



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΓΚΟΥΜΕ ΛΑΟΥΡΕΝΤΙΑΝ

ΑΜ: 031 18 014

ΕΞΑΜΗΝΟ: 7^ο

ΟΜΑΔΑ: 4

MAC ADDRESS: B4-69-21-1B-6C-FF

IPv4: Άσκ1: 10.3.20.67, Άσκ2: 147.102.131.71, Άσκ3: 147.102.131.188

ΌΝΟΜΑ ΥΠΟΛΟΓΙΣΤΗ: LAPTOP-B2DVAJKK

ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ: WINDOWS 10

ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ



ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 8: TELNET, FTP ΚΑΙ TFTP

Άσκηση 1: TELNET

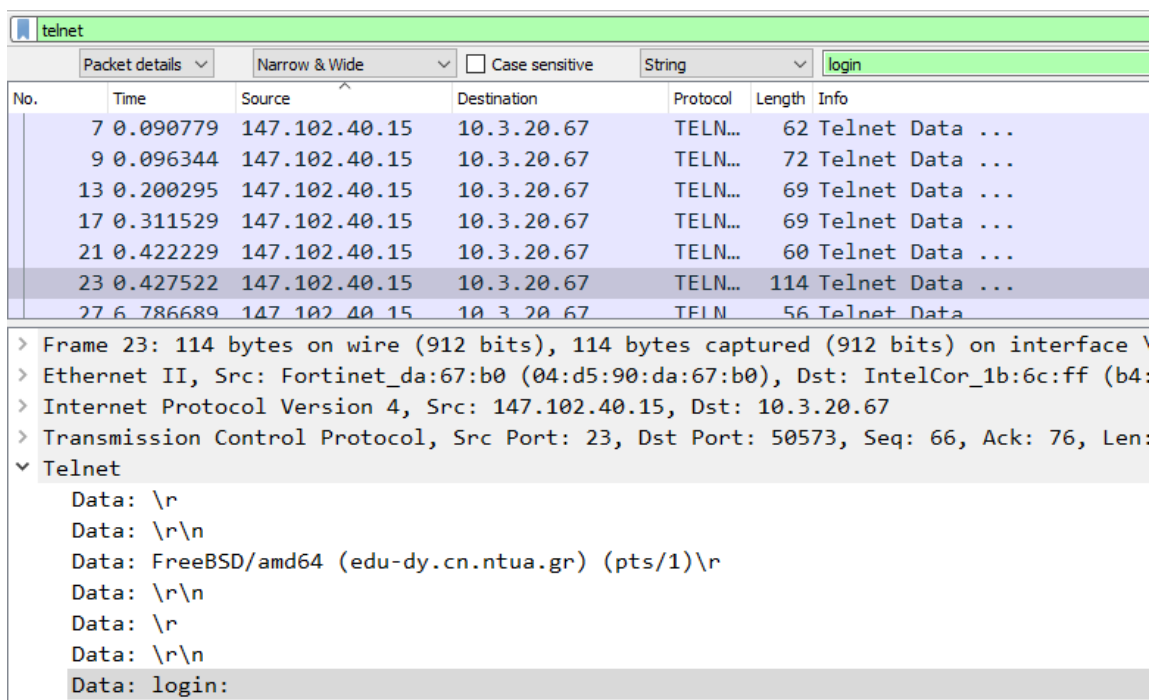
1.1) Το πρωτόκολλο εφαρμογής TELNET χρησιμοποιεί το πρωτόκολλο μεταφοράς TCP.

1.2) Χρησιμοποιούνται τα **ports 50573 και 23**.

1.3) Η θύρα 23 αντιστοιχεί στο πρωτόκολλο TELNET.

1.4) Με το φίλτρο απεικόνισης **telnet** βλέπουμε τεμάχια σχετιζόμενα με το πρωτόκολλο εφαρμογής telnet.

1.5) Βρίσκουμε με τις κατάλληλες ρυθμίσεις πως το επιθυμητό πακέτο είναι το υπ' αριθμόν 23.



No.	Time	Source	Destination	Protocol	Length	Info
7	0.090779	147.102.40.15	10.3.20.67	TELN...	62	Telnet Data ...
9	0.096344	147.102.40.15	10.3.20.67	TELN...	72	Telnet Data ...
13	0.200295	147.102.40.15	10.3.20.67	TELN...	69	Telnet Data ...
17	0.311529	147.102.40.15	10.3.20.67	TELN...	69	Telnet Data ...
21	0.422229	147.102.40.15	10.3.20.67	TELN...	60	Telnet Data ...
23	0.427522	147.102.40.15	10.3.20.67	TELN...	114	Telnet Data ...
27	6.786689	147.102.40.15	10.3.20.67	TELN...	56	Telnet Data ...

> Frame 23: 114 bytes on wire (912 bits), 114 bytes captured (912 bits) on interface \

> Ethernet II, Src: Fortinet_da:67:b0 (04:d5:90:da:67:b0), Dst: IntelCor_1b:6c:ff (b4:

> Internet Protocol Version 4, Src: 147.102.40.15, Dst: 10.3.20.67

> Transmission Control Protocol, Src Port: 23, Dst Port: 50573, Seq: 66, Ack: 76, Len:

▼ Telnet

Data: \r

Data: \r\n

Data: FreeBSD/amd64 (edu-dy.cn.ntua.gr) (pts/1)\r

Data: \r\n

Data: \r

Data: \r\n

Data: login:

Επομένως αναζητούμε στα πακέτα 1-22 για εντολές Telnet τύπου Echo. Βρίσκουμε με αύξουσα σειρά εμφάνισης:

- **Do Echo** (τεμάχιο 17: 147.102.40.15 → 10.3.20.67)
- **Will Echo** (τεμάχιο 20: 10.3.20.67 → 147.102.40.15)
- **Don't Echo** (τεμάχιο 21: 147.102.40.15 → 10.3.20.67)
- **Will Echo** (τεμάχιο 21: 147.102.40.15 → 10.3.20.67)
- **Won't Echo** (τεμάχιο 22: 10.3.20.67 → 147.102.40.15)

1.6) Ναι, ο edu-dy.cn.ntua.gr ζητάει από τον υπολογιστή μας να επαναλαμβάνει τους χαρακτήρες που λαμβάνει (τεμάχιο 17: Do Echo) και ο υπολογιστής μας **δέχεται** (τεμάχιο 20: Will Echo).

1.7) Ναι, ο edu-dy.cn.ntua.gr ζητάει από τον υπολογιστή μας να μην επαναλαμβάνει τους χαρακτήρες που λαμβάνει (τεμάχιο 21: Don't Echo) και ο υπολογιστής μας **δέχεται** (τεμάχιο 22: Won't Echo).

1.8) Ο edu-dy.cn.ntua.gr προτίθεται να επαναλαμβάνει τους χαρακτήρες που λαμβάνει από τον υπολογιστή μας (τεμάχιο 21: Will Echo).

1.9) Αναζητούμε μεταξύ των τεμαχίων που έχουν ως πηγή τον υπολογιστή μας και με αύξοντα αριθμό μεγαλύτερο του 23. Βρίσκουμε το ζητούμενο στο πακέτο 26:

26	6.781063	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
29	6.928154	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
32	7.153064	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
35	7.391262	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
38	7.927759	10.3.20.67	147.102.40.15	TELN...	56 Telnet Data ...

> Frame 26: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface \Device\NPF_{94774A54}

> Ethernet II, Src: IntelCor_1b:6c:ff (b4:69:21:1b:6c:ff), Dst: Fortinet_da:67:b0 (04:d5:90:da:67:b0)

> Internet Protocol Version 4, Src: 10.3.20.67, Dst: 147.102.40.15

> Transmission Control Protocol, Src Port: 50573, Dst Port: 23, Seq: 79, Ack: 126, Len: 1

▼ Telnet

Data: a

Προηγουμένως, ο υπολογιστής μας έχει ζητήσει την επανάληψη των χαρακτήρων από τον edu-dy.cn.ntua.gr (τεμάχιο 24: Do Echo).

24	0.427599	10.3.20.67	147.102.40.15	TELN...	57 Telnet Data ...
26	6.781063	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
29	6.928154	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
32	7.153064	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
35	7.391262	10.3.20.67	147.102.40.15	TELN...	55 Telnet Data ...
38	7.927759	10.3.20.67	147.102.40.15	TELN...	56 Telnet Data ...

> Frame 24: 57 bytes on wire (456 bits), 57 bytes captured (456 bits) on interface \Device\NPF_{94774A54}

> Ethernet II, Src: IntelCor_1b:6c:ff (b4:69:21:1b:6c:ff), Dst: Fortinet_da:67:b0 (04:d5:90:da:67:b0)

> Internet Protocol Version 4, Src: 10.3.20.67, Dst: 147.102.40.15

> Transmission Control Protocol, Src Port: 50573, Dst Port: 23, Seq: 76, Ack: 126, Len: 1

▼ Telnet

> Do Echo

1.10) Έχουμε την εξής εικόνα:

```
Wireshark - Follow TCP Stream (tcp.stream eq 0) - Lab8_log1_1.pcapng
..%..%..%.....%.....&.....#..'..$.&.....#..'..$.....x.....'.....ANSI.....".....!.....".....!.....
FreeBSD/amd64 (edu-dy.cn.ntua.gr) (pts/1)
login: ...aabbccdd
Password for abcd@edu-dy.cn.ntua.gr:efgh
Login incorrect
login:
```

Αμέσως μετά την προτροπή login (τεμάχιο 23), εμφανίζονται αρχικά 3 τελείες από τη μεριά μας (τεμάχιο 24), το οποίο στο Wireshark βλέπουμε πως μεταφράζεται σε Do Echo, δηλαδή ο υπολογιστής μας ζητάει από τον edu-dy.cn.ntua.gr να επαναλαμβάνει τους χαρακτήρες που λαμβάνει. Στη συνέχεια, βλέπουμε π.χ. την εισαγωγή του χαρακτήρα 'a' (κόκκινο χρώμα) εκ μέρους μας (τεμάχιο 26) και την εμφάνισή του επίσης στον σέρβερ (μπλε χρώμα). Το ίδιο συμβαίνει και για τους υπόλοιπους χαρακτήρες που εισάγουμε κατά το login, **δηλαδή τους πληκτρολογούμε και αυτοί εμφανίζονται επίσης στον edu-dy.cn.ntua.gr.**

1.11) Όσα παρατηρήσαμε, δικαιολογούνται, καθώς όπως είδαμε νωρίτερα, ο edu-dy.cn.ntua.gr προτίθεται να επαναλαμβάνει τους χαρακτήρες που του στέλνουμε και επιπλέον ο δικός μας υπολογιστής του έχει ζητήσει να το κάνει.

1.12) Εφαρμόζουμε το φίλτρο απεικόνισης: `ip.src==10.3.20.67 and ip.dst==147.102.40.15 and telnet`.

1.13) Χρειάζονται 4 πακέτα (υπ' αριθμόν 26, 29, 32, 35), ένα για κάθε χαρακτήρα.

1.14) Επίσης, για τον κωδικό efgh χρειάζονται επίσης 4 πακέτα (43, 45, 47, 49).

1.15) Όχι, ο εξυπηρετητής δε στέλνει την ηχώ των χαρακτήρων efgh του κωδικού χρήστη προς τον πελάτη.

1.16) Ενώ πριν την εισαγωγή των χαρακτήρων για το login, βλέπουμε πως ο υπολογιστής μας στέλνει Do Echo (τεμάχιο 24), δε παρατηρούμε κάποια εντολή Don't Echo πριν τη μεταφορά του κωδικού.

1.17) Υπάρχει περίπτωση ένα κακόβουλο λογισμικό (ή ακόμη και κάποιος άνθρωπος) να μπορεί να διαβάσει την οθόνη όσο εισάγεται ο κωδικός και να αποκτήσει πρόσβαση ενώ δε θα έπρεπε.

1.18) Το Telnet υστερεί από άποψη ασφαλείας, καθώς αρκεί κάποιος να μπορεί να “ακούει” την επικοινωνία μεταξύ 2 κόμβων για να υποκλέψει ευαίσθητα δεδομένα. Συγκεκριμένα, εφόσον η επικοινωνία δεν είναι κρυπτογραφημένη, με έναν αναλυτή πακέτων όπως το Wireshark και όπως είδαμε, είναι εύκολο να αναγνωστούν τα δεδομένα αυτά.

Άσκηση 2: FTP

2.1) Χρησιμοποιήθηκε το φίλτρο σύλληψης `host edu-dy.cn.ntua.gr`.

2.2) Το όρισμα `-d` ενεργοποιεί την αποσφαλμάτωση (`enables debugging`).

2.3) Το FTP πρωτόκολλο εφαρμογής χρησιμοποιεί το **TCP πρωτόκολλο μεταφοράς**.

2.4) Καταγράφονται οι εξής θύρες:

<u>Θύρα Πηγής</u>	<u>Θύρα Προορισμού</u>	<u>Αύξων Αριθμός Πακέτου</u>
56.356	21	1
21	56.356	2
20	56.357	32
56.357	20	33

Από τα παραπάνω γνωρίζουμε πως η θύρα ²¹ χρησιμοποιείται για τις εντολές ελέγχου, ενώ η θύρα 20 για τις εντολές δεδομένων (για ενεργό FTP τρόπο λειτουργίας).

2.5) Η TCP σύνδεση για τη μεταφορά δεδομένων γίνεται **από τον εξυπηρετητή προς τον πελάτη**.

2.6) Στάλθηκαν οι εξής εντολές FTP από τον πελάτη:

- **Request: OPTS UTF8 ON (packet 5)**
- **Request: USER anonymous (packet 8)**
- **Request: PASS [labuser@cn](#) (packet 11)**
- **Request: HELP (packet 14)**
- **Request: PORT 147,102,131,71,220,37**
- **Request: NLST**
- **Request: QUIT**

2.7) Όπως μπορούμε να δούμε παρακάτω, οι εντολές αυτές **εμφανίζονται στις πληροφορίες αποσφαλμάτωσης στην οθόνη του προγράμματος φλοιού ftp με ένα βέλος μπροστά τους**.

```
C:\Users\Άλεξ>ftp -d edu-dy.cn.ntua.gr
Connected to edu-dy.cn.ece.ntua.gr.
220 ProFTPD 1.3.4a Server (ProFTPD Default Installation) [147.102.40.15]
---> OPTS UTF8 ON
200 UTF8 set to on
User (edu-dy.cn.ece.ntua.gr:(none)): anonymous
---> USER anonymous
331 Anonymous login ok, send your complete email address as your password
Password:
---> PASS labuser@cn
230 Anonymous access granted, restrictions apply
```

```

ftp> help
Commands may be abbreviated.  Commands are:

!            delete          literal          prompt          send
?            debug           ls               put             status
append      dir              mdelete        pwd             trace
ascii       disconnect      mdir           quit            type
bell        get              mget           quote           user
binary      glob              mkdir          recv            verbose
bye         hash              mls            remotehelp
cd          help             mput           rename
close       lcd              open           rmdir

ftp> remotehelp
---> HELP
214-The following commands are recognized (* =>'s unimplemented):
214-CWD      XCWD      CDUP      XCUP      SMNT*    QUIT      PORT      PASV
214-EPRT     EPSV      ALLO*     RNFR      RNT0     DELE      MDTM      RMD
214-XRMD     MKD        XMKD      PWD       XPWD     SIZE      SYST      HELP
214-NOOP     FEAT       OPTS      AUTH*     CCC*     CONF*     ENC*      MIC*
214-PBSZ*    PROT*     TYPE      STRU      MODE     RETR      STOR      STOU
214-APPE     REST      ABOR      USER      PASS     ACCT*     REIN*     LIST
214-NLST     STAT      SITE      MLSD      MLST
214 Direct comments to root@edu-dy.cn.ece.ntua.gr

```

```

ftp> ls
---> PORT 147,102,131,71,220,37
200 PORT command successful
---> NLST
150 Opening ASCII mode data connection for file list
FreeBSD10.4.ova
PCATTCP.exe
lab6.cap
router.ova
FreeBSD.ova
firewall.ova
MagicAdb.exe
Asterisk.ova
TDIQ.exe
MacAddr2.exe
putty.exe
FreeBSD11.3.ova
psftp.exe
pcattcp.pcap
icmpv6.pcap
226 Transfer complete
ftp: 200 bytes received in 0.01Seconds 15.38Kbytes/sec.
ftp> bye
---> QUIT
221 Goodbye.

```

2.8) Με την εντολή **USER**.

2.9) Απαιτείται **ένα πακέτο** (το 8 συγκεκριμένα, όπως φαίνεται παρακάτω).

8 3.426143	147.102.131.71	147.102.40.15	FTP	70 Request: USER anonymous
> Frame 8: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface \Device\NPF_{BA25E72C-F2F0-4548-80D4-0C718D8B9A53}, id 0 > Ethernet II, Src: 00:ff:ba:25:e7:2c (00:ff:ba:25:e7:2c), Dst: 00:ff:bb:25:e7:2c (00:ff:bb:25:e7:2c) > Internet Protocol Version 4, Src: 147.102.131.71, Dst: 147.102.40.15 > Transmission Control Protocol, Src Port: 56356, Dst Port: 21, Seq: 15, Ack: 95, Len: 16 > File Transfer Protocol (FTP) > USER anonymous\r\n Request command: USER Request arg: anonymous [Current working directory:]				

2.10) Με την εντολή **PASS**.

2.11) Χρειάζεται επίσης **ένα μόνο IPv4 πακέτο** για να μεταφερθεί ο κωδικός.

2.12) Αναφορικά με τη μεταφορά ονόματος/κωδικού με τα πρωτόκολλα TELNET και FTP παρατηρούμε πως **ενώ το πρώτο απαιτεί ένα τεμάχιο για κάθε χαρακτήρα του ονόματος/κωδικού, το ftp στέλνει ολόκληρο το όνομα/κωδικό σε ένα πακέτο**. Αυτό που έχουν κοινό είναι πως και στο FTP αλλά και στο TELNET όπως είδαμε πριν, οι πληροφορίες αυτές δε στέλνονται κρυπτογραφημένες.

2.13) Όπως παρατηρούμε, η εντολή help του προγράμματος φλοιού **δε μεταφράζεται σε εντολή του πρωτοκόλλου FTP**. Ωστόσο, αυτή που μεταφράζεται είναι η εντολή remotehelp, η οποία και μεταφράζεται στην εντολή HELP.

```
ftp> help
Commands may be abbreviated.  Commands are:

!                delete          literal          prompt          send
?                debug           ls              put             status
append          dir              mdelete        pwd            trace
ascii          disconnect     mdir           quit           type
bell           get           mget          quote          user
binary         glob          mkdir         recv           verbose
bye            hash          mls           remotehelp
cd             help          mput          rename
close          lcd           open          rmdir
ftp> remotehelp
---> HELP
```

2.14) Δύο εντολές FTP που δεν υποστηρίζονται από τον FTP εξυπηρετητή είναι οι **AUTH, CCC**.

2.15) Όπως βλέπουμε, **ο υπολογιστής μας έστειλε 1, ενώ ο εξυπηρετητής 9 πακέτα σχετικά με την εντολή remotehelp**.

14	16.034904	147.102.131.71	147.102.40.15	FTP	60	Request: HELP							
15	16.040693	147.102.40.15	147.102.131.71	FTP	121	Response: 214-The following commands are recognized (* =>'s unimplemented):							
17	16.096967	147.102.40.15	147.102.131.71	FTP	124	Response: 214-CWD	XCWD	CDUP	XCUP	SMNT*	QUIT	PORT	PASV
18	16.097338	147.102.40.15	147.102.131.71	FTP	124	Response: 214-EPRT	EPSV	ALLO*	RNFR	RNTO	DELE	MDTM	RMD
20	16.097474	147.102.40.15	147.102.131.71	FTP	124	Response: 214-XRMD	MKD	XMKD	PWD	XPWD	SIZE	SYST	HELP
21	16.097636	147.102.40.15	147.102.131.71	FTP	124	Response: 214-NOOP	FEAT	OPTS	AUTH*	CCC*	CONF*	ENC*	MIC*
23	16.097726	147.102.40.15	147.102.131.71	FTP	124	Response: 214-PBSZ*	PROT*	TYPE	STRU	MODE	RETR	STOR	STOU
24	16.097842	147.102.40.15	147.102.131.71	FTP	124	Response: 214-APPE	REST	ABOR	USER	PASS	ACCT*	REIN*	LIST
26	16.097942	147.102.40.15	147.102.131.71	FTP	100	Response: 214-NLST	STAT	SITE	MLSD	MLST			
27	16.098048	147.102.40.15	147.102.131.71	FTP	105	Response: 214 Direct comments to root@edu-dy.cn.ece.ntua.gr							

2.16) Βλέποντας το παραπάνω στιγμιότυπο, το πρώτο μήνυμα (πακέτο 15) από τον εξυπηρετητή περιλαμβάνει το μήνυμα “214-The following commands are recognized...”. Ο εξυπηρετητής, δηλώνει πως τελείωσε η αποστολή πακέτων στέλνοντας ένα πακέτο, το μήνυμα του οποίου ξεκινάει με τον ίδιο κωδικό (214 εν προκειμένω), ακολουθείται από κενό και έχει ενδεχομένως κάποιο κείμενο, όπως και επαληθεύεται παραπάνω (πακέτο 27).

2.17) Την IP του υπολογιστή μας.

29	19.276282	147.102.131.71	147.102.40.15	FTP	82	Request: PORT 147,102,131,71,220,37
----	-----------	----------------	---------------	-----	----	-------------------------------------

2.18) Στο ερώτημα 2.4 βρήκαμε πως ο υπολογιστής μας δέχεται δεδομένα στη θύρα 56.357. Αυτό, προκύπτει από τους τελευταίους δεκαδικούς αριθμούς ως εξής: Πολλαπλασιάζουμε τον πρώτο από τους 2 με 256 και προσθέτουμε τον δεύτερο. Άρα, στην περίπτωση μας: $220 * 256 + 37 = 56.357$.

2.19) Είδαμε τα αρχεία του τρέχοντος καταλόγου με την εντολή φλοιού ls, η οποία αντιστοιχεί στην εντολή πρωτοκόλλου FTP: NLST.

2.20) Διότι, όπως βλέπουμε, ο υπολογιστής μας λέει πριν το NLST, ότι ακούει για δεδομένα στο PORT 56.357.

2.21) Η bye μεταφράζεται στην QUIT.

2.22) Ο εξυπηρετητής αποκρίνεται στο Request: QUIT με Response: 221 Goodbye.

2.23) Φίλτρο απεικόνισης: `tcp.flags.fin==1`.

2.24) Παρατηρούμε πως η απόλυση των συνδέσεων έγινε από την πλευρά του σέρβερ όσον αφορά τα μηνύματα δεδομένων (πακέτο 36) και από την πλευρά του πελάτη όσον αφορά τις εντολές ελέγχου FTP (πακέτο 45).

No.	Time	Source	Destination	Protocol	Length	Info
36	19.345341	147.102.40.15	147.102.131.71	FTP-...	251	FTP Data: 197 bytes (PORT) (NLST)
38	19.365340	147.102.131.71	147.102.40.15	TCP	54	56357 → 20 [FIN, ACK] Seq=1 Ack=199 Win=131072 Len=0
45	24.355302	147.102.131.71	147.102.40.15	TCP	54	56356 → 21 [FIN, ACK] Seq=94 Ack=924 Win=7835 Len=0
46	24.360768	147.102.40.15	147.102.131.71	TCP	54	21 → 56356 [FIN, ACK] Seq=924 Ack=94 Win=65920 Len=0
48	24.364327	147.102.40.15	147.102.131.71	TCP	54	[TCP Out-Of-Order] 21 → 56356 [FIN, ACK] Seq=924 Ack=95 Win=65920 Len=0

2.25) Με το φίλτρο απεικόνισης που φαίνεται στο παρακάτω στιγμιότυπο, βλέπουμε τις θύρες πηγής/προορισμού 57.170/21 για τις εντολές ελέγχου και τις θύρες πηγής/προορισμού 57.171/59.203 για τη μεταφορά δεδομένων.

No.	Time	Source	Destination	Protocol	Length	Info
386	7.373657	147.102.131.71	147.102.40.15	TCP	66	57170 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
389	7.431788	147.102.40.15	147.102.131.71	TCP	66	21 → 57170 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=536 WS=64 SACK_PERM=1
441	7.703335	147.102.131.71	147.102.40.15	TCP	66	57171 → 59203 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
443	7.722909	147.102.40.15	147.102.131.71	TCP	66	59203 → 57171 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=536 WS=64 SACK_PERM=1

2.26) Παρατηρούμε τις παρακάτω εντολές:

- Request: USER anonymous
- Request: PASS IEUser@
- Request: opts utf8 on

- **Request: syst**
- **Request: site help**
- **Request: PWD**
- **Request: TYPE A**
- **Request: PASV**
- **Request: LIST**

2.27) Στην περίπτωση μας, χρησιμοποιήθηκε το **όνομα χρήστη anonymous** και ο κωδικός χρήστη **IEUser@**.

2.28) Για την εμφάνιση της λίστας αρχείων, χρησιμοποιήθηκε η εντολή FTP πρωτοκόλλου **LIST**.

2.29) Εφαρμόζουμε το φίλτρο **ftp.response** και βλέπουμε τα αιτήματα του πελάτη και τις αποκρίσεις του εξυπηρετητή.

No.	Time	Source	Destination	Protocol	Length	Info
394	7.492034	147.102.40.15	147.102.131.71	FTP	128	Response: 220 ProFTPD 1.3.4a Server (ProFTPD Default Installation) [147.102.40.15]
398	7.492452	147.102.131.71	147.102.40.15	FTP	70	Request: USER anonymous
401	7.569996	147.102.40.15	147.102.131.71	FTP	129	Response: 331 Anonymous login ok, send your complete email address as your password
404	7.570222	147.102.131.71	147.102.40.15	FTP	68	Request: PASS IEUser@
407	7.582671	147.102.40.15	147.102.131.71	FTP	104	Response: 230 Anonymous access granted, restrictions apply
409	7.582857	147.102.131.71	147.102.40.15	FTP	68	Request: opts utf8 on
411	7.600865	147.102.40.15	147.102.131.71	FTP	74	Response: 200 UTF8 set to on
413	7.601079	147.102.131.71	147.102.40.15	FTP	60	Request: syst
415	7.621172	147.102.40.15	147.102.131.71	FTP	73	Response: 215 UNIX Type: L8
417	7.621406	147.102.131.71	147.102.40.15	FTP	65	Request: site help
419	7.640366	147.102.40.15	147.102.131.71	FTP	125	Response: 214-The following SITE commands are recognized (* =>'s unimplemented)
421	7.643479	147.102.40.15	147.102.131.71	FTP	64	Response: 214-HELP
423	7.643580	147.102.40.15	147.102.131.71	FTP	65	Response: 214-CHGRP
425	7.643634	147.102.40.15	147.102.131.71	FTP	65	Response: 214-CHMOD
427	7.643686	147.102.40.15	147.102.131.71	FTP	105	Response: 214 Direct comments to root@edu-dy.cn.ece.ntua.gr
429	7.643865	147.102.131.71	147.102.40.15	FTP	59	Request: PWD
431	7.660669	147.102.40.15	147.102.131.71	FTP	88	Response: 257 "/" is the current directory
433	7.660930	147.102.131.71	147.102.40.15	FTP	62	Request: TYPE A
435	7.681008	147.102.40.15	147.102.131.71	FTP	73	Response: 200 Type set to A
437	7.681530	147.102.131.71	147.102.40.15	FTP	60	Request: PASV
439	7.702837	147.102.40.15	147.102.131.71	FTP	105	Response: 227 Entering Passive Mode (147,102,40,15,231,67).
445	7.723092	147.102.131.71	147.102.40.15	FTP	60	Request: LIST
447	7.741075	147.102.40.15	147.102.131.71	FTP	108	Response: 150 Opening ASCII mode data connection for file list
456	7.821241	147.102.40.15	147.102.131.71	FTP	77	Response: 226 Transfer complete

Βλέπουμε ότι ο εξυπηρετητής απαντά με **Response: 227 Entering Passive Mode (147,102,40,15,231,67)**.

2.30) Η εγκατάσταση σύνδεσης TCP που αφορούν τα μηνύματα δεδομένων FTP γίνεται από την πλευρά του **πελάτη**.

2.31) Για τη μεταφορά δεδομένων FTP, ο εξυπηρετητής χρησιμοποιεί τη **θύρα 59.203 για τη μεταφορά δεδομένων**. Παρατηρώντας την απόκριση στο 2.29, ο αριθμός αυτός προκύπτει από τους 2 τελευταίους δεκαδικούς αριθμούς που εμφανίζονται στην απόκριση (231,67) ως εξής: **231 * 256 + 67 = 59.203**.

2.32) Αντίστοιχα, από την πλευρά του πελάτη, η θύρα 57.171 που χρησιμοποιείται για τη μεταφορά δεδομένων **προκύπτει ως η αμέσως επόμενη της θύρας που χρησιμοποιήθηκε για τη σύνδεση ελέγχου (57.170)**.

2.33) Όπως βλέπουμε, στάλθηκαν 2 μηνύματα FTP δεδομένων.

ftp-data						
No.	Time	Source	Destination	Protocol	Length	Info
449	7.745373	147.102.40.15	147.102.131.71	FTP-...	590	FTP Data: 536 bytes (PASV) (LIST)
451	7.745472	147.102.40.15	147.102.131.71	FTP-...	544	FTP Data: 490 bytes (PASV) (LIST)

2.34) Γνωρίζουμε (από προηγούμενες ασκήσεις) πως ο σέρβερ 147.102.40.15 έχει MTU 576 bytes (άρα συνολικά με την προσθήκη του Ethernet Header 590 bytes).

Για τα παρακάτω 2 ερωτήματα, εφαρμόζουμε φίλτρο **tcp.flags.fin==1** και παίρνουμε τα εξής αποτελέσματα.

tcp.flags.fin==1						
No.	Time	Source	Destination	Protocol	Length	Info
56	8.768918	147.102.40.15	147.102.131.71	FTP-...	544	FTP Data: 490 bytes (PASV) (TYPE A)
58	8.769112	147.102.131.71	147.102.40.15	TCP	54	59648 → 42843 [FIN, ACK] Seq=1 Ack=1028 Win=261632 Len=0
64	11.688965	147.102.131.71	147.102.40.15	TCP	54	59647 → 21 [FIN, ACK] Seq=87 Ack=574 Win=261888 Len=0
66	11.752895	147.102.40.15	147.102.131.71	TCP	54	21 → 59647 [FIN, ACK] Seq=574 Ack=88 Win=65920 Len=0

2.35) Η απόλυση TCP συνδέσεων που αφορούν εντολές ελέγχου, γίνεται από τον πελάτη.

2.36) Η απόλυση TCP συνδέσεων που αφορούν μηνύματα δεδομένων γίνεται από τον εξυπηρετητή.

Άσκηση 3: TFTP

3.1) Το TFTP χρησιμοποιεί το πρωτόκολλο μεταφοράς UDP.

3.2) Για την πρώτη επικοινωνία πελάτη-εξυπηρετητή TFTP: **Θύρα πηγής: 49.671** και **Θύρα προορισμού: 69**.

3.3) Κατά τη μεταφορά δεδομένων, έχουμε **Θύρα πελάτη: 49.671** και **Θύρα εξυπηρετητή: 34.721**.

3.4) Η **θύρα 69** αντιστοιχεί στο πρωτόκολλο TFTP.

3.5) Σχετικά με τους αριθμούς θυρών γνωρίζουμε τα εξής. Προκειμένου να δημιουργηθεί μια σύνδεση, κάθε άκρο επιλέγει ένα Transfer Identifier (TID), το οποίο και θα χρησιμοποιείται κατά τη διάρκεια της σύνδεσης. Το κάθε άκρο της επικοινωνίας αυτής επιλέγει **τυχαία μία από τις διαθέσιμες θύρες**, έτσι ώστε να μειωθεί στο ελάχιστο η πιθανότητα τα 2 άκρα να επέλεξαν ίδια θύρα. Κάθε πακέτο που μεταδίδεται κατά τη σύνδεση αυτή φέρει και τα 2 TID των τερματικών της σύνδεσης, τα οποία και δίνει στο UDP πρωτόκολλο ως Source και Destination Port.

Ο κόμβος που κάνει την αρχική αίτηση (εν προκειμένω ο δικός μας, ο οποίος στέλνει RRQ – Read Request), έχει επιλέξει τυχαία τη θύρα που θα χρησιμοποιήσει και στέλνει το αρχικό αίτημα στη θύρα 69₁₀ στον εξυπηρετητή. Με τη σειρά του, ο σέρβερ αποκρίνεται, υπό κανονικές συνθήκες με το TID που εκείνος επέλεξε και που διατηρεί για το υπόλοιπο της σύνδεσης.

3.6) Το αρχείο rfc1350.txt μεταφέρεται με **ASCII**.

3.7) Ο τρόπος μεταφοράς καθορίζεται στο **πρώτο πακέτο** και ειδικότερα στο πεδίο **Type** της επικεφαλίδας **TFTP**.

```
▼ Trivial File Transfer Protocol
  Opcode: Read Request (1)
  Source File: rfc1350.txt
  Type: netascii
```

3.8) Καταγράφηκαν οι εξής τύποι TFTP μηνυμάτων:

- **Opcode: Read Request (1)**
- **Opcode: Data Packet (3)**
- **Opcode: Acknowledgment (4)**

3.9) Ενώ το UDP είναι αναξιόπιστο λόγω έλλειψης μηχανισμού επιβεβαιώσεων, το TFTP λύνει αυτό το πρόβλημα, καθώς για κάθε πακέτο που λαμβάνεται με έναν συγκεκριμένο (αύξοντα) αριθμό Block από το ένα άκρο, **στέλνεται και ένα TFTP μήνυμα τύπου Acknowledgment για το Block από το άλλο άκρο με τον ίδιο αριθμό προκειμένου να σιγουρευτούμε πως ολοκληρώθηκε επιτυχώς η μεταφορά.**

3.10) Όπως είπαμε, χρησιμοποιούνται τα μηνύματα τύπου **Acknowledgment**.

3.11) Κάθε μήνυμα TFTP που μεταφέρει δεδομένα από τον σέρβερ σε εμάς (πλην του τελευταίου) έχει μέγεθος **516 bytes** (αφορά το μέγεθος της επικεφαλίδας TFTP και των δεδομένων TFTP, το συνολικό μέγεθος του πακέτου είναι 558 bytes).

3.12) Από αυτά, δεδομένα είναι τα **512 bytes**.

3.13) Ο πελάτης αντιλαμβάνεται το τέλος της μετάδοσης δεδομένων όταν λαμβάνει πακέτο με **δεδομένα μεγέθους το πολύ έως 511 bytes**.