#對於 TCP 是必要的，TCP 是面向連接通訊，要確保客戶端與伺服器的穩定連結，通過連結發送和接收數據。
sock.connect(("127.0.0.1", 20213))

#傳送資料
print(f'Client sending: {message.decode("utf-8")}')
sock.send(message)

#接收資料
#解碼與列印:接收到的資料被解碼為 UTF-8 字串並列印出來
#recv(bufsize)，bufsize:一次接收最大字數值，可自己設定
data = sock.recv(10240)
print(f'Client received: {data.decode("utf-8")}')

#關閉 socket
print('Client closing socket')
sock.close()
```

```
24 print('Client closing socket')
25 sock.close()
```

終端機 連接埠 問題 輸出 偵錯主控台

Python Debug Console

Windows PowerShell
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。

請嘗試新的跨平台 PowerShell <https://aka.ms/pscore6>

PS C:\Users\rick\Desktop\DS分散式系統\HW1> & 'c:\Users\rick\AppData\Local\Microsoft\WindowsApps\python3.12.exe' 'c:\Users\rick\.vscode\extensions\ms-python.debugpy-2024.2.0-win32-x64\bundled\libs\debugpy\adapter\..\..\debugpy\launcher' '56306' '--' 'c:\Users\rick\Desktop\DS分散式系統\HW1\Client.py'

Client sending: This is a test from python client
Client received: 51427:This is a test from python client
Client closing socket
PS C:\Users\rick\Desktop\DS分散式系統\HW1>

第二部份 群播

1. 執行 `multicastReceiver.js` 與 `multicastSender.js`，此時，`multicastSender` 會每 5 秒送一個封包到群播位址，並被 `multicastReceiver` 所接收

```
Windows PowerShell
Copyright (C) Microsoft Corporation. 著作權所有，並保留一切權利。
請嘗試新的跨平台 PowerShell https://aka.ms/pscore6

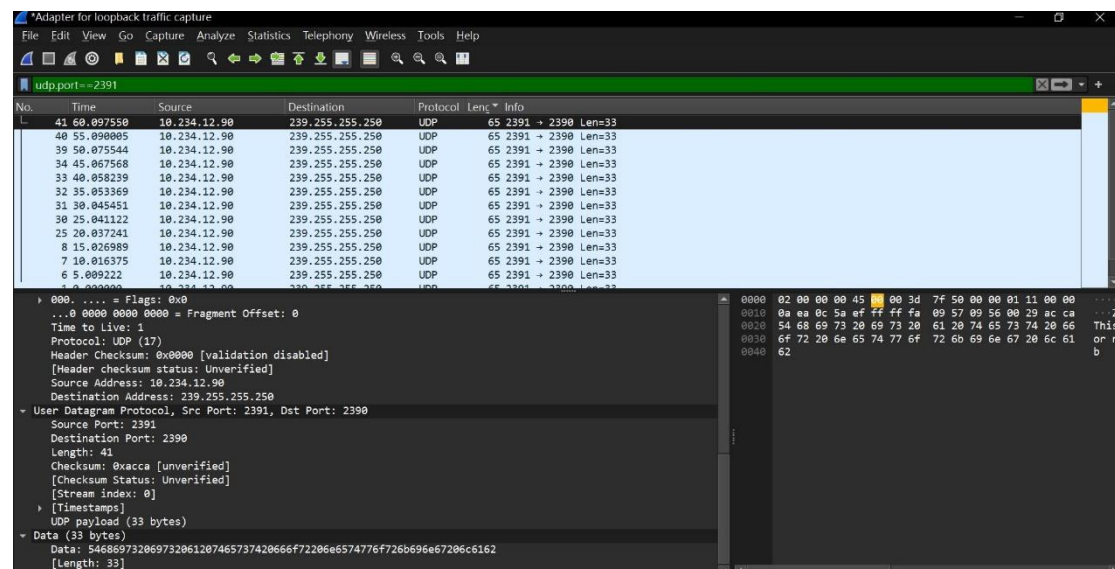
PS C:\Users\rick> cd C:\Users\rick\Desktop\DS\HW1
PS C:\Users\rick\Desktop\DS\HW1> node multicastReceiver.js
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
From: 10.234.12.90:2391 - This is a test for networking lab
```

2. 請問它送到那一個群播位址(不含 port)? 所傳送的内容為何?

答: **239.255.255.250, This is a test for networking lab**

3. 使用 Wireshark，設定 filter 為 `udp.port==2391`，抓取 `multicastSender` 送出的封包並觀察它 UDP 的 Length 欄位值為多少? UDP Payload 的大小(值)為多少?

答: **41 bytes, 33 bytes**



4. UDP Header 中的 Length 欄位代表什麼意思? 為什麼它會比 UDP Payload 的大小多 8 bytes?

答: **Length 欄位代表 UDP 封包的長度，包括 UDP header 和 UDP Payload 的長度。因為在計算 UDP 封包的長度時，需要把 UDP header 的長度也算進去，而 UDP header 的長度固定為 8 bytes，其中包含 Source port、Destination port、Length、Checksum 欄位，各佔 2 bytes。**