Όνοματεπώνυμο: Μάρκος Δεληγιάννης	Ομάδα: 3
Όνομα PC/ΛΣ: LAPTOP-THP9FHNU / W11	Ημερομηνία: 30 / 11 / 2022
Διεύθυνση IP: 147.102.131.119	Διεύθυνση MAC: 00:ff:c9:b7:f8:72

Εργαστηριακή Άσκηση 8
TELNET, FTP και TFTP

Απαντήστε στα ερωτήματα στον χώρο που σας δίνεται παρακάτω και στην πίσω σελίδα εάν δεν επαρκεί. Το φυλλάδιο αυτό θα παραδοθεί στον επιβλέποντα.

1

1.1 TCP.

1.2 Εξυπηρετητής: 23 PC: 54444

1.3 Η θύρα 23.

1.4 “telnet”

1.5 Server: Do echo
PC: Will echo
Server: Don’t echo / Will echo
PC: Won’t echo

1.6 Ναι και ο υπολογιστής δέχεται

1.7 Επίσης ναι και ο υπολογιστής δέχεται.

1.8 Ναι

1.9 Ναι, έχει προηγηθεί.

1.10 Κάθε χαρακτήρας ASCII που αποστέλλεται γίνεται Echoed από τον εξυπηρετητή.

1.11 Αυτή η συμπεριφορά οφείλεται στο ότι ο server είχε στείλει μήνυμα Will echo (1.8), δηλώνοντας την πρόθεση να κάνει echo, και εμείς δεχθήκαμε με την εντολή Do echo (1.9).

1.12 “telnet and ip.dst_host == 147.102.40.15”

1.13 5 πακέτα (a, b, c, d, \r\n).

1.14 5 πακέτα (e, f, g, h, \r\n).

1.15 Όχι, δεν στέλνει.

1.16 Όχι, δεν παρατηρήσαμε.

1.17 Δεν εμφανίζεται στην οθόνη διότι ο εξυπηρετητής γνωρίζει ότι πρόκειται για κωδικό χρήστη και δεν εκτελεί Echo (αυτόματα, χωρίς προτροπή), αφού αυτό θα ενείχε τον κίνδυνο υποκλοπής χωρίς κανένα όφελος.

1.18 Το Telnet δεν κρυπτογραφεί τα δεδομένα που αποστέλλονται. Έτσι, κάποιος θα μπορούσε εύκολα αν έχει πρόσβαση στα δεδομένα που αποστέλλονται, με χρήση ενός αναλυτή πρωτοκόλλων, να δει τα login credentials του χρήστη και όλα τα δεδομένα που ανταλλάσονται μεταξύ αυτού και του server. Το Telnet συνεπώς δεν παρέχει καμία σημαντική ασφάλεια δεδομένων.

2

2.1 “host 147.102.40.15”

2.2 Σημαίνει debugging. Με αυτή τη σημαία εμφανίζονται στον φλοιό οι εντολές FTP που στέλνονται.

2.3 Tcp

2.4 Εντολές ελέγχου: Θύρα πηγής: 55114 (PC) Θύρα προορισμού: 21 (Server)
Εντολές δεδομένων: Θύρα πηγής: 20 (Server) Θύρα προορισμού: 55115 (PC)

2.5 Από τη πλευρά του εξυπηρετητή.

2.6 Οι εντολές είναι: "**OPTS** UTF8 ON", "**USER** anonymous", "**PASS** labuser@cn", "**HELP**", (κύριο τμήμα με bold) "**PORT** 147, 102, 238, 63, 215, 75", "**NLST**", "**QUIT**".

2.7 Ναι, εμφανίζονται στο terminal κάθε φορά με την εξής μορφή: ---> [command]

2.8 Με την εντολή USER.

2.9 Μόνο ένα.

2.10 Με την εντολή PASS.

2.11 Μόνο ένα.

2.12 Ομοιότητα: Έλλειψη κρυπτογραφίας
Διαφορά: Το TELNET στέλνει πολλά μηνύματα για τη μεταφορά του ονόματος και κωδικού, ενώ το ftp στέλνει μόνο ένα για το καθένα.

2.13 Όχι, διότι η εντολή ζητά τις εντολές που δέχεται ο φλοιός και συνεπώς αφορά το client πρόγραμμα.

2.14 Οι εντολές ALLO και AUTH, αφού έχουν *.

2.15 Ο υπολογιστής μας έστειλε 1 πακέτο και ο εξυπηρετητής 9 πακέτα.

2.16 Αποστέλλοντας ένα μήνυμα ftp με τον τριψήφιο κωδικό της απάντησης, **κενό αντί για –** και κείμενο.

2.17 Οι πρώτοι 4 αριθμοί της εντολής PORT αντιστοιχούν στην IPv4 διεύθυνση της διεπαφής του υπολογιστή μας (από το πιο σημαντικό προς το λιγότερο σημαντικό byte).

2.18 Οι δύο αριθμοί αυτοί είναι η δεκαδική αναπαράσταση του πιο σημαντικού και λιγότερο σημαντικού byte, αντίστοιχα, της θύρας. Έτσι είναι: αριθμός θύρας = 256*[5ος αριθμός]+[6ος αριθμός].

2.19 Η εντολή NLST.

2.20 Η εντολή NLST χρειάζεται μία εγκατεστημένη σύνδεση δεδομένων για την εκτέλεσή της. Συνεπώς όταν εκτελούμε ls στον φλοιό αποστέλλεται πρώτα η FTP εντολή PORT ώστε να ενημερώσει τον server πού πρέπει να συνδεθεί ώστε να εγκατασταθεί η απαραίτητη σύνδεση δεδομένων.

2.21 Στην εντολή QUIT.

2.22 "221 Goodbye.\r\n"

2.23 "tcp.flags.fin == 1"

2.24 Η απόλυση και των δύο συνδέσεων γίνεται από την πλευρά του εξυπηρετητή.

2.25 Σύνδεση ελέγχου: Client: 55116 Server: 21
Σύνδεση δεδομένων: Client: 55117 Server: 13634

2.26 Οι εντολές είναι: **USER** anonymous, **PASS** IEUser@, **opts** utf8 on, **syst**, **site** help, (κύριο τμήμα με bold) **PWD**, **noop**, **CWD** /, **TYPE** A, **PASV**, **LIST**.

2.27 User: anonymous Password: IEUser@

.....

2.28 Η εντολή LIST.

2.29 “227 Entering Passive Mode (147,102,40,15,53,66).\r\n”

2.30 Από την πλευρά του πελάτη.

2.31 Χρησιμοποιεί τη θύρα 13634 = 256 *** 53** + **66**.

.....

2.32 Προκύπτει ως η ακριβώς επόμενη (+1) της θύρας που χρησιμοποιήθηκε για τη σύνδεση ελέγχου.

.....

2.33 Στάλθηκαν 3 μηνύματα δεδομένων FTP με μέγεθος δεδομένων 536, 536 και 307 bytes αντίστοιχα.

.....

2.34 Το ελάχιστο των MSS κατά την τριπλή χειραψία TCP για τη σύνδεση δεδομένων είναι 536, το οποίο ταυτίζεται με το μέγεθος δεδομένων των πρώτων δύο μηνυμάτων FTP.

2.35 Από την πλευρά του πελάτη.

2.36 Από την πλευρά του εξυπηρετητή.

3

3.1 Το πρωτόκολλο UDP.

3.2 Θύρα πηγής (Client): 56395 Θύρα προορισμού (Server): 69

3.3 Θύρα πηγής (Server): 15245 Θύρα προορισμού (Client): 56395

3.4 Η θύρα 69.

3.5 Ο client επιλέγει μία τυχαία θύρα για τον εαυτό του (εδώ 56395) και αποστέλλει το μήνυμα στη θύρα 69 του server. Ο server απαντά στη θύρα του χρήστη (56395) από μία τυχαία θύρα που έχει επιλέξει ο ίδιος (εδώ 15245). Η υπολοιπη επικοινωνία γίνεται μεταξύ των δύο αυτών θυρών.

3.6 Με ASCII τρόπο.

3.7 Καθορίζεται στο 1ο μήνυμα (πελάτη προς εξυπηρετητή) → Type = netascii

.....

3.8 Read Request (opcode 1), Data (opcode 3) και Acknowledgment (opcode 4).

3.9 Το πρόβλημα αντιμετωπίζεται με ξεχωριστά acknowledgment μηνύματα TFTP για κάθε μήνυμα που φέρει πληροφορία. Αν κάποιο μήνυμα έχει λάθη ή δεν έχει φτάσει στον προορισμό του τότε ο παραλήπτης δεν στέλνει κανένα μήνυμα και ο αποστολέας μετά από μία χρονική περίοδο (timeout) το ξαναστέλνει (εκτός αν είναι error message).

3.10 Το μήνυμα τύπου “Acknowledgement” με πεδία επικεφαλίδας το opcode: 4 και το block: [αριθμός του block δεδομένων που επιβεβαιώνεται].

3.11 516 bytes.

3.12 512 bytes.

3.13 Κάθε μήνυμα δεδομένων TFTP έχει μέγεθος δεδομένων ακριβώς 512 bytes, **εκτός και αν είναι το τελευταίο**. Συνεπώς, για τον έλεγχο του τέλους της μετάδοσης δεδομένων αρκεί να ελεγχθεί αν το μήκος του μηνύματος (χωρίς την επικεφαλίδα TFTP) είναι μικρότερο από 512 bytes.