Ονοματεπώνυμο: ΔΗΜΗΤΡΙΟΣ ΓΕΩΡΓΟΥΣΗΣ	**ΑΜ: 03119005**	**Ομάδα: 4**
Όνομα PC/ΛΣ: dimitris-Laptop / WINDOWS 11 HOME		**Ημερομηνία: 26/10/2022**
Διεύθυνση IP: 192.168.2.6	**Διεύθυνση MAC: 7C-8A-E1-C3-47-5C**	

Άσκηση 1:

1.1: Σύνταξη εντολής που χρησιμοποίησα: ping -n 3 -4 www.mit.edu

Θέλουμε να παραχθούν 3 πακέτα IPv4/ICMP, δηλαδή, να στείλουμε 3 τέτοια πακέτα. Οπότε χρησιμοποιούμε το -n 3 option και το -4 option είναι για να αναγκάσουμε το σύστημα να έχει μόνο IPv4/ICMP πακέτα.

1.2: Εφαρμόσαμε το capture filter: “not multicast and not broadcast”
Σημασία του φίλτρου αυτού: Συλλαμβάνουμε μόνο ατομική κίνηση – ξεφορτωνόμαστε τον «θόρυβο» του δικτύου, αφού θέλουμε να δούμε μόνο την κίνηση από και προς την δική μας κάρτα δικτύου.

PS C:\Users\dimig> ping -n 3 -4 www.mit.edu

Pinging e9566.dscb.akamaiedge.net [104.76.158.49] with 32 bytes of data:
Reply from 104.76.158.49: bytes=32 time=39ms TTL=50
Reply from 104.76.158.49: bytes=32 time=40ms TTL=50
Reply from 104.76.158.49: bytes=32 time=40ms TTL=50

Ping statistics for 104.76.158.49:
Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 39ms, Maximum = 40ms, Average = 39ms

1.3: Ποσοστό απωλειών: 0%
Μέση καθυστέρηση: 39ms

1.4: RTT1 = 39ms
RTT2 = 40ms
RTT3 = 40ms

1.5: RTT1 = 39.665ms
RTT2 = 40.078ms
RTT3 = 40.160ms
Minimum = RTT1, Maximum = RTT3, Average = (RTT1 + RTT2 + RTT3)/3 = 39.968ms
Οι τιμές συμφωνούν με αυτές που έβγαλε το παράθυρο εντολών.

1.6: Το φίλτρο απεικόνισης που εφαρμόζουμε είναι το “ip”.

1.7: Η εντολή ping προκαλεί μόνο request/reply ζεύγη άρα θα εφαρμόσουμε το φίλτρο:
“icmp.type == 8 or icmp.type == 0”, γιατί 8 → request και 0 → reply.

1.8: Από το πεδίο type των μηνυμάτων ICMP που έστειλε ο υπολογιστής μου βλέπουμε ότι στάλθηκαν μόνο πακέτα τύπου request.

1.9: IPv4 Source = 192.168.2.6
IPv4 Destination = 104.76.158.49

1.10: Ελήφθησαν πακέτα ICMP τύπου reply.

1.11: IPv4 Source = 104.76.158.49
IPv4 Destination = 192.168.2.6

1.12: Παρελθόν:
Pinging www.mit.edu [18.7.22.83] with 32 bytes of data:

Τώρα:
Pinging e9566.dscb.akamaiedge.net [104.76.158.49] with 32 bytes of data:

Έχει αλλάξει και η IP address και το όνομα της διεύθυνσης.

Άσκηση 2:

PS C:\Users\dimig> ping -n 5 -4 192.168.2.2; ping -n 5 -4 192.168.2.6; ping -n 5 -4 127.0.0.1

192.168.2.2 είναι το κινητό μου
192.168.2.6 είναι η κάρτα δικτύου μου
127.0.0.1 είναι ο βρόχος επιστροφής

2.1: Για κάθε ping χρησιμοποιήσαμε την σύνταξη ping -n 5 -4 <address>

2.2: Το wireshark έχει καταγράψει μόνο 5 μηνύματα ICMP Echo request.

2.3: Ο προορισμός τους ήταν το κινητό μου (192.168.2.2).

2.4: Όχι, δεν παρατηρούμε την αποστολή τέτοιων μηνυμάτων στο Wireshark. Ο προορισμός των πακέτων αυτών είναι η τοπική διεύθυνση IPv4 του υπολογιστή μας. Σύμφωνα και με το σχήμα της εκφώνησης του ερωτήματος τα πακέτα αυτά οδηγούνται στον υπολογιστή μας μέσω του οδηγού loopback και δεν θα περάσουν από την ουρά εισόδου IPv4 του οδηγού Ethernet, άρα δεν θα τα καταγράψει το Wireshark.

2.5: Όχι, για τον ίδιο λόγο με πριν.

2.6:
ping <διεύθυνση loopback>: Το πακέτο εξέρχεται από τον υπολογιστή, μπαίνει στον Οδηγό loopback και επιστρέφει στον υπολογιστή.

ping <διεπαφή υπολογιστή>: Το πακέτο εξέρχεται από τον υπολογιστή, έπειτα μπαίνει στον Οδηγό Ethernet, ο οποίος το στέλνει στον Οδηγό loopback και έπειτα επιστρέφει στον υπολογιστή μας.

```
PS C:\Users\dimig> ping www.netflix.com

Pinging apiproxy-website-nlb-prod-3-ac110f6ae472b85a.elb.eu-west-1.amazonaws.com [3.251.50.149]
with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 3.251.50.149:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PS C:\Users\dimig> ping www.amazon.com

Pinging d3ag4hukkh62yn.cloudfront.net [52.85.159.175] with 32 bytes of data:
Reply from 52.85.159.175: bytes=32 time=9ms TTL=244
Reply from 52.85.159.175: bytes=32 time=10ms TTL=244
Reply from 52.85.159.175: bytes=32 time=10ms TTL=244
Reply from 52.85.159.175: bytes=32 time=10ms TTL=244

Ping statistics for 52.85.159.175:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 10ms, Average = 9ms
```

2.7: Σε αντίθεση με το amazon δεν υπάρχουν ping replies στο netflix. Επίσης, το netflix χρησιμοποιεί το amazon aws το οποίο είναι το cloud platform της amazon, ενώ το amazon χρησιμοποιεί το cloudfront το οποίο είναι επίσης υπηρεσία της amazon.

Το netflix δεν απαντά στα pings μας, οπότε είναι πιθανό να παρεμβάλλεται στη διαδρομή κάποιο τείχος προστασίας που μπλοκάρει τα μηνύματα πρωτοκόλλου ICMP.

Άσκηση 3:

3.1: Capture filter: “host 147.102.40.15”

3.2: Φίλτρο για να παραμείνουν μόνο τα πακέτα IPv4 που έστειλε ο υπολογιστής μου:
“ip.src==192.168.2.6”

3.3:

Version (4 bit)
(Internet) Header Length (4 bit)
Differentiated Service Field (or Type of Service) (8 bit)
Total Length (16 bit)
Identification (16 bit)
Flags (3 bit)
Fragment Offset (13 bit)
TTL (Time to Live) (8 bit)
Protocol (8 bit)
Header Checksum (16 bit)
Source Address (32 bit)
Destination Address (32 bit)

<table>
<tr><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>20</th><th>21</th><th>22</th><th>23</th><th>24</th><th>25</th><th>26</th><th>27</th><th>28</th><th>29</th><th>30</th><th>31</th></tr>
<tr><td colspan="4">Version</td><td colspan="4">HL</td><td colspan="6">DSF</td><td colspan="16">Total Length</td></tr>
<tr><td colspan="16">Identification</td><td colspan="3">Flags</td><td colspan="13">Fragment Offset</td></tr>
<tr><td colspan="8">Time To Live</td><td colspan="8">Protocol</td><td colspan="16">Header Checksum</td></tr>
<tr><td colspan="32">Source Address</td></tr>
<tr><td colspan="32">Destination Address</td></tr>
</table>

3.4: Τα πεδία Total Length και Identification αλλάζουν τιμές.

3.5: Ναι. Είναι 20 bytes.

3.6: Κάνουμε sort by length στο Wireshark. Το μικρότερο μήκος είναι 40 bytes και το μεγαλύτερο μήκος είναι 65 bytes.

3.7: Differentiated Services Field = 0x00. Αντιστοιχεί στην Standard ποιότητα υπηρεσίας (CS0)

3.8: Οι τιμές του πεδίου Identification αυξάνονται κατά 1 καθώς κινούμαστε από το πρώτο μήνυμα που έστειλε ο υπολογιστής μας προς το τελευταίο.

3.9: To Don’t Fragment bit είναι το 2^{ο} bit των flags. Έχει τιμή 1.

3.10: Fragment Offset = 0

3.11: Protocol = 6, αντιστοιχεί στο TCP.

3.12: Το πεδίο Header Checksum είναι “a 16 bit one’s compliment checksum of the IP header” (από το network sorcery). Συνεπώς, αφού το checksum προκύπτει από αθροίσματα 16 bit «ομάδων» της επικεφαλίδας (και το συμπλήρωμα ως προς 1 του τελικού αθροίσματος) η τιμή του αλλάζει αν αλλάζουν τα bit της επικεφαλίδας. Αφού όπως είδαμε το identification αλλάζει πάντα και το total length έχει ένα εύρος τιμών στα πακέτα που στείλαμε, είναι αναμενόμενο το header checksum να έχει διαφορετική τιμή από πακέτο σε πακέτο.

Όνομα PC/ΛΣ: pc-c04 (υπολογιστής σχολής)/ WINDOWS XP	
Διεύθυνση IP: 147.102.38.154	**Διεύθυνση MAC: 00-1D-09-0C-A6-FD**

Άσκηση 4:

4.1: Σύνταξη εντολής ping ώστε να σταλθεί 1 μόνο πακέτο IPv4 χωρίς θρυματισμό μεγέθους SIZE:
ping -f -l SIZE -n 1 -4 <address>

4.2: Μέγιστη τιμή για την οποία επιτυγχάνει η αποστολή: SIZE = 1472

4.3: Μικρότερη τιμή δεδομένων ICMP για την οποία απαιτείται θρυματισμός: 1473 bytes

4.4: Φίλτρο σύλληψης: “not broadcast and not multicast”

4.5: Φίλτρο απεικόνισης: “(icmp.type == 0 or icmp.type == 8) and ip.addr == 147.102.38.153”

4.6: Όχι, δεν παράγονται πακέτα, γιατί δεν μπορούν να σταλθούν λόγω του μήκους τους.

4.7: MTU = 1500 bytes. Είδαμε ότι 1472 bytes είναι τα ICMP Data και έχουμε 8 bytes για τα υπόλοιπα πεδία του ICMP protocol και 20 bytes για το IPv4 header άρα συνολικά 1500 bytes.

4.8: Max IPv4 Length = 65535 bytes. Άρα Max ICMP Data Length = 65535 – 20 – 8 = 65507 bytes. Στο δικό μας τοπικό δίκτυο Max ICMP Data Length = 1472 bytes.

4.9: Το ping δεν επιτυγχάνει. Η μέγιστη τιμή για την οποία επιτυγχάνει είναι 65500 bytes.

4.10: Max IPv4 Packet Length = 65500 + 20 + 8 = 65528 bytes

4.11: Όχι

4.12: Χρειάστηκαν 5 πακέτα

4.13:

Identification	0xcc7e	0xcc7e	0xcc7e	0xcc7e	0xcc7e
Don’t Fragment Bit	0	0	0	0	0
More Fragments Bit	1	1	1	1	0
Fragment Offset	0	1480	2960	4440	5920

4.14: More Fragments Bit = 1

4.15: Fragment Offset = 0

4.16: Μήκος πρώτου θραύσματος = 1480 bytes (χωρίς το IPv4 header)

4.17: Το Fragment Offset δεν είναι 0

4.18: Ναι

4.19: More Fragments Bit = 1

4.20: πρώτο vs τελευταίο θραύσμα
Τα “More Fragments Bit”, “Fragment Offset” και “Header Checksum” αλλάζουν

4.21: Το προτελευταίο έχει τιμή 4440 και το τελευταίο έχει 5920 στο Fragment Offset. Πριν το προτελευταίο έχουν προηγηθεί 3 πακέτα με 1480 ICMP data bytes και 1480 * 3 = 4440 ενώ πριν το τελευταίο έχουν προηγηθεί 4 τέτοια πακέτα και 1480 * 4 = 5920. Το 1480 είναι τα Data bytes (αν δεν υπάρχουν τα υπόλοιπα πεδία του ICMP στο πακέτο αυτό) που μπορούν να μεταφερθούν με MTU 1500.

4.22: Μεταξύ των θραυσμάτων αλλάζουν τα Fragment Offset και Header Checksum. Ενώ για το τελευταίο fragment το Total Length και το More Fragments Bit είναι διαφορετικά.