

HW1

Get SSL Key: <https://support.f5.com/csp/article/K50557518>

Export PNG file from Wireshark: <https://osqa-ask.wireshark.org/questions/35123/fastest-way-to-display-a-png-file/>

NA Question: Computer Network : A Top To Down Approach

NA

看個影集也會不小心洩漏密碼？！

<https://support.f5.com/csp/article/K50557518>

1.

```
Source: 192.168.88.194  
Destination: 140.140.112.30,159  
Transmission Control Protocol, Src Port: 56424, Dst Port: 80, Seq: 2388, Ack: 234287, Len: 809  
Source Port: 56424  
Destination Port: 80  
Stream Index: 165  
[TCP Segment Len: 809]  
Sequence number: 2388 (relative sequence number)  
Sequence number (raw): 4043084842  
Next sequence number: 3127 (relative sequence number)  
Acknowledgment number: 234287 (relative ack number)  
Acknowledgment number (raw): 1552918823  
1000 ... = Header Length: 32 bytes (8)  
Flags: 0x018 (PSH, ACK)  
Window size value: 2441  
(Calculated window size: 312448)  
Window size scaling factor: 128  
Checksum: 8x7c8 [unverified]  
(Checksum Status: Unverified)  
Urgent pointer: 0  
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps  
» [SEQ/ACK analysis]  
» [Timestamps]  
TCP payload (809 bytes)  
» Hypertext Transfer Protocol  
» POST /netfilz/login.html HTTP/1.1\r\n  
» [Expert Info (Chat/Sequence): POST /netfilz/login.html HTTP/1.1\r\n]Request Method: POST  
» Request URI: /netfilz/login.html  
» Request Version: HTTP/1.1  
Host: nasahwi.csie.ntu.edu.tw/\r\nUser-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:97.0) Gecko/20100101 Firefox/97.0/\r\nAccept: */*\r\nAccept-Language: en-US,en;q=0.5\r\nAccept-Encoding: gzip, deflate\r\nContent-Type: application/x-www-form-urlencoded\r\nContent-Length: 38\r\nOrigin: http://nasahwi.csie.ntu.edu.tw/\r\nConnection: keep-alive\r\nReferer: http://nasahwi.csie.ntu.edu.tw/netfilz/\r\n» [truncatedCookie: ga=GA1.3.1673844394.1645467841; utma=1543809519.1673844394.1645467841.1645546077.1; utmz=1543809519.1645546077.1.1.utmcsrc=google|utmccn=(organic)|utmcid=organic|utmctr=(not%20provided); _gid=GA1.3.1657577\r\nUpgrade-Insecure-Requests: 1\r\n\r\n» [Full request URI: http://nasahwi.csie.ntu.edu.tw/netfilz/login.html]\r\n» [HTTP 200 OK]\r\n(Prev request in frame: 5481)  
Response in frame: 27245  
» [Full request in frame: 27312]  
» [Next request in frame: 27312]  
File Size: 30 bytes  
» HTML Form URL Encoded: application/x-www-form-urlencoded  
» Form item: "username" = "190728731"  
» Form item: "password" = "49947045"
```

2.

```

Internet Protocol Version 4, Src: 192.168.88.194, Dst: 140.112.30.159
Ethernet II, Src: Intel(R) Dual Band Wireless-AC (82:5F:A6:28:E9:57), Dst: Intel(R) Dual Band Wireless-AC (82:5F:A6:28:E9:57)
    ... .. EID = Header Length: 20 bytes (5)
    ... .. Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 900
    Identification: 0xc349 (4937)
    Flags: 0x4000, Don't Fragment
    Fragment Offset: 0
    Time to live: 64
    Protocol: TCP (6)
    Header checksum: 0x5f57 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.88.194
    Destination: 140.112.30.159
Transmission Control Protocol, Src Port: 54336, Dst Port: 443, Seq: 582, Ack: 4593, Len: 938
    Source Port: 54336
    Destination Port: 443
    [Stream index: 17]
    [TCP Segment Len: 938]
    Sequence number: 582 (relative sequence number)
    Sequence number (raw): 366259940
    Next sequence number: 1528 (relative sequence number)
    Acknowledgment number: 4593 (relative ack number)
    Acknowledgment number (raw): 208727238
    1000 .... = Header Length: 32 bytes (8)
    Flags: 0x018 (PSH, ACK)
    Window size value: 501
    [Calculated window size: 64128]
    [Window size scaling factor: 128]
    Checksum: 0xc84a [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
    Options (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    [Seq/Ack analysis]
    [Timestamps]
    TCP payload: 938 bytes
Transport Layer Security
    TLSv1.3 Record Layer: Application Data Protocol: http-over-tls
Hypertext Transfer Protocol
    POST /netfix/login.html HTTP/1.1\r\n
    [Expert Info (Chat/Sequence): POST /netfix/login.html HTTP/1.1\r\n]
    Request Method: POST
    Request URI: /netfix/login.html
    Request Version: HTTP/1.1
    Host: nsasahel.csie.ntu.edu.tw/r/n
    User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:97.0) Gecko/20100101 Firefox/97.0\r/n
    Accept: text/html,application/xhtml+xml,application/javascript;q=0.9,image/avif,image/webp,*/*;q=0.8\r/n
    Accept-Language: en-US,en;q=0.5\r/n
    Accept-Encoding: gzip, deflate, br\r/n
    Content-Type: application/x-www-form-urlencoded\r/n
    Content-Length: 36\r/n
    Origin: https://nsasahel.csie.ntu.edu.tw/r/n
    Connection: keep-alive\r/n
    Referer: https://nsasahel.csie.ntu.edu.tw/netfix/r/n
    [Truncated Cookie: gaGA1.3.1673044394.1645467041.1645546077.1645546077.1.utmz=1543009519.1673044394.1645467041.1645546077.1.1.utmcsrc=google|utmcrc=(organic)|utmcrc=organic|utmcrc=(notK20provided); _gid=GA1.3.1657577
    [Full request URI: https://nsasahel.csie.ntu.edu.tw/netfix/login.html]
    [HTTP request 172]
    Response in frame: 90(14)
    Next request in frame: 90(1)
    File Data: 36 bytes
    ... ..
    Form item: "password" = "W490476AT5"

```

農場危機

1. Wireshark 打開，把給的檔案塞進去，filter設定

```
1 | ip.addr==140.112.30.159 #IP為token網站的IP
```

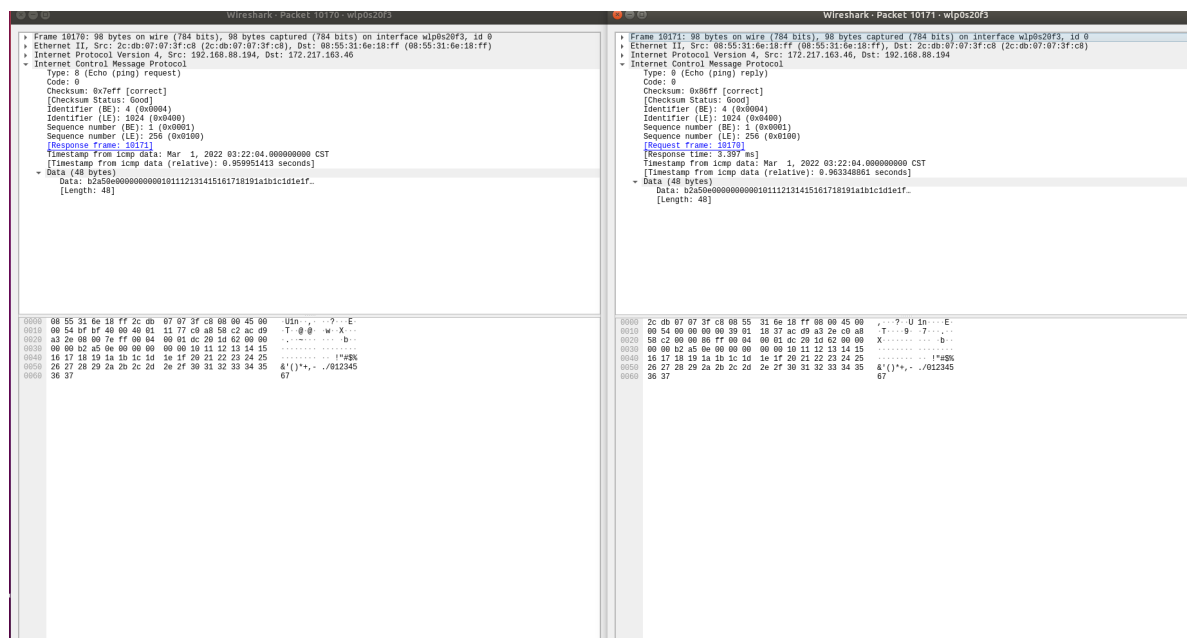
找到一個很奇怪的PNG回應(會找它是因為全部只有他是PNG)，丟到題目給的token網站，出現Wireshark的圖片，回到Wireshark，把那個PNG封包導出，看起來是一個QR Code，

偷偷點開token網頁的src，發現內容好像有跑某個script，看了一下是對圖片做某些變換，把script抓下來，在本地對QR Code跑，就找到答案了

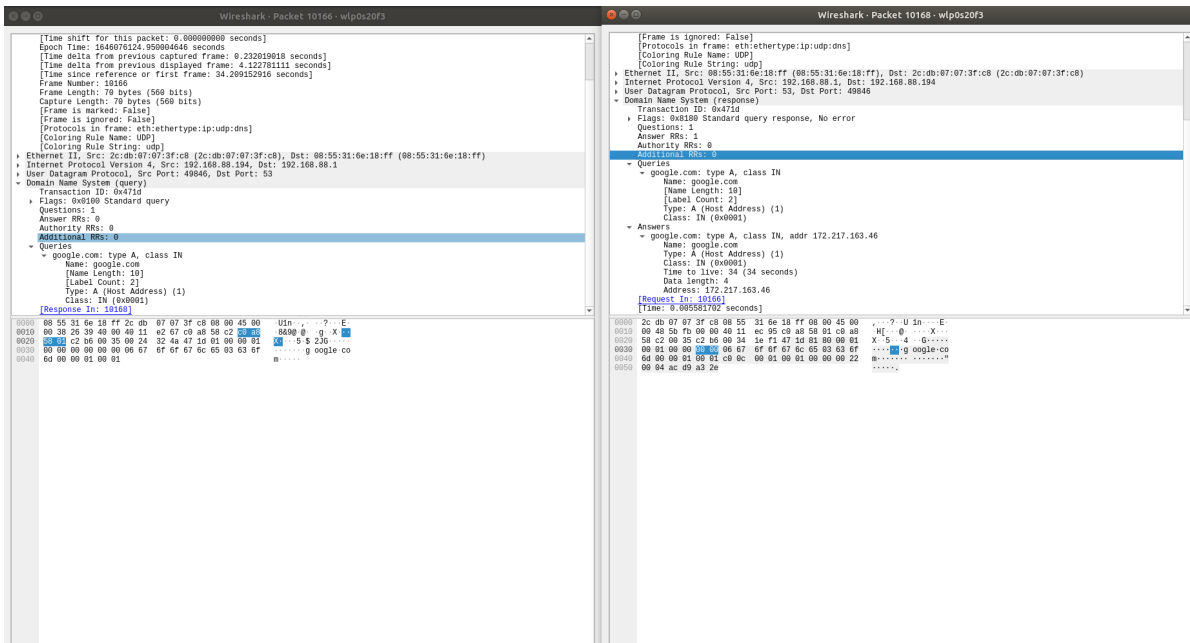
2. NASA{2022_pig_pig 🐷🐷🐷}

這麼多的網路協定要是能全部都認識的話該有多好

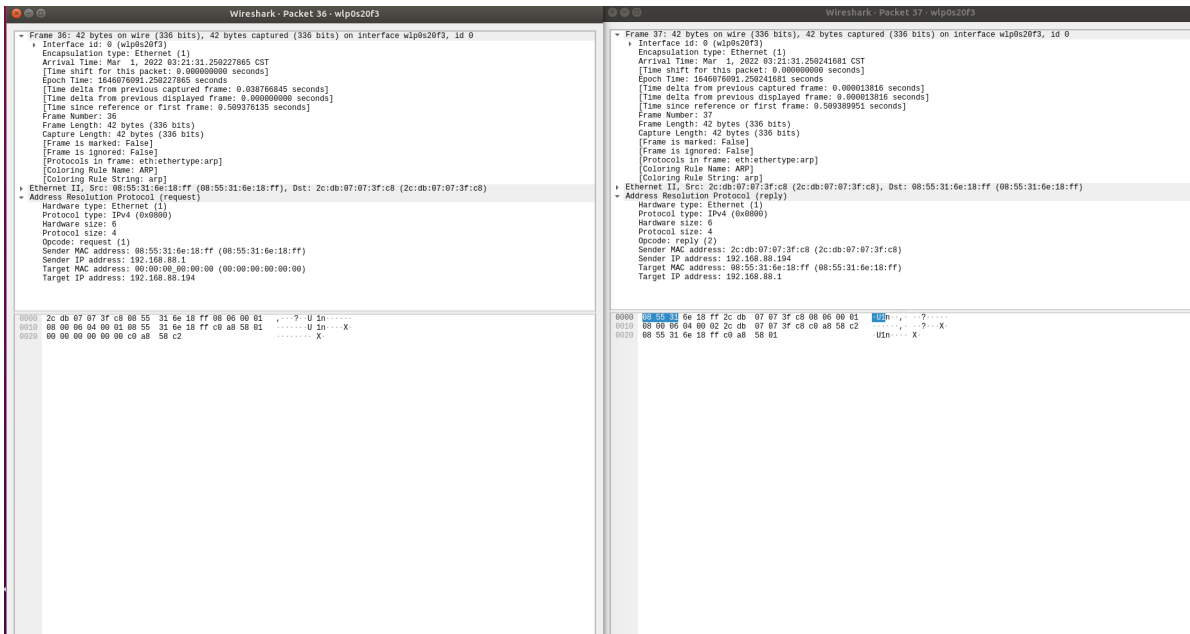
1. ICMP運作在Network Layer，它作為IP Packet的一個Payload, 提供各式各樣網路層的資訊，比如封包傳送時在中間過程找不到目標地址爛掉，就會得到Network Unreachable的ICMP封包，或者是像Ping之類的功能一樣，送出對應的Type跟Code(Type 8, Code 0)，回傳對應的Echo(Type 0, Code 0)，下圖就是一個ping的來回封包截圖



2. DNS Query跑在應用層上，用的是UDP協議(Send: UDP ?-> 53, Recv: ?-> 53)，主要用來做域名->IP Address的查詢，其中內部有許多不同的設置方式(A, CNAME... etc)，導致域名能有不同的設計對到不同或同一個實體IP



3. ARP跑在Network Layer上，是拿來建立IP->MAC Address間的映射關係用的，因為一台設備不會一開始就有所有接收端的MAC Address, 且使用同個IP的目標機器也不會一直一樣，但是每個封包都要跑在L2上(都要有一個L2的Address), 所以就有ARP這個協議經由廣播的方式去隨時跟蹤IP->MAC Address的對應關係。



4. DHCP跑在Application Layer上，是用來動態分配NAT底下內網IP的協議，使用UDP封包，減緩了IPv4耗盡的速度，也可以拿來做複雜的filter功能，阻擋某些特定封包進出某些機器的能力。



SA

Permission

I. Basic

1. 否，該目錄沒有r權限
2. 是，該目錄有r權限，父目錄有x
3. 是，該目錄有r權限，父目錄有x
4. 是，該目錄有x權限，父目錄有x
5. 否，該目錄沒有x權限
6. 否，dir2沒有x權限
7. 否，該檔案沒有w權限
8. 是，該檔案有w權限，，父目錄有x
9. 是，該資料夾有w權限，，父目錄有x
10. 否，該資料夾沒有w權限

II. ACL

ACL for specific user: <https://officeguide.cc/linux-acl-access-control-list-setfacl-getfacl-command-tutorial/>

1. 我們不是該資料夾的擁有者而且other沒有w權限

```
1 | chmod 700 40947047s
2 | setfacl -m g:ta:rx 40947047s
```

```
1 | # friend: 40947047s
2 | setfacl -m u:40947030s:x 40947047s      #b
3 | cd 40947047s                          #b
4 | chmod 700 chatroom                    #b
5 | setfacl -m u:40947030s:rw chatroom/    #b
6 | setfacl -m d:u:40947030s:rw chatroom/  #c
7 | chmod 500 chatroom/                  #d
8 | setfacl -m m::rx chatroom/            #d
```

```
1 | chmod 700 wordle/                    #a
2 | setfacl -m g:ta:rw ./.wordle/game.sh  #a
3 | setfacl -m g:ta:x wordle/            #a
4 | chmod 700 wordlist.txt                #b
```

c.

setuid是讓其他用戶在執行某個檔案的時候,使用擁有者的身份去執行。

如果今天有一個script,有寫檔的功能跟setuid,那我們將有權限寫檔到某個我們原本沒有權限的檔案因為setuid可以用該檔案擁有者的權限去寫入。

Shell Scripting

getopts --help

getopts: <https://stackoverflow.com/questions/11279423/bash-getopts-with-multiple-and-mandatory-options>

check time stamp: <https://stackoverflow.com/questions/806906/how-do-i-test-if-a-variable-is-a-number-in-bash>

get line of a file: <https://stackoverflow.com/questions/26789762/bash-difference-between-rw-string-and-string-in-variable>

regex: <https://regex101.com/>

timestamp conversion: <https://checkmk.com/linux-knowledge/convert-unix-timestamp-date>

bash's regex: <https://stackoverflow.com/questions/1891797/capturing-groups-from-a-grep-regex>

bash's regex's manual: https://en.wikibooks.org/wiki/Regular_Expressions/POSIX-Extended_Regular_Expressions

sort array in bash: <https://stackoverflow.com/questions/7442417/how-to-sort-an-array-in-bash>

2.

Get substr: <https://reactgo.com/bash-get-first-character-of-string/>

Function argument: <https://stackoverflow.com/questions/6212219/passing-parameters-to-a-bash-function>

Check directory: <https://devconnected.com/how-to-check-if-file-or-directory-exists-in-bash/>

Create repeat character: <http://www.unixcl.com/2009/03/repeat-character-in-bash-scripting.html>

man ls

Check substr: <https://stackoverflow.com/questions/229551/how-to-check-if-a-string-contains-a-substring-in-bash>

Symlink Checking: <https://koenwoortman.com/bash-script-check-if-file-is-symlink/>

Get symlink target and get fullpath: <https://stackoverflow.com/questions/7665/how-to-resolve-symbolic-links-in-a-shell-script>

sort with ascii value: <https://stackoverflow.com/questions/5296428/how-to-sort-a-text-file-according-to-character-code-or-ascii-code-value>