jamk.fi

# Assignment 01

## Data Security Testing

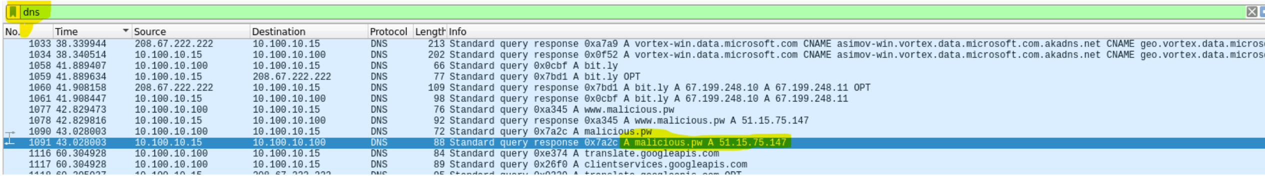
Jere Pesonen – TTV18S1

10-2020
Tieto – ja viestintätekniikka
Tekniikan ja liikenteen ala

Jyväskylän ammattikorkeakoulu
JAMK University of Applied Sciences

# Phishcap – Part1

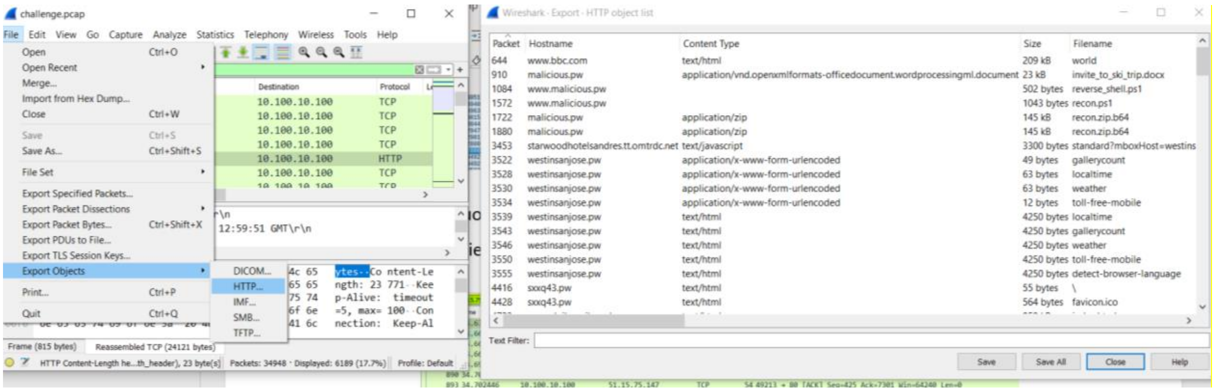
I started with the tip and filtered traffic with ”dns”. There were query for malicious.pw, which indeed sounded suspicious. Sites ip was 51.15.75.147.



dns

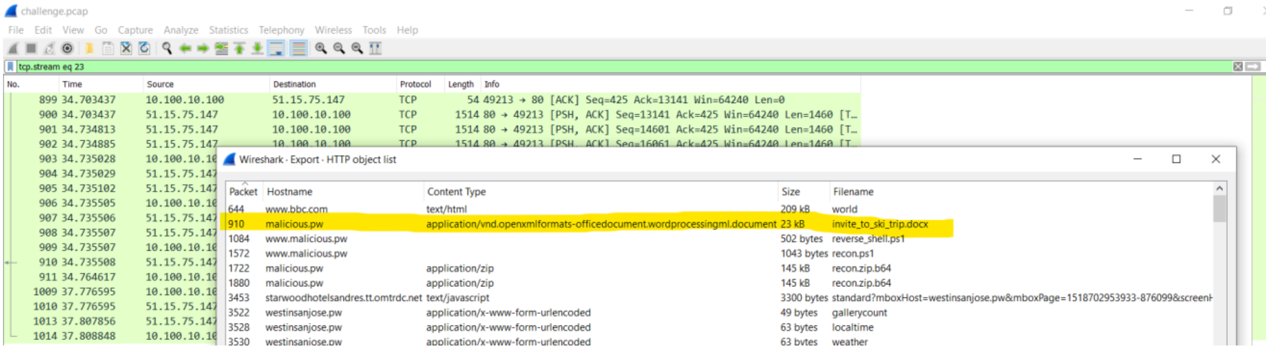
```
No.   Time       Source          Destination     Protocol Length Info
 1033 38.339944  208.67.222.222  10.100.10.15    DNS   213 Standard query response 0xa7a9 A vortex-win.data.microsoft.com CNAME asimov-win.vortex.data.microsoft.com.akadns.net CNAME geo.vortex.data.micros
 1034 38.340514  10.100.10.15    10.100.10.100   DNS   202 Standard query response 0x0f52 A vortex-win.data.microsoft.com CNAME asimov-win.vortex.data.microsoft.com.akadns.net CNAME geo.vortex.data.micros
 1058 41.889407  10.100.10.100   10.100.10.15    DNS    66 Standard query 0x0cbf A bit.ly
 1059 41.889634  10.100.10.15    208.67.222.222  DNS    77 Standard query 0x7bd1 A bit.ly OPT
 1060 41.908158  208.67.222.222  10.100.10.15    DNS   109 Standard query response 0x7bd1 A bit.ly A 67.199.248.10 A 67.199.248.11 OPT
 1061 41.908447  10.100.10.15    10.100.10.100   DNS    98 Standard query response 0x0cbf A bit.ly A 67.199.248.10 A 67.199.248.11
 1077 42.829473  10.100.10.100   10.100.10.15    DNS    76 Standard query 0xa345 A www.malicious.pw
 1078 42.829816  10.100.10.15    10.100.10.100   DNS    92 Standard query response 0xa345 A www.malicious.pw A 51.15.75.147
 1090 43.028003  10.100.10.100   10.100.10.15    DNS    72 Standard query 0x7a2c A malicious.pw
 1091 43.028003  10.100.10.15    10.100.10.100   DNS    88 Standard query response 0x7a2c A malicious.pw A 51.15.75.147
 1116 60.304928  10.100.10.100   10.100.10.15    DNS    84 Standard query 0xe374 A translate.googleapis.com
 1117 60.304928  10.100.10.100   10.100.10.15    DNS    89 Standard query 0x26f0 A clientservices.googleapis.com
```

I filtered traffic with destination to that ip and found interesting GET request, for docx file ”invite_to_ski_trip”.



| Packet | Hostname | Content Type | Size | Filename |
|---|---|---|---|---|
| 644 | www.bbc.com | text/html | 209 kB | world |
| 910 | malicious.pw | application/vnd.openxmlformats-officedocument.wordprocessingml.document | 23 kB | invite_to_ski_trip.docx |
| 1084 | www.malicious.pw | | 502 bytes | reverse_shell.ps1 |
| 1572 | www.malicious.pw | | 1043 bytes | recon.ps1 |
| 1722 | malicious.pw | application/zip | 145 kB | recon.zip.b64 |
| 1880 | malicious.pw | application/zip | 145 kB | recon.zip.b64 |
| 3453 | starwoodhotelsandres.tt.omtrdc.net | text/javascript | 3300 bytes | standard?mboxHost=westins |
| 3522 | westinsanjose.pw | application/x-www-form-urlencoded | 49 bytes | gallerycount |
| 3528 | westinsanjose.pw | application/x-www-form-urlencoded | 63 bytes | localtime |
| 3530 | westinsanjose.pw | application/x-www-form-urlencoded | 63 bytes | weather |
| 3534 | westinsanjose.pw | application/x-www-form-urlencoded | 12 bytes | toll-free-mobile |
| 3539 | westinsanjose.pw | text/html | 4250 bytes | localtime |
| 3543 | westinsanjose.pw | text/html | 4250 bytes | gallerycount |
| 3546 | westinsanjose.pw | text/html | 4250 bytes | weather |
| 3550 | westinsanjose.pw | text/html | 4250 bytes | toll-free-mobile |
| 3555 | westinsanjose.pw | text/html | 4250 bytes | detect-browser-language |
| 4416 | sxxq43.pw | text/html | 55 bytes | \ |
| 4428 | sxxq43.pw | text/html | 564 bytes | favicon.ico |

With ip source filter i found the file itself. I downloaded the docx file from to my kali virtul machine, from file/export objects/http. Packet 910



challenge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Wireshark · Export · HTTP object list

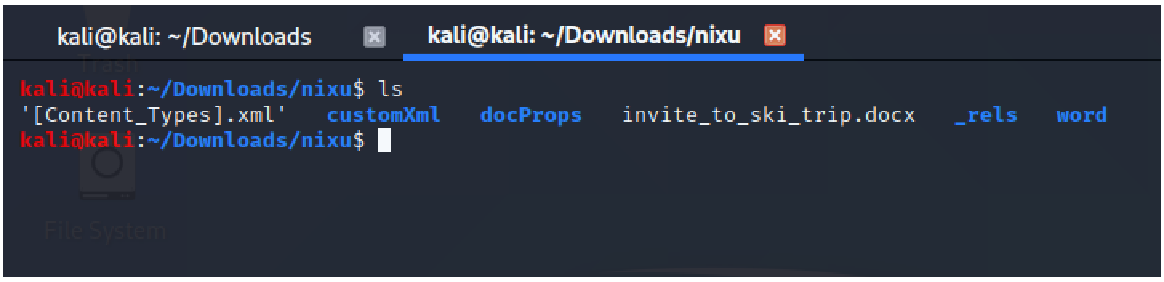
tcp.stream eq 23

```
No.   Time       Source          Destination     Protocol Length Info
  899 34.703437  10.100.10.100   51.15.75.147    TCP     54 49213 → 80 [ACK] Seq=425 Ack=13141 Win=64240 Len=0
  900 34.703437  51.15.75.147    10.100.10.100   TCP   1514 80 → 49213 [PSH, ACK] Seq=13141 Ack=425 Win=64240 Len=1460 [T…
  901 34.734813  51.15.75.147    10.100.10.100   TCP   1514 80 → 49213 [PSH, ACK] Seq=14601 Ack=425 Win=64240 Len=1460 [T…
  902 34.734885  51.15.75.147    10.100.10.100   TCP   1514 80 → 49213 [PSH, ACK] Seq=16061 Ack=425 Win=64240 Len=1460 [T…
  903 34.735028  10.100.10.10
  904 34.735029  51.15.75.147
  905 34.735102  51.15.75.147
  906 34.735505  10.100.10.10
  907 34.735506  51.15.75.147
  908 34.735507  51.15.75.147
  909 34.735507  10.100.10.10
  910 34.735508  51.15.75.147
  911 34.764617  10.100.10.10
 1009 37.776595  10.100.10.10
 1010 37.776595  51.15.75.147
 1013 37.807856  51.15.75.147
 1014 37.808848  10.100.10.10
```

| Packet | Hostname | Content Type | Size | Filename |
|---|---|---|---|---|
| 644 | www.bbc.com | text/html | 209 kB | world |
| 910 | malicious.pw | application/vnd.openxmlformats-officedocument.wordprocessingml.document | 23 kB | invite_to_ski_trip.docx |
| 1084 | www.malicious.pw | | 502 bytes | reverse_shell.ps1 |
| 1572 | www.malicious.pw | | 1043 bytes | recon.ps1 |
| 1722 | malicious.pw | application/zip | 145 kB | recon.zip.b64 |
| 1880 | malicious.pw | application/zip | 145 kB | recon.zip.b64 |
| 3453 | starwoodhotelsandres.tt.omtrdc.net | text/javascript | 3300 bytes | standard?mboxHost=westinsanjose.pw&mboxPage=1518702953933-876099&screenH |
| 3522 | westinsanjose.pw | application/x-www-form-urlencoded | 49 bytes | gallerycount |
| 3528 | westinsanjose.pw | application/x-www-form-urlencoded | 63 bytes | localtime |
| 3530 | westinsanjose.pw | application/x-www-form-urlencoded | 63 bytes | weather |

the flag can be found at the word document, but i desided not to open it, but unzip and find the flag from xml files.

on kali i unzipped docx file and got:

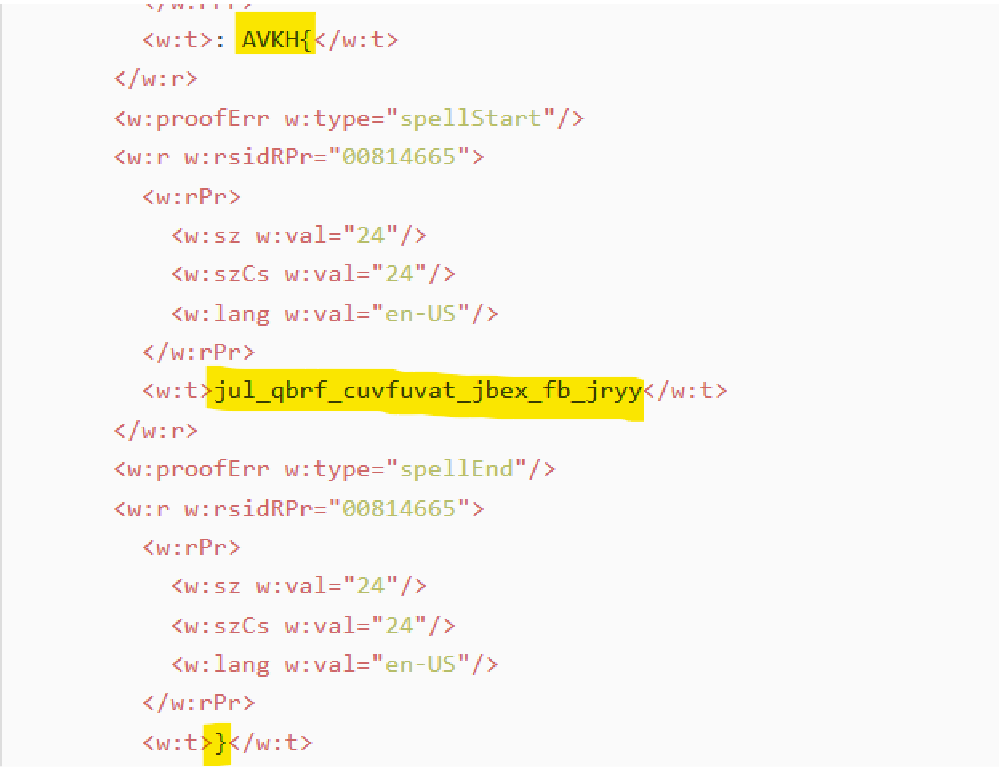


```
kali@kali:~/Downloads/nixu$ ls
'[Content_Types].xml'   customXml   docProps   invite_to_ski_trip.docx   _rels   word
kali@kali:~/Downloads/nixu$ 
```

the content of the word document itself can be found at word/document.xml.

I copied the content of document.xml file to browser tool which pretty prints xml.

And first flag was found.



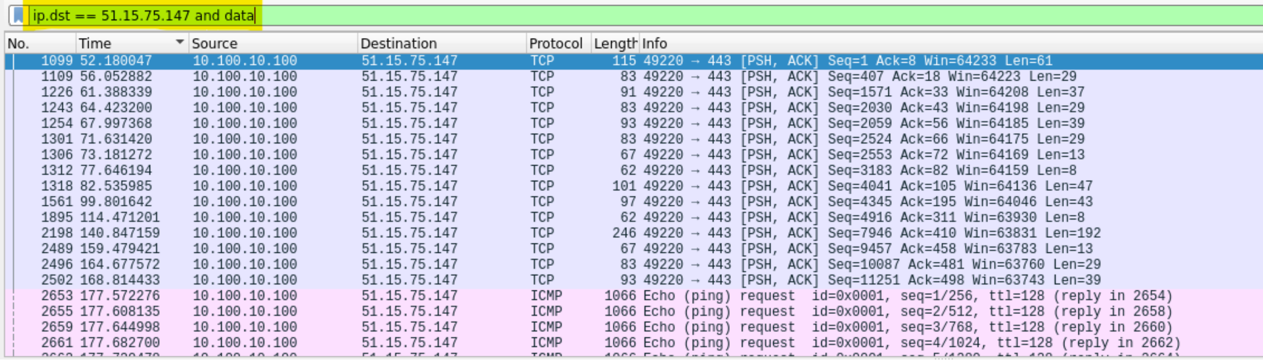
```
          <w:t>: AVKH{</w:t>
        </w:r>
        <w:proofErr w:type="spellStart"/>
        <w:r w:rsidRPr="00814665">
          <w:rPr>
            <w:sz w:val="24"/>
            <w:szCs w:val="24"/>
            <w:lang w:val="en-US"/>
          </w:rPr>
          <w:t>jul_qbrf_cuvfuvat_jbex_fb_jryy</w:t>
        </w:r>
        <w:proofErr w:type="spellEnd"/>
        <w:r w:rsidRPr="00814665">
          <w:rPr>
            <w:sz w:val="24"/>
            <w:szCs w:val="24"/>
            <w:lang w:val="en-US"/>
          </w:rPr>
          <w:t>}</w:t>
```

it was encrypted with ceasar cipher with rotation of 13 (or rot13).

Flag = NIXU{why_does_phishing_work_so_well}

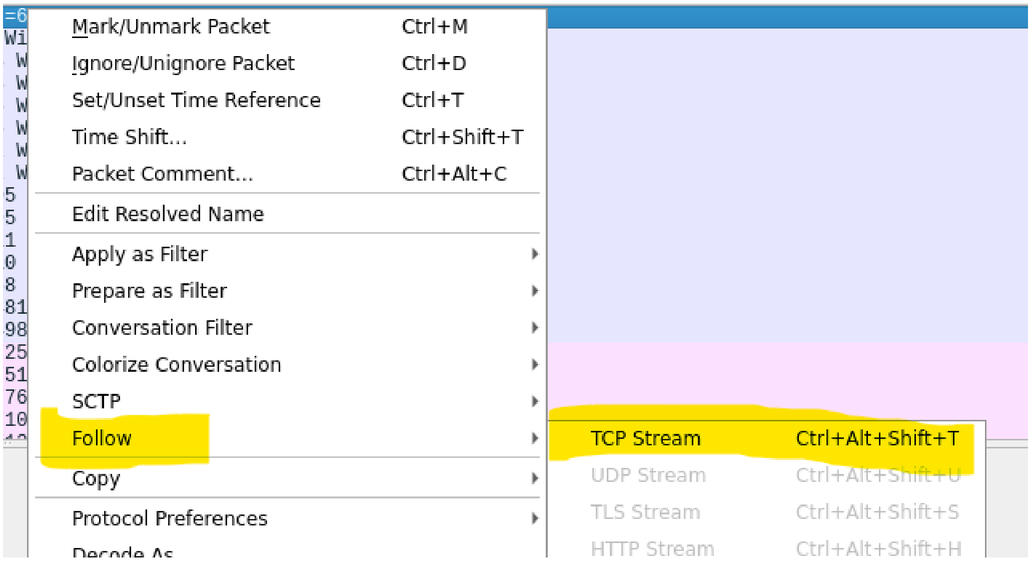
# Phishcap – Part2

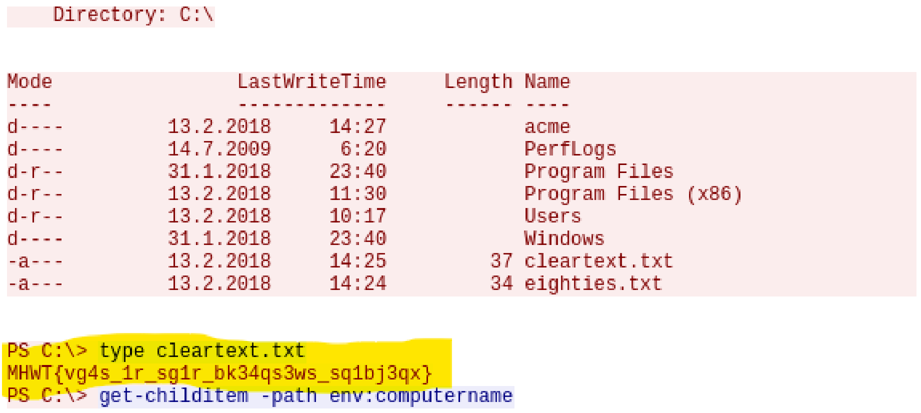
Second flag i found with a little accident. I just added ”data” to search parameters, and followed tcp stream. then i scrolled down tcp stream for a while, and file cleartext.txt was opened, and it cointained encrypted flag.



ip.dst == 51.15.75.147 and data

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1099 | 52.180047 | 10.100.10.100 | 51.15.75.147 | TCP | 115 | 49220 → 443 [PSH, ACK] Seq=1 Ack=8 Win=64233 Len=61 |
| 1109 | 56.052882 | 10.100.10.100 | 51.15.75.147 | TCP | 83 | 49220 → 443 [PSH, ACK] Seq=407 Ack=18 Win=64223 Len=29 |
| 1226 | 61.388339 | 10.100.10.100 | 51.15.75.147 | TCP | 91 | 49220 → 443 [PSH, ACK] Seq=1571 Ack=33 Win=64208 Len=37 |
| 1243 | 64.423200 | 10.100.10.100 | 51.15.75.147 | TCP | 83 | 49220 → 443 [PSH, ACK] Seq=2030 Ack=43 Win=64198 Len=29 |
| 1254 | 67.997368 | 10.100.10.100 | 51.15.75.147 | TCP | 93 | 49220 → 443 [PSH, ACK] Seq=2059 Ack=56 Win=64185 Len=39 |
| 1301 | 71.631420 | 10.100.10.100 | 51.15.75.147 | TCP | 83 | 49220 → 443 [PSH, ACK] Seq=2524 Ack=66 Win=64175 Len=29 |
| 1306 | 73.181272 | 10.100.10.100 | 51.15.75.147 | TCP | 67 | 49220 → 443 [PSH, ACK] Seq=2553 Ack=72 Win=64169 Len=13 |
| 1312 | 77.646194 | 10.100.10.100 | 51.15.75.147 | TCP | 62 | 49220 → 443 [PSH, ACK] Seq=3183 Ack=82 Win=64159 Len=8 |
| 1318 | 82.535985 | 10.100.10.100 | 51.15.75.147 | TCP | 101 | 49220 → 443 [PSH, ACK] Seq=4041 Ack=105 Win=64136 Len=47 |
| 1561 | 99.801642 | 10.100.10.100 | 51.15.75.147 | TCP | 97 | 49220 → 443 [PSH, ACK] Seq=4345 Ack=195 Win=64046 Len=43 |
| 1895 | 114.471201 | 10.100.10.100 | 51.15.75.147 | TCP | 62 | 49220 → 443 [PSH, ACK] Seq=4916 Ack=311 Win=63930 Len=8 |
| 2198 | 140.847159 | 10.100.10.100 | 51.15.75.147 | TCP | 246 | 49220 → 443 [PSH, ACK] Seq=7946 Ack=410 Win=63831 Len=192 |
| 2489 | 159.479421 | 10.100.10.100 | 51.15.75.147 | TCP | 67 | 49220 → 443 [PSH, ACK] Seq=9457 Ack=458 Win=63783 Len=13 |
| 2496 | 164.677572 | 10.100.10.100 | 51.15.75.147 | TCP | 83 | 49220 → 443 [PSH, ACK] Seq=10087 Ack=481 Win=63760 Len=29 |
| 2502 | 168.814433 | 10.100.10.100 | 51.15.75.147 | TCP | 93 | 49220 → 443 [PSH, ACK] Seq=11251 Ack=498 Win=63743 Len=39 |
| 2653 | 177.572276 | 10.100.10.100 | 51.15.75.147 | ICMP | 1066 | Echo (ping) request  id=0x0001, seq=1/256, ttl=128 (reply in 2654) |
| 2655 | 177.608135 | 10.100.10.100 | 51.15.75.147 | ICMP | 1066 | Echo (ping) request  id=0x0001, seq=2/512, ttl=128 (reply in 2658) |
| 2659 | 177.644998 | 10.100.10.100 | 51.15.75.147 | ICMP | 1066 | Echo (ping) request  id=0x0001, seq=3/768, ttl=128 (reply in 2660) |
| 2661 | 177.682700 | 10.100.10.100 | 51.15.75.147 | ICMP | 1066 | Echo (ping) request  id=0x0001, seq=4/1024, ttl=128 (reply in 2662) |


Mark/Unmark Packet Ctrl+M
Ignore/Unignore Packet Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comment... Ctrl+Alt+C
Edit Resolved Name
Apply as Filter
Prepare as Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As
TCP Stream Ctrl+Alt+Shift+T
UDP Stream Ctrl+Alt+Shift+U
TLS Stream Ctrl+Alt+Shift+S
HTTP Stream Ctrl+Alt+Shift+H



```
    Directory: C:\


Mode                LastWriteTime     Length Name
----                -------------     ------ ----
d----        13.2.2018     14:27            acme
d----        14.7.2009      6:20            PerfLogs
d-r--        31.1.2018     23:40            Program Files
d-r--        13.2.2018     11:30            Program Files (x86)
d-r--        13.2.2018     10:17            Users
d----        31.1.2018     23:40            Windows
-a---        13.2.2018     14:25         37 cleartext.txt
-a---        13.2.2018     14:24         34 eighties.txt


PS C:\> type cleartext.txt
MHWT{vg4s_1r_sg1r_bk34qs3ws_sq1bj3qx}
PS C:\> get-childitem -path env:computername
```

Flag was encrypted the same as the first one

Flag: NIXU{wh4t_1s_th1s_cl34rt3xt_tr1ck3ry}

There were also GET request for ”reverse_shell.ps1” file between the packets.  I think hacker planted reverse shell script to target computer through ”invite_to_ski_trip.docx” . So the tcp stream is hacker reading target computers files through ssl.