Wireshark Analysis for DWD <file> command for all encryption

Server Port = 49154

Caesar Cipher (Shift = 12)

Table of alphabets and their shifted by 12 versions

Letter	Letter shifted by 13
a/A	m/M
b/B	n/N
c/C	o/O
d/D	p/P
e/E	q/Q
f/F	r/R
g/G	s/S
h/H	t/T
i/I	u/U
j/J	v/V
k/K	w/W
l/L	x/X
m/M	y/Y
n/N	z/Z
o/O	a/A
p/P	b/B
q/Q	c/C
r/R	d/D
s/S	e/E
t/T	f/F

u/U	g/G
v/V	h/H
x/X	i/I
y/Y	j/J
z/Z	k/K

```
┌──(kali㉿kali)-[~/Downloads/newproj/client]
└─$ python client.py
Socket Created at client end
Connnected to Server IP= 127.0.0.1
DWD server_test_file.txt
STATUS: OK
```

Firstly we are sending the command from client to server

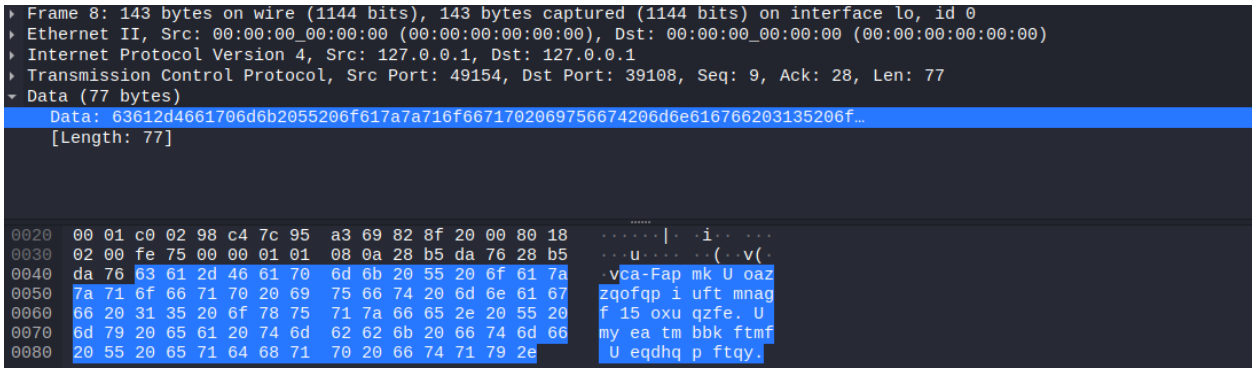
Also for the command we are sending i.e. DWD server_test_file.txt
We expect ca-PIP eqdhqd_fdef_ruxq.fif

```
▸ Frame 4: 93 bytes on wire (744 bits), 93 bytes captured (744 bits) on interface lo, id 0
▸ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
▸ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▸ Transmission Control Protocol, Src Port: 39108, Dst Port: 49154, Seq: 1, Ack: 1, Len: 27
▾ Data (27 bytes)
    Data: 63612d504950206571646871645f667165665f727578712e666a66
    [Length: 27]

0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ........ ......E·
0010  00 4f d2 c6 40 00 40 06  69 e0 7f 00 00 01 7f 00   ·O··@·@· i·····
0020  00 01 98 c4 c0 02 82 8f  1f e5 7c 95 a3 61 80 18   ........ ··|··a··
0030  02 00 fe 43 00 00 01 01  08 0a 28 b5 da 76 28 b5   ···C···· ··(··v(·
0040  8a c7 63 61 2d 50 49 50  20 65 71 64 68 71 64 5f   ··ca-PIP  eqdhqd_
0050  66 71 65 66 5f 72 75 78  71 2e 66 6a 66           fqef_rux q.fjf
```

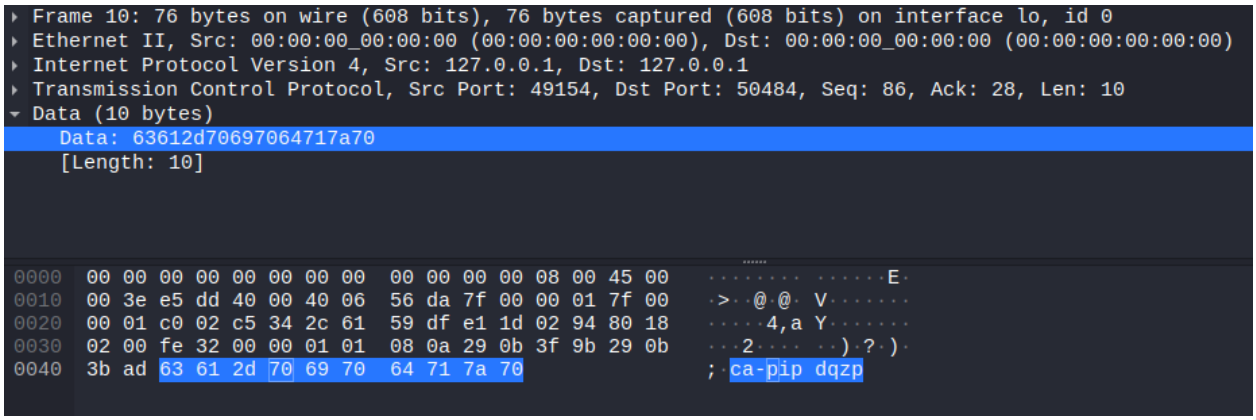
Here we see that source protocol is random port but the destination port is 49154
We have an encrypted command on the dump which matches our expectations.

Now for contents of server_test_file.txt are
Today I connected with about 15 clients. I am so happy that I served them.
We expect encrypted text to be
Fapmk U oazzqofqp iuft mnagf 15 oxuqzfe. U my ea tmbbk ftmf U eqdhqp ftqy.



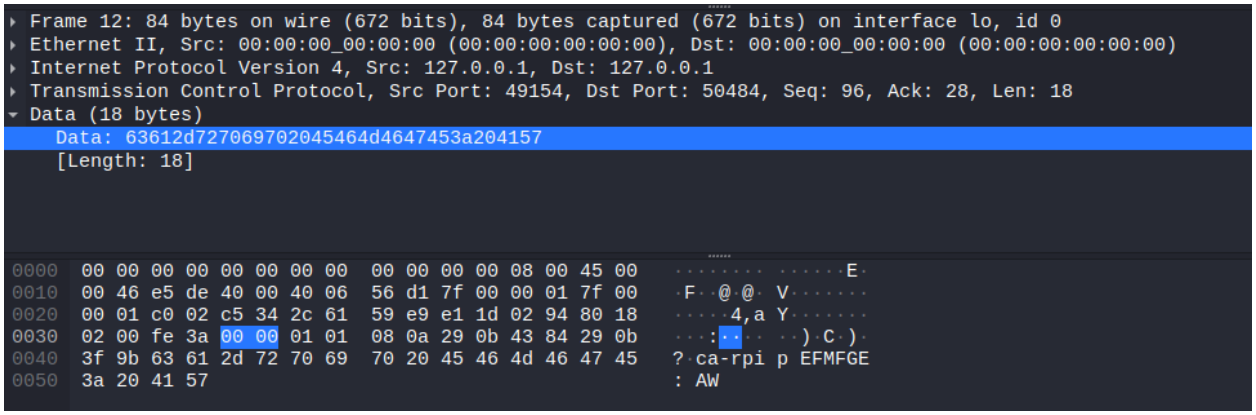
Text in the dump is same as what we expected.

Now we look for response footer



We found footer as pipdqzp which encrypted version of dwdrend

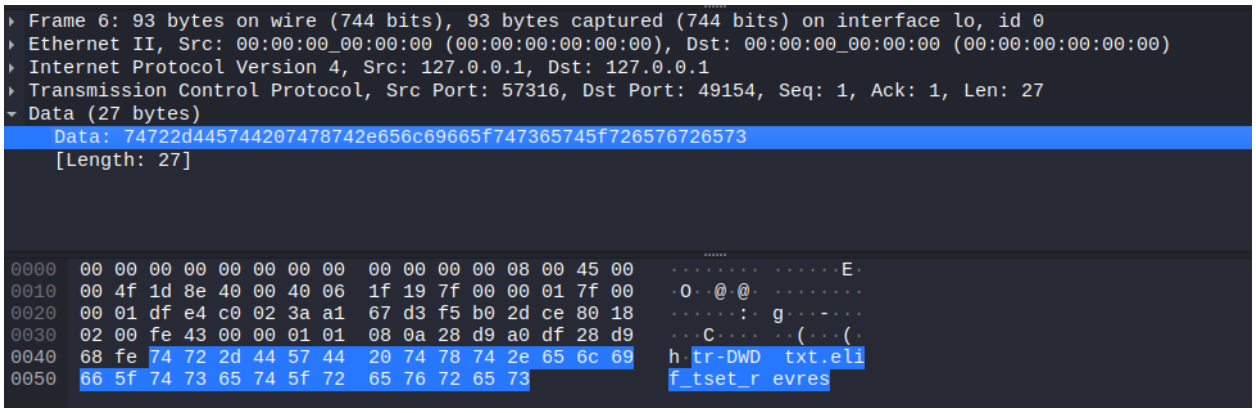
Now we look for status response



We found status response as rpip EFMFGE: AW which is encrypted version of fdwd STATUS: OK

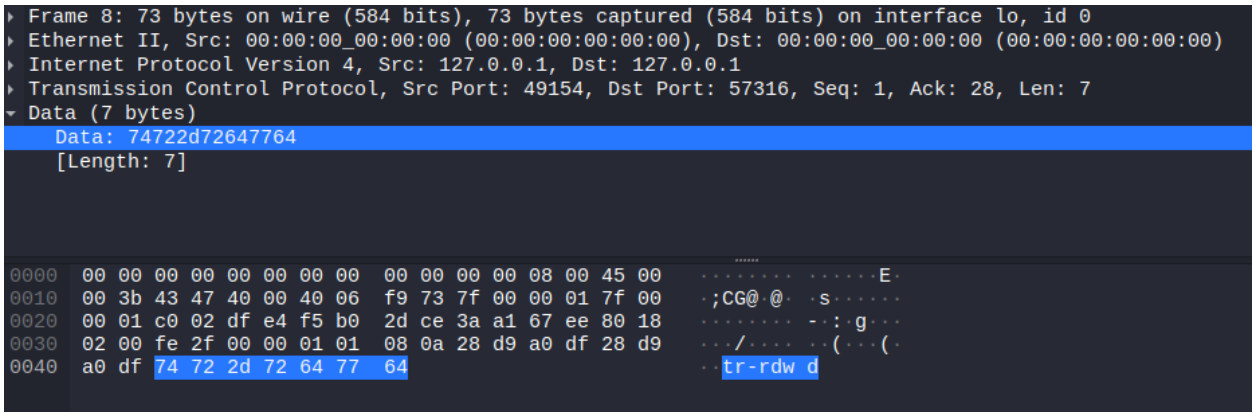
Transpose

Our command is DWD server_test_file.txt and we are expecting encrypted command to be DWD txt.elif_tset_revres



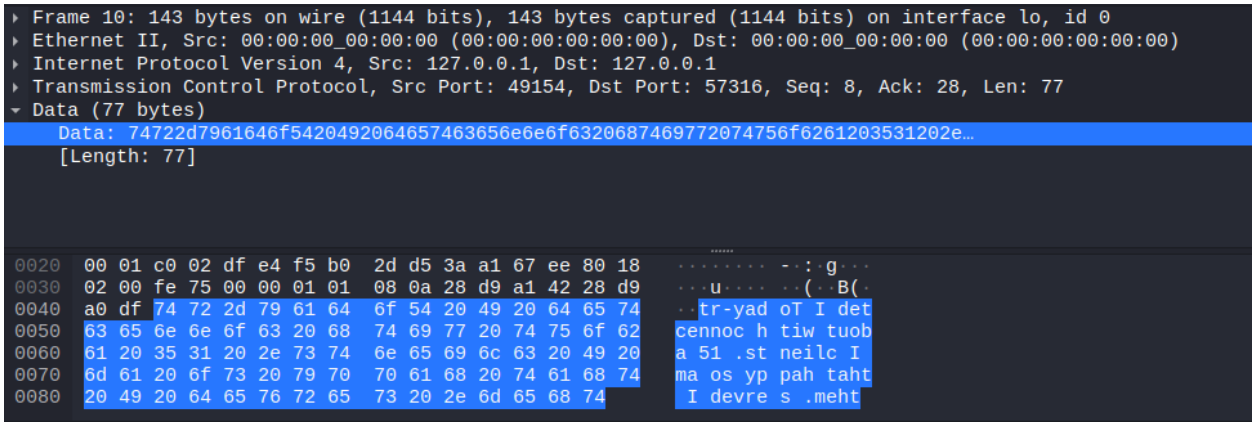
Encrypted command in the dump is the same as what we expected.

We look for response header

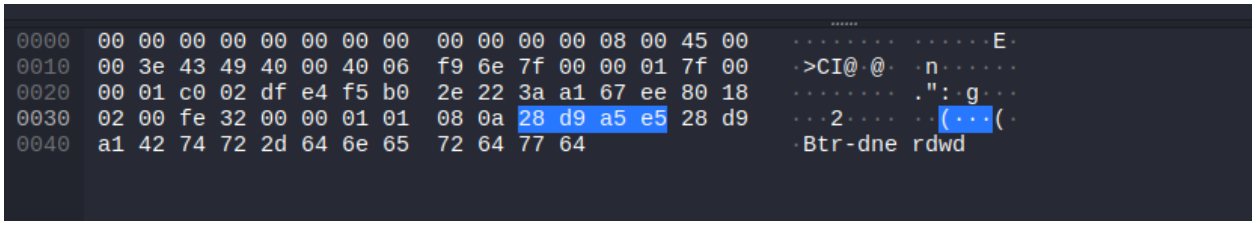


Originally we expect it to be dwdr but it is reversed

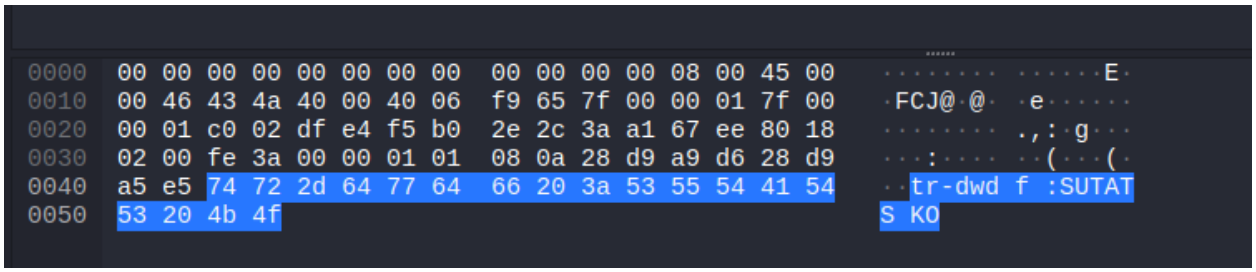
Now for contents of server_test_file.txt are
Today I connected with about 15 clients. I am so happy that I served them.
We expect encrypted text to be
yad-oT I detcennoc htiw tuoba 51 .neilc I ma os yppah tath I devsres .meht



The response file matches our expectations also we see a tr in front which indicates that data is transposed



We also see a response footer encrypted version of dwdrend



And we also have response fdwd STATUS: OK

Plain

Firstly we look for the command

```
▸ Frame 4: 93 bytes on wire (744 bits), 93 bytes captured (744 bits) on interface lo, id 0
▸ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
▸ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▸ Transmission Control Protocol, Src Port: 41640, Dst Port: 49154, Seq: 1, Ack: 1, Len: 27
▾ Data (27 bytes)
    Data: 706c2d44574420736572665725f746573745f66696c652e747874
    [Length: 27]

0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ·····E·
0010  00 4f 58 a0 40 00 40 06  e4 06 7f 00 00 01 7f 00   ·OX·@·@· ·····
0020  00 01 a2 a8 c0 02 d9 89  8d cb f5 9f ea 3d 80 18   ·····=··
0030  02 00 fe 43 00 00 01 01  08 0a 28 f2 f0 c0 28 f2   ···C···· ··(···(·
0040  bf 3d 70 6c 2d 44 57 44  20 73 65 72 76 65 72 5f   =pl-DWD  server_
0050  74 65 73 74 5f 66 69 6c  65 2e 74 78 74            test_fil e.txt
```

We see the plain command DWD server_test_file.txt

Now we look for the response header

```
▸ Frame 6: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface lo, id 0
▸ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
▸ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▸ Transmission Control Protocol, Src Port: 49154, Dst Port: 41640, Seq: 1, Ack: 28, Len: 8
▸ [Malformed Packet: LBMSRS]

0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ·····E·
0010  00 3c 2e d4 40 00 40 06  0d e6 7f 00 00 01 7f 00   ·<.·@·@· ·····
0020  00 01 c0 02 a2 a8 f5 9f  ea 3d d9 89 8d e6 80 18   ·····=··
0030  02 00 fe 30 00 00 01 01  08 0a 28 f2 f0 c0 28 f2   ···0···· ··(···(·
0040  f0 c0 70 6c 2d 64 77 64  72 20                     ··pl-dwd r
```

We see the dwdr header

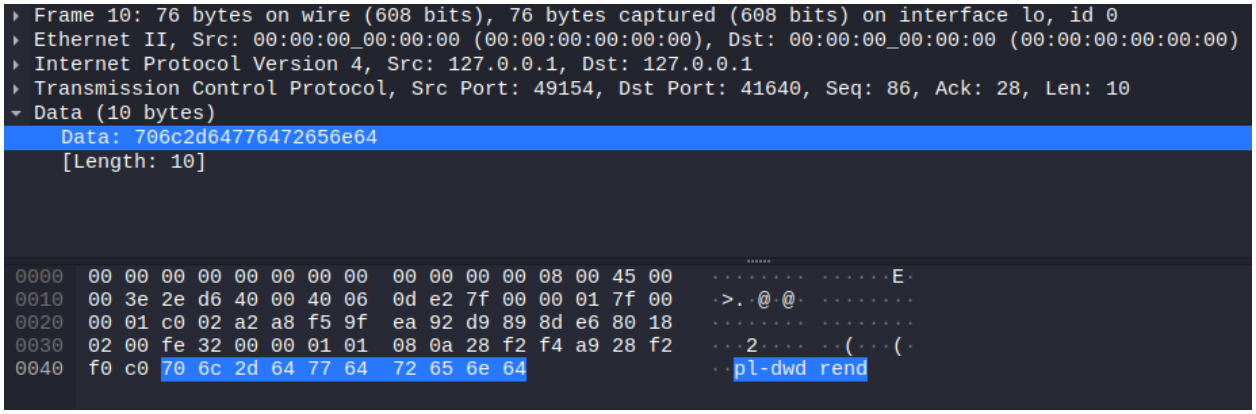
Now we look for response contents of the file

```
▸ Frame 8: 143 bytes on wire (1144 bits), 143 bytes captured (1144 bits) on interface lo, id 0
▸ Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
▸ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
▸ Transmission Control Protocol, Src Port: 49154, Dst Port: 41640, Seq: 9, Ack: 28, Len: 77
▾ Data (77 bytes)
    Data: 706c2d546f6461792049206f6e6e65637465642077697468206162f75742031352063…
    [Length: 77]

0020  00 01 c0 02 a2 a8 f5 9f  ea 45 d9 89 8d e6 80 18   ·····E·····
0030  02 00 fe 75 00 00 01 01  08 0a 28 f2 f0 c0 28 f2   ···u···· ··(···(·
0040  f0 c0 70 6c 2d 54 6f 64  61 79 20 49 20 63 6f 6e   ··pl-Tod ay I con
0050  6e 65 63 74 65 64 20 77  69 74 68 20 61 62 6f 75   nected w ith abou
0060  74 20 31 35 20 63 6c 69  65 6e 74 73 2e 20 49 20   t 15 cli ents. I
0070  61 6d 20 73 6f 20 68 61  70 70 79 20 74 68 61 74   am so ha ppy that
0080  20 49 20 73 65 72 76 65  64 20 74 68 65 6d 2e       I serve d them.
```

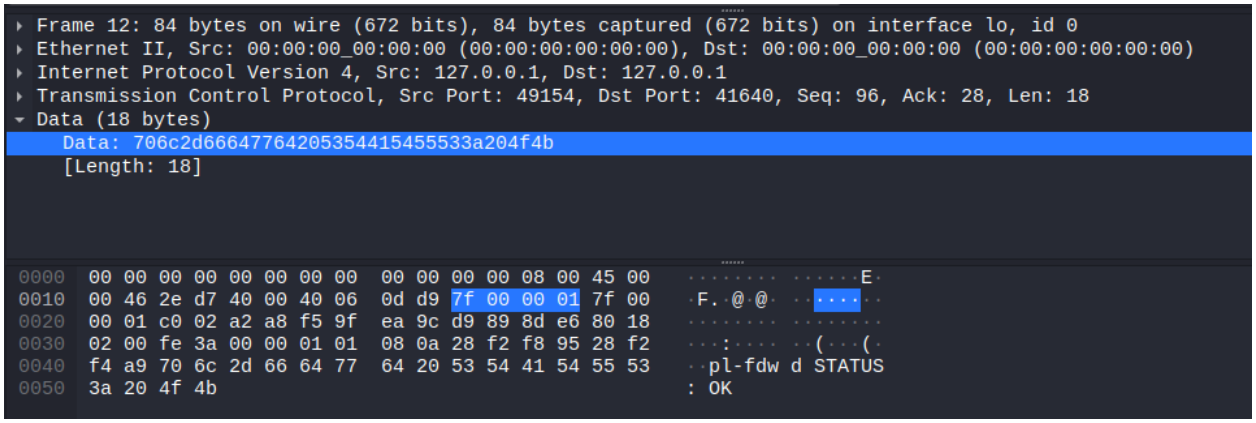
We have exact same contents of the file

Now we look for the response footer



We found the footer as dwdrend

Now we look for Status response



We found the status as fdwd STATUS: OK