Network System Capstone – Final Project
Project Progress Report
Network monitoring and intrusion detection system

1. Introduction

本期末企劃靈感來源為HW1的封包監測系統，雖然現在市面上常用來監測網路封包的工具wireshark的功能完整，但是缺乏圖形化的資料呈現方式，無法很直觀、靈活展現整體封包的種類及分佈，此外，wireshark雖然提供很完整的封包監測功能，但它對使用者的專業性要求很高，不同使用者來分析同一份wireshark檔案可能會有不同的理解，有時候過於完整的資料反而導致使用者分析上的混亂，因此，本企劃將簡化wireshark的功能，提供最基本的網路監測的功能，這樣只需要基本功能的使用者可透過我們的程式來過濾對他們來說多餘的資訊，需要更詳細資料的專業使用者依舊可用wireshark來打開用我們程式抓取到的資料，另外我們也可產生圖表(ex. 網路頻寬使用率圖、封包種類統計圖.....) 來讓使用者能一目了然當前網路的狀況。

本企劃除了提供監測功能外，還加入了網路入侵偵測的功能，在資安的重要性日漸增加的背景下，我們的程式會在監測網路封包的同時，運用機器學習的模型來分析網路界面的封包，以檢查有沒有網路入侵的情況發生，如果有的話，會在程式中跳出警示供使用者及早發現並應對，降低資安疑慮。

2. Expected Functionality

本期末企劃預計會有以下幾點功能:

- 網路流量監視：系統將實時監視網路流量中的封包，將出入的封包記錄並統計信息，例如流量大小、流量來源和目的地、協議類型等。
- 網路性能分析：系統將分析網路性能指標，例如頻寬使用率、延遲時間、丟包率等，並可以生成報告或圖以便於分析和評估網路性能。
- 歷史紀錄：能夠將捕獲到的封包和相關分析結果保存，以便後續查詢和分析。
- 網路安全監視：能夠檢測網路中的異常流量、潛在風險和安全事件，並發出警報通知。

3. Details

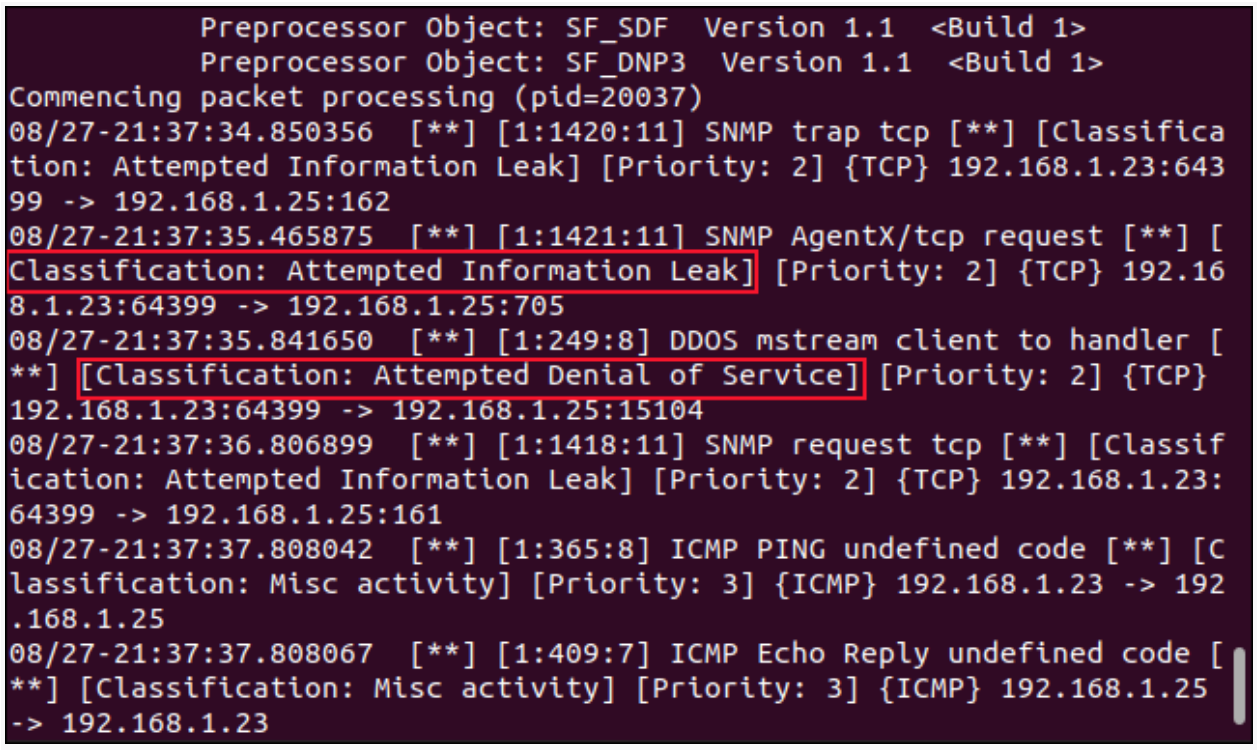
首先是網路監測的部份(Network Monitoring and Analyzing)，我會使用pyshark來捕獲經過網路的封包。程式將提供一個互動式的Command line界面，讓使用者設定捕獲時間、過濾條件等參數，以適應不同使用者的監控需求。在擷取到封包後，程式也會顯示最基本的封包資訊，例如來源IP Address、目的地IP Address、通訊協議等等.....，讓使用者對封包資料有初步認識。如果使用者想獲得圖形化的封包統計資訊，也可以對我們程式下指令來獲得相應的圖表，封包資訊的來源可以是我們剛抓取的封包資料，也可以是使用者在其他獲得的封包資料檔，只要檔案格式是pcap都可以產出統計圖表。

```
[>>>] Sniffer initialized. Waiting for incoming packets. Press Ctrl-C to abort...

[>] Packet #1 at 18:20:51:
    [+] MAC ......e4:6f:13:a2:44:1b -> 01:80:c2:00:00:00
[>] Packet #2 at 18:20:52:
    [+] MAC ......b0:fc:36:5e:8f:31 -> e4:6f:13:a2:44:1b
    [+] IPv4 ...........192.168.1.4 -> 51.132.193.105  | PROTO: TCP TTL: 64
    [+] TCP ..................57811 -> 443              | Flags: 0x011 > ACK FIN
[>] Packet #3 at 18:20:52:
    [+] MAC ......b0:fc:36:5e:8f:31 -> e4:6f:13:a2:44:1b
    [+] IPv4 ...........192.168.1.4 -> 20.189.173.4    | PROTO: TCP TTL: 64
    [+] TCP ..................42931 -> 443              | Flags: 0x002 > SYN
[>] Packet #4 at 18:20:52:
    [+] MAC ......00:00:00:00:00:00 -> 00:00:00:00:00:00
    [+] IPv4 .............127.0.0.1 -> 127.0.0.53       | PROTO: UDP TTL: 64
    [+] UDP ..................57928 -> 53
[>] Packet #5 at 18:20:52:
    [+] MAC ......00:00:00:00:00:00 -> 00:00:00:00:00:00
    [+] IPv4 .............127.0.0.1 -> 127.0.0.53       | PROTO: UDP TTL: 64
    [+] UDP ..................57928 -> 53
[>] Packet #6 at 18:20:52:
    [+] MAC ......00:00:00:00:00:00 -> 00:00:00:00:00:00
    [+] IPv4 .............127.0.0.1 -> 127.0.0.53       | PROTO: UDP TTL: 64
    [+] UDP ..................44488 -> 53
[>] Packet #7 at 18:20:52:
    [+] MAC ......00:00:00:00:00:00 -> 00:00:00:00:00:00
```

^ Network Monitoring 示意圖

另外是關於入侵檢測，初步的構想是會先找到網路上代表intrusion並有公信力的dataset，再用sklearn來train出好的model來供我們期末企劃的程式來判斷是否有入侵的依據，接下來根據使用者實時偵測的數據或是引入的pcap檔案來判斷是否有入侵的情況發生，有的話就在程式跳出警語提醒使用者有入侵的可能，及早讓使用者線來讓他做後續的處理。

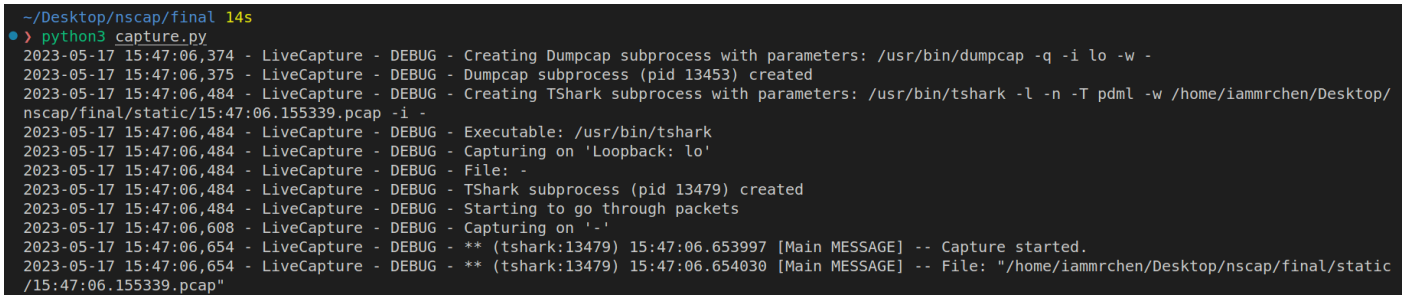


^ Intrusion Detection 示意圖

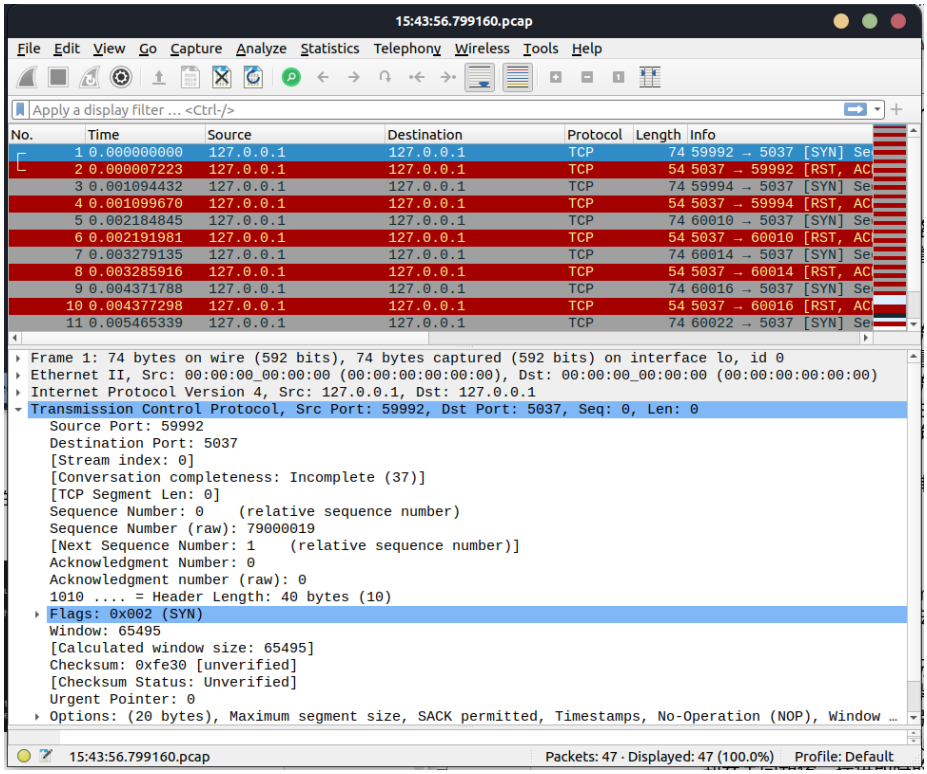
4. **Experimental/Development Environment**
   - 開發語言：Python3.10
   - 作業系統：Ubuntu 22.04
   - 網路封包抓取函式庫：Pyshark
   - 機器學習函式庫：Scikit-learn
   - 可視化函式庫：Matplotlib

5. **Progress So Far……**

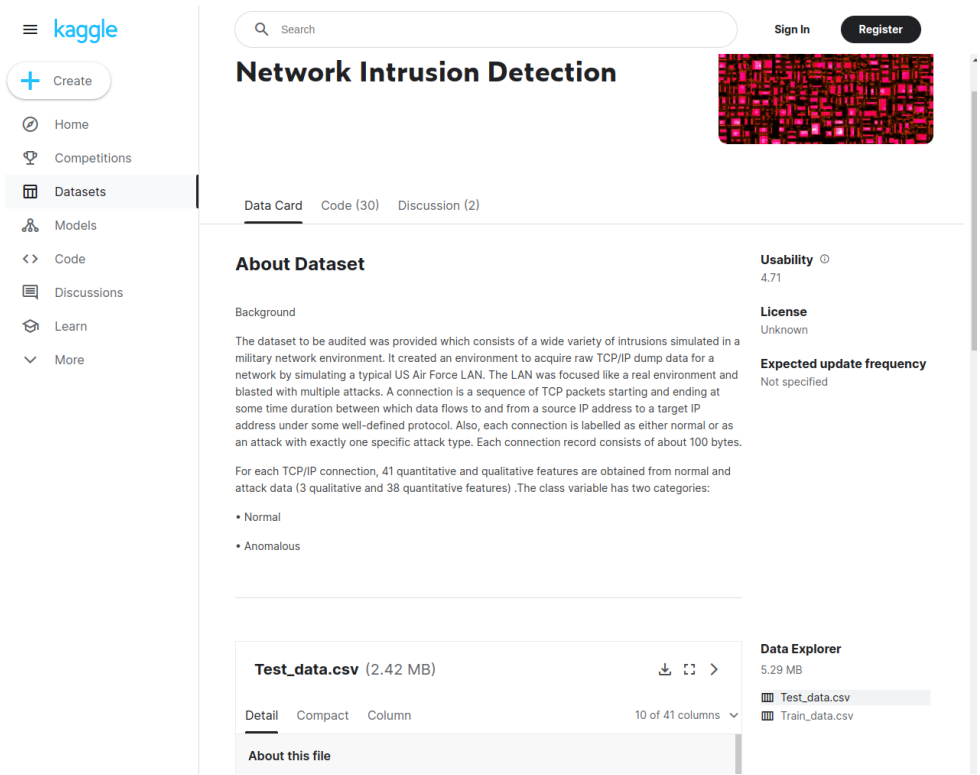
目前已經完成需要函式庫的安裝，並已經可以用Pyshark來監測選定的network interface上的封包(可自訂時間長短與監控哪個interface):



而監測的檔案已經確定可由wireshark來打開並查看封包的詳細資訊:



而關於Intrusion Detection的部份，目前已經找到kaggle的dataset來作為train model用:



目前正在透過網路上sklearn的教學來學習如何使用sklearn，預計應該是可以在期限內train出model供期末專題來做使用。