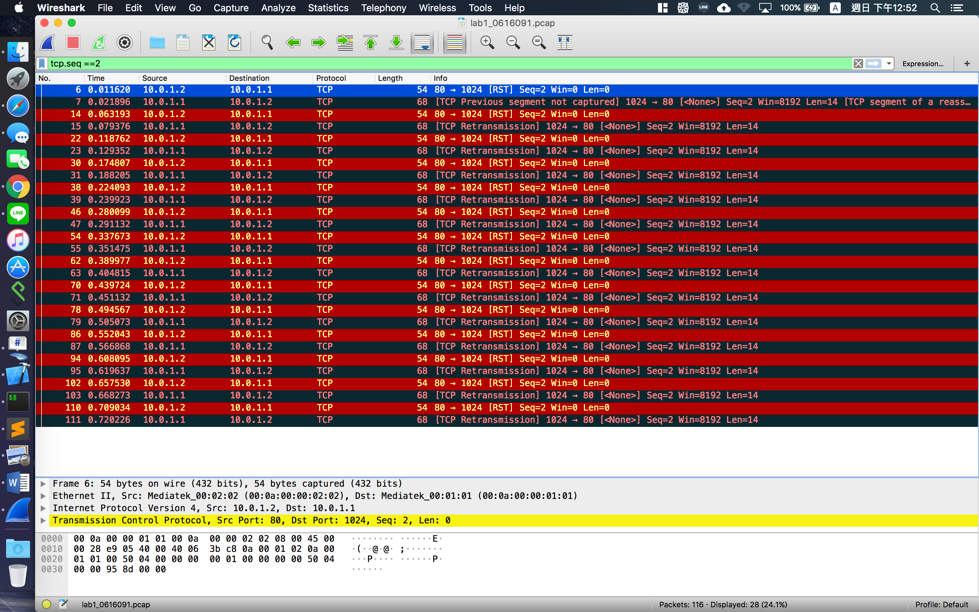
Student name: 陳昱螢Student ID: 0616091 Department: CS

Part A. Questions

1. What is your command to filter the packet with customized header on Wireshark?

Ans: Use “tcp.seq == 2” to filter the packet with customized header.

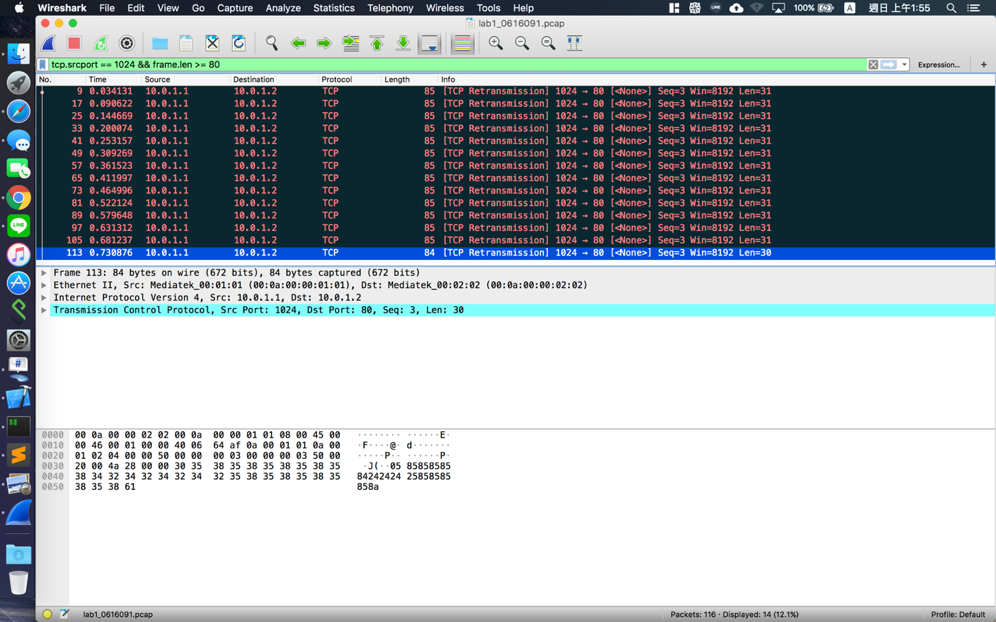
2. Show the screenshot of filtering the packet with customized header.



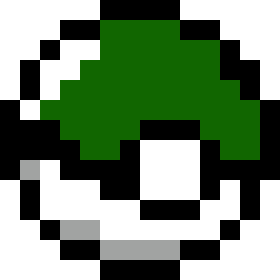
3. What is your command to filter the packet with “secret” payload on Wireshark?

Ans: Use “tcp.srcport == 1024 && frame.len >= 80“ to filter the packet.

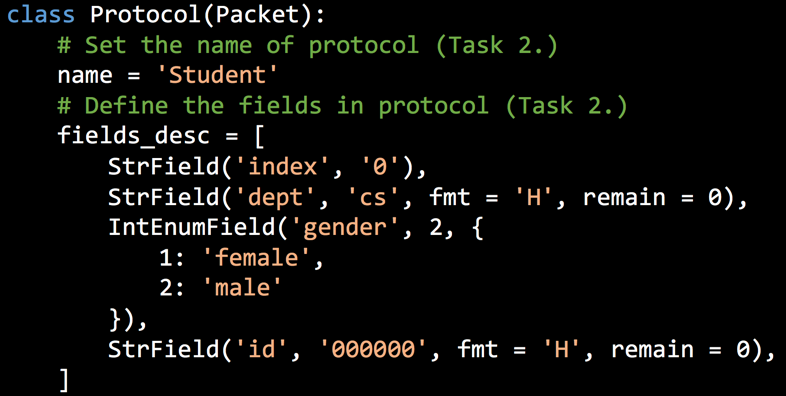
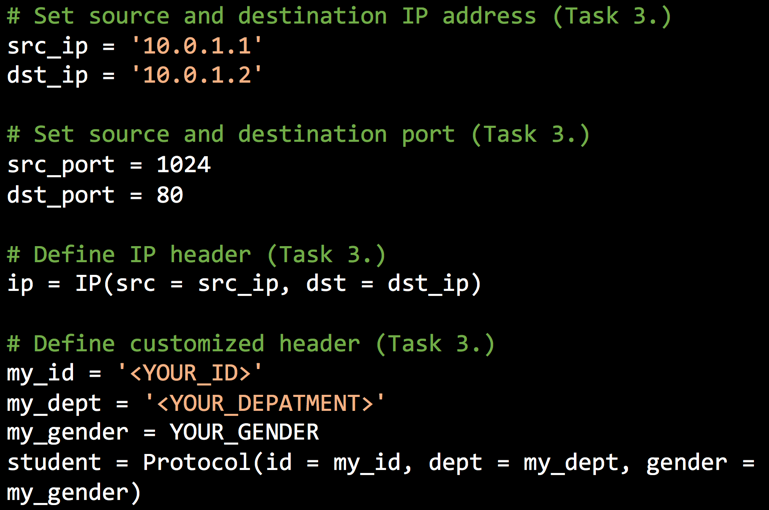
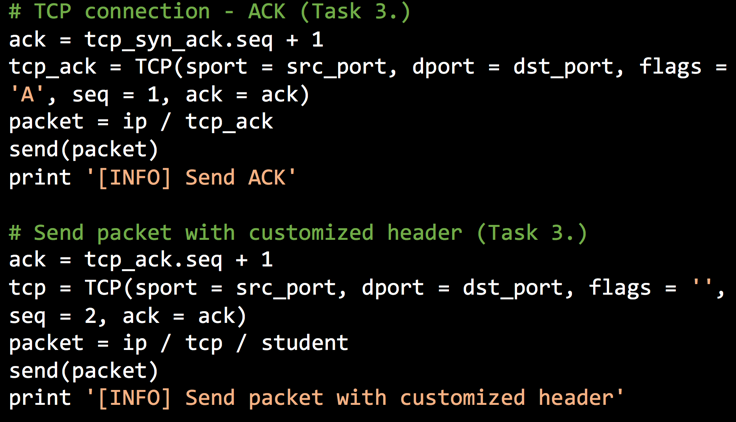
4. Show the screenshot of filtering the packet with “secret” payload .

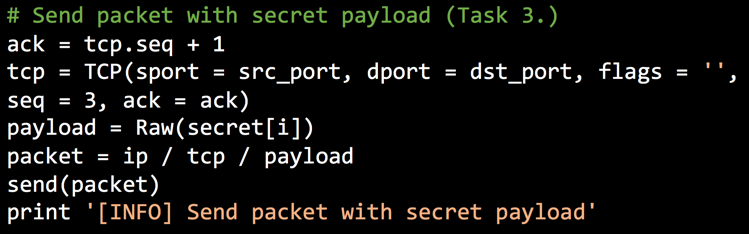


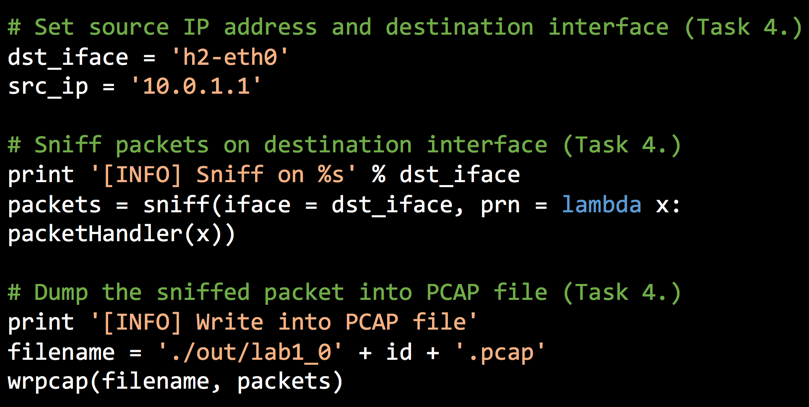
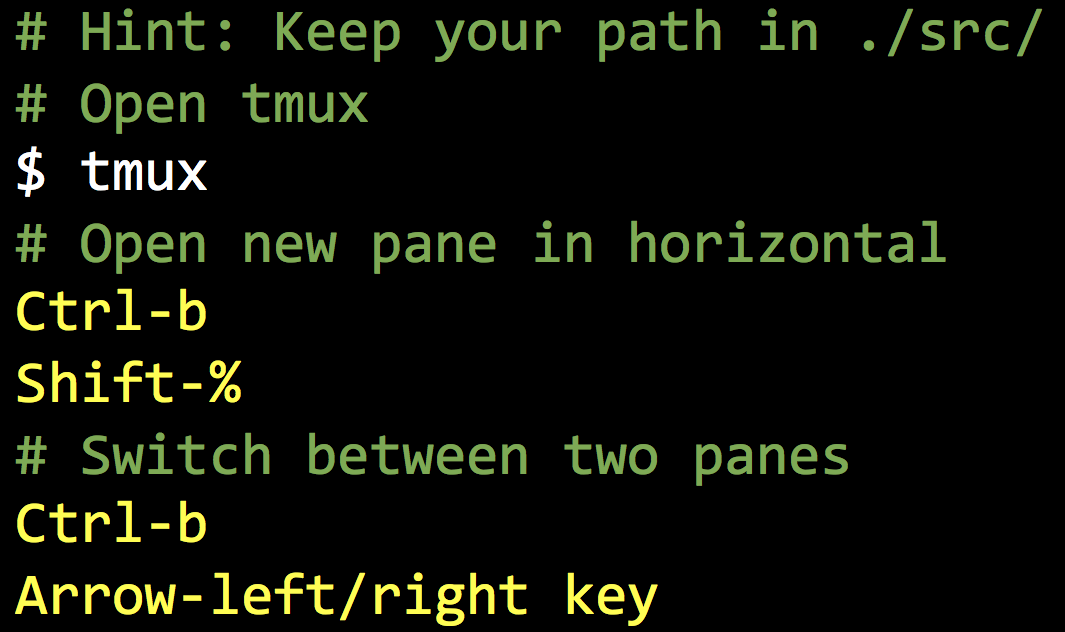
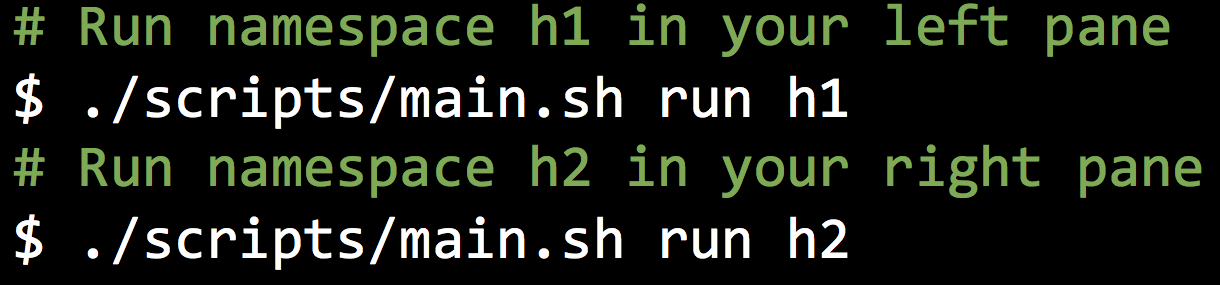
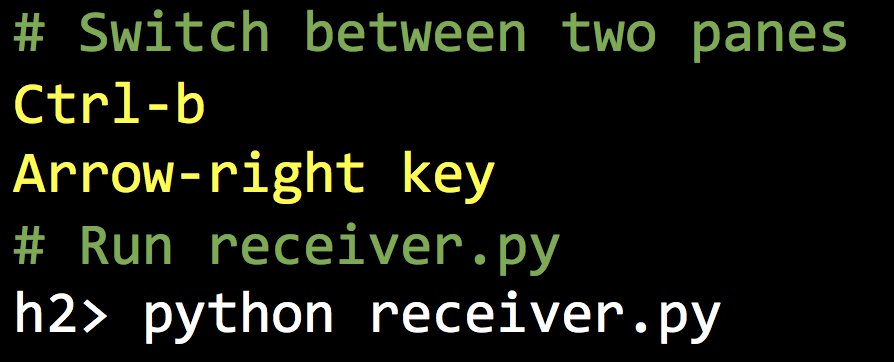
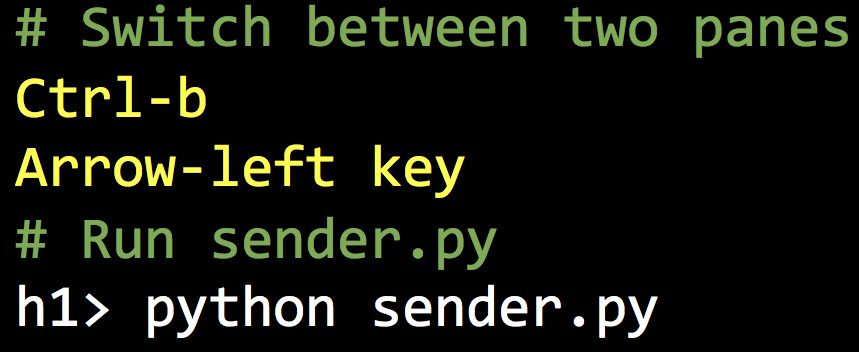
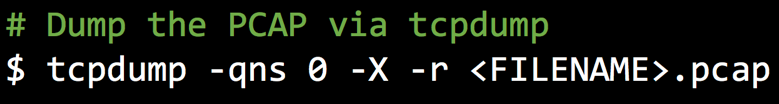
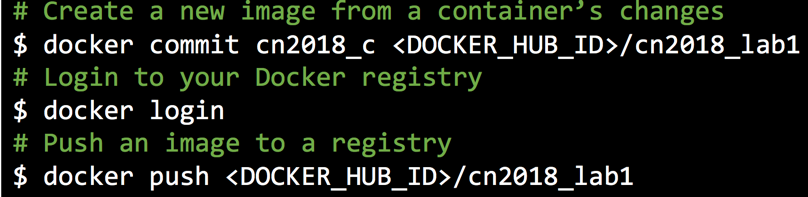
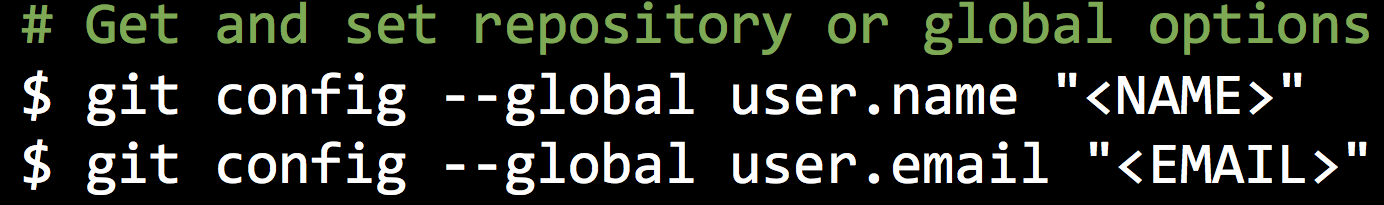
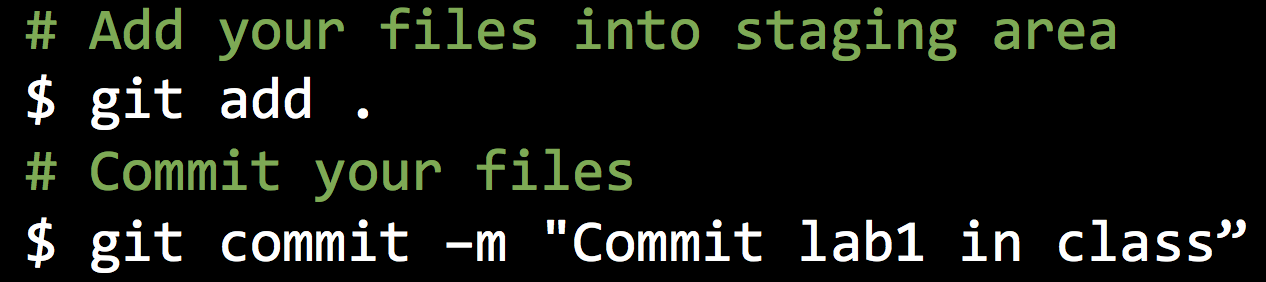
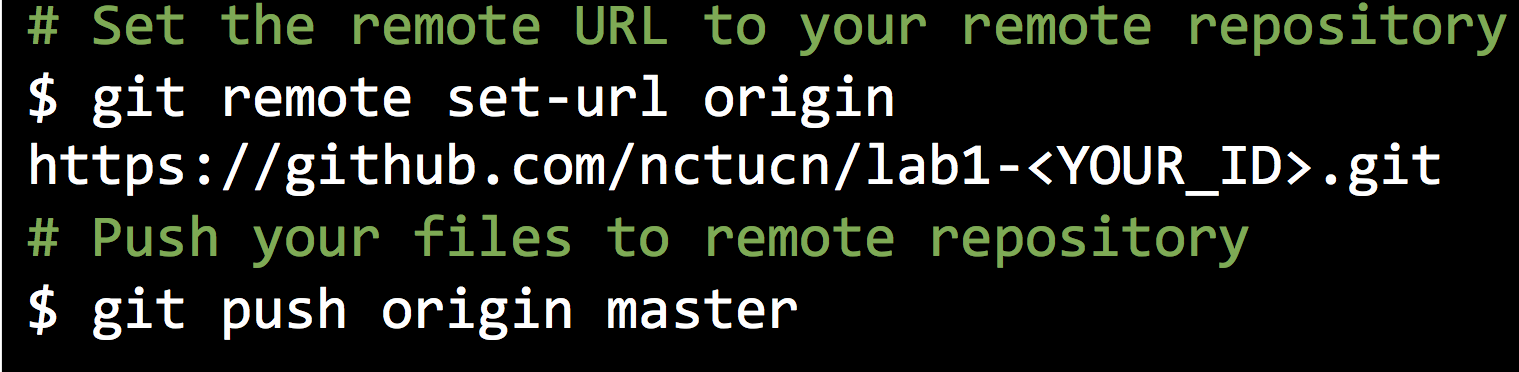
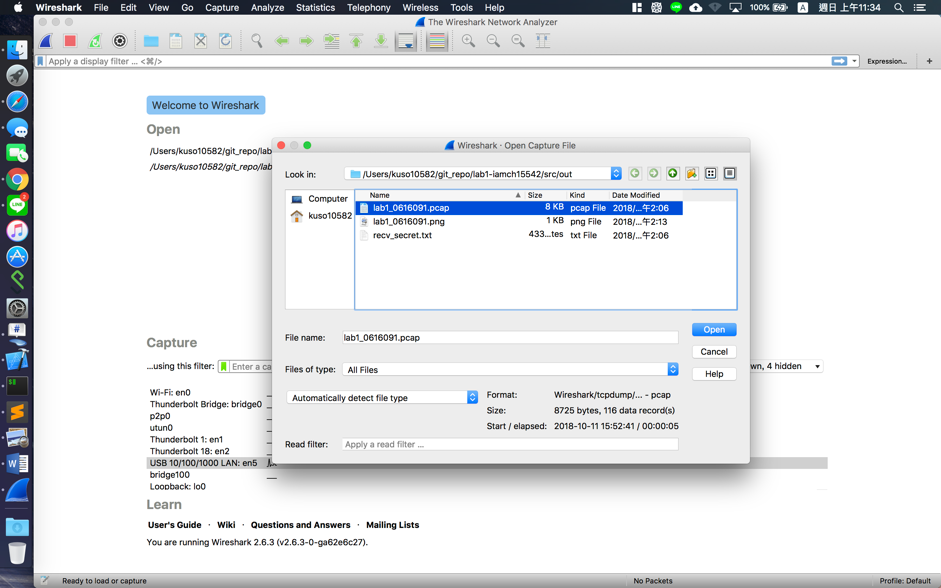
5. Show the result after decoding the “secret” payload.

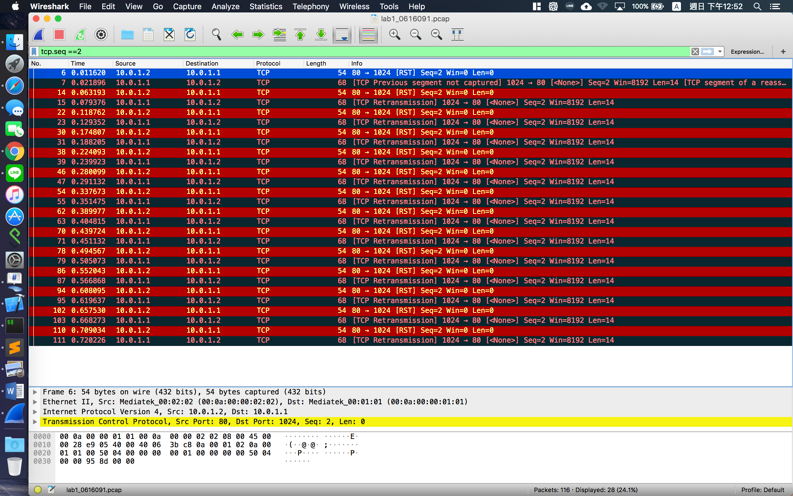
Ans:

Part B. Description

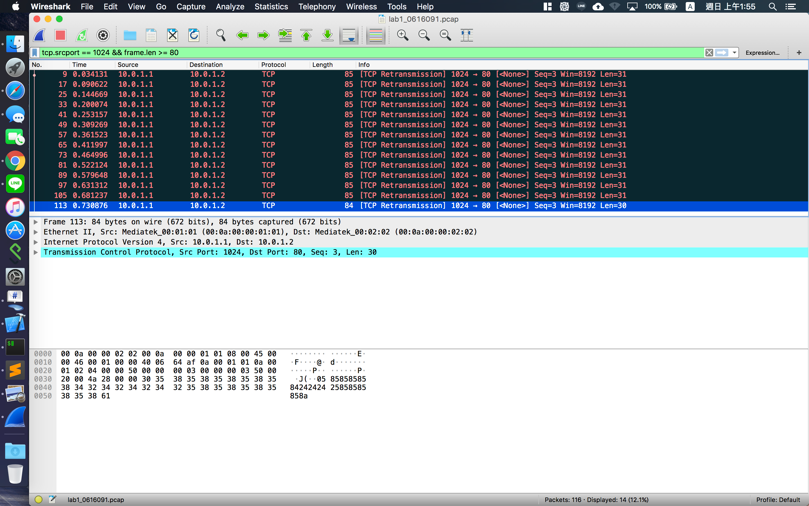
* Task 1 – Environment setup
* Task 2 – Define protocol via Scapy
  1. Define your protocol: ID header format
* 在Protocol.py裡新增下列的code，code的內容是在設定基本的資訊及定義。
* Task 3 – Send packets
  1. Set your own packet header
     + 在sender.py新增下列的code，第1跟第2句是在定義來源及目的地的ip位置，第3跟第4句則是定義來源跟目的地的port。第五句是在定義IP header的內容，剩下的五句則是定義自己的header的資訊。
  2. Send packets
     + 在sender.py新增下列的code，第一張圖片裡的code是在定義ACK的內容以及自己header的內容，然後傳遞出去。第二張圖片的code則是定義傳遞secret payload的內容。



* Task 4 – Sniff packets
  1. Receive and sniff packets
* 在receiver.py裡新增下列的code，前面兩行是先設置來源的ip位置跟目的地。再來則是在目的地的介面print”Sniff on 目的地”，最後先print出”write into PCAP file”後，便將所截取得到資訊寫入pcap檔案。
* Task 5 – Run sender and receiver
  1. Open tmux with horizontal two panes
* 在終端機使用tmux，再用下列的按鍵，將視窗分割成兩個視窗。
  1. Switch into two namespaces
* 將左邊的視窗開啟名為h1的namespace，再右邊的視窗開啟名為h2的namespace。
  1. Run receiver.py first
* 上面是切換左右視窗的指令。再來是先在h2開啟receiver.py，這樣才不會遺漏訊息。
  1. Run sender.py
* 上面是切換左右視窗的指令。再來是在h1開啟sender.py，開始傳遞訊息。
  1. Use tcpdump to show your PCAP file
* 使用tcpdump來解讀pcap的檔案，-r的意思是從指定的文件讀取數據包數據。
* Task 6 – Push your files to remote
  1. Push your image to Docker Hub
* docker commit 是根據你container的change創造新的image，而docker login 是在終端機登入你的docker帳號，docker push則是將你的image傳到Docker Hub
  1. Push your files to GitHub
* 設定git所需要的名稱及信箱。
* add是將資料夾裡的檔案讓git追蹤，commit 則是將暫存區的檔案存檔，後面的-m “Commit lab1 in class”，則是說明你在這次commit 做了什麼事。
* git remote set-url origin 是拿來設定遠端伺服器的網址，而git push origin master則是把master這個分支的內容推向origin的位置，若origin的遠端server並沒有master這個分支的話，便會建立一個叫做master的分支。
* Task 7 – Load PCAP via Wireshark
  1. Download your code from GitHub
* 在終端機輸入git clone <https://iamch15542@github.com/nctucn/lab1-iamch15542.git/>
  1. Install Wireshark 2.6.3
* 因為我是使用macos，所以我是到<https://www.wireshark.org/download.html>他的官網下載
  1. Open the PCAP file using Wireshark
* 直接選取資料夾裡面的pcap檔案就可以了
* 
* Task 8 – Filter the target packets
  1. Filter the packets of our defined protocol
     + Use “tcp.seq == 2” to filter the packets



* 1. Filter the packets with the “secret” bits
     + Use “tcp.seq == 3” or “tcp.srcport == 1024 && frame.len >= 80” to filter the packets



* Task 9 – Decode the secret key
  1. Input the secret key into ./src/decoder.py
* 在終端機執行decoder.py
* 輸入 python decoder.py 19061601906160
  1. Will have output in ./src/out/lab1\_0616091.png
* 圖片結果為