



Δίκτυα Υπολογιστών
 Λύσεις σειράς ασκήσεων επανάληψης

(1)

Απρόκλητο ARP. Θα αντιμετωπισθεί από τους άλλους host και από τον πράκτορα οικείων ως κανονικό ARP και θα ενημερωθούν οι πίνακές τους με την διεύθυνση του επανεληθόντος host.

Address Resolution Protocol (request/gratuitous ARP)

Hardware type: Ethernet (0x0001)
 Protocol type: IP (0x0800)
 Hardware size: 6 (06)
 Protocol size: 4 (04)
 Opcode: request (0x0001)
 Sender MAC address: 02:e0:81:01:ad:4d
 Sender IP address: 158.125.1.118 (9e 7d 01 76)
 Target MAC address: ff:ff:ff:ff:ff:ff (Broadcast)
 Target IP address: 158.125.1.118 (9e 7d 01 76)

(2)

Η διάκριση γίνεται από το πεδίο των δύο byte που ακολουθεί μετά τις διευθύνσεις προορισμού και πηγής στο πλαίσιο. Στο μεν Ethernet το πεδίο αυτό δηλώνει το πρωτόκολλο του ανωτέρου στρώματος, στο δε IEEE 802.3 δηλώνει το μήκος του πλαισίου σε byte. Δεδομένου ότι η MTU στο Ethernet είναι 1500 byte, αν το πεδίο αυτό έχει τιμή πάνω από 1536 (0600 hexadecimal) το πλαίσιο είναι Ethernet και κάτω από 1536 το πλαίσιο είναι IEEE 802.3.

(3)

Σύμφωνα με τον αλγόριθμο του Karn, η εκτίμηση του RTT για επαναμεταδιδόμενο τεμάχιο αγνοείται.

Το RTT δεν ενημερώνεται μέχρι να ληφθεί κανονική ACK χωρίς να υπάρχει λόγος επαναμετάδοσης. Επιπρόσθετα, το RTO διπλασιάζεται σε κάθε επαναμετάδοση.

Η τιμή του RTO ήταν 8 sec όταν εστάλη το αρχικό τεμάχιο TCP, διότι

$$4:30:28 - 4:30:20 = 8.$$

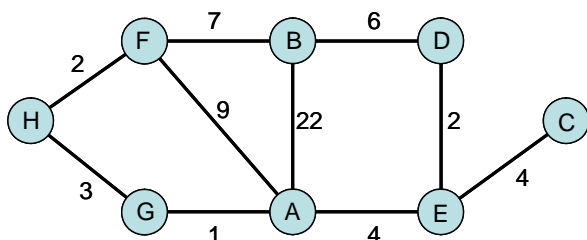
Έτσι, οι νέες τιμές για τα RTT και RTO είναι οι εξής:

$$RTT_{new} = 4 \text{ sec}$$

$$RTO_{new} = 2 \times RTO_{old} = 2 \times 8 = 16 \text{ sec}.$$

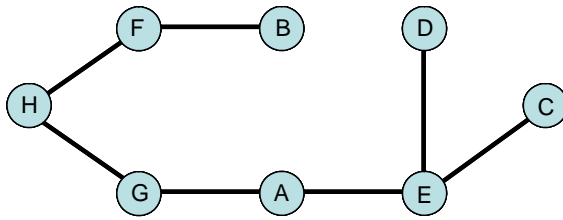
(4)

(α)

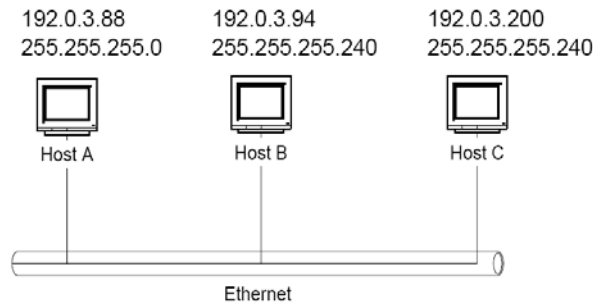


Προς	Μέσω	Κόστος
A	H	6
B	B	7
C	H	14
D	H	12
E	H	10
G	H	5
H	H	2

(β)



(5)



(α) ΟΧΙ. Μετά την εφαρμογή της μάσκας του C στην IP του A, προκύπτει ότι ο A δεν ανήκει στο ίδιο υποδίκτυο με τον C, οπότε το πακέτο απορρίπτεται. $192.0.3.80 \neq 192.0.3.192$.

(β) ΝΑΙ. Διότι μετά την εφαρμογή της μάσκας του A στην IP του B προκύπτει το ίδιο υποδίκτυο, $192.0.3.0$

(γ) ΝΑΙ. Διότι μετά την εφαρμογή της μάσκας του A στην IP του C προκύπτει το ίδιο υποδίκτυο, $192.0.3.0$.

(δ) ΝΑΙ. Διότι μετά την εφαρμογή της μάσκας του B στην IP του A προκύπτει το ίδιο υποδίκτυο, $192.0.3.80$.

(ε) ΟΧΙ. Μετά την εφαρμογή της μάσκας του B στην IP του C, προκύπτει ότι ο C δεν ανήκει στο ίδιο υποδίκτυο με τον B, οπότε το πακέτο απορρίπτεται. $192.0.3.80 \neq 192.0.3.192$.

(6)

```

00 0a e4 37 f8 36 00 12 3f 61 d7 ac 08 00 45 00
00 54 4a 25 00 00 80 01 d8 c5 80 64 0b f0 80 64
0b 06 08 00 6d 02 44 0d 06 00 cf 1c 15 47 68 89
09 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15
16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
36 37
  
```

(α) 00 12 3f 61 d7 ac

(β) 00 0a e4 37 f8 36

(γ) 128.100.11.240 (80 64 0b f0)

(δ) 128.100.11.6 (80 64 0b 06)

(ε) 84 (00 54)

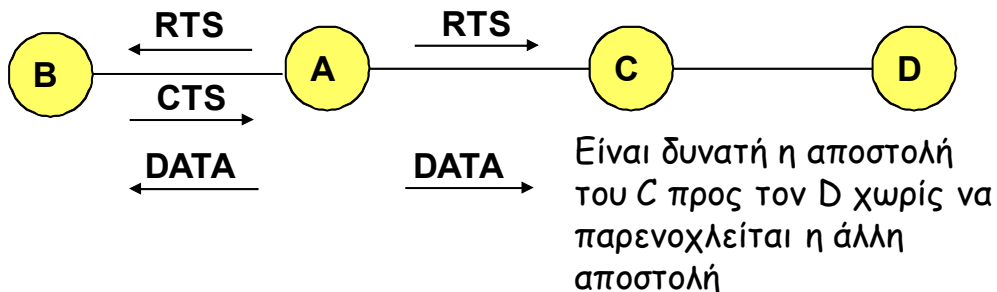
(στ) 5

(ζ) 1, είναι πακέτο ICMP

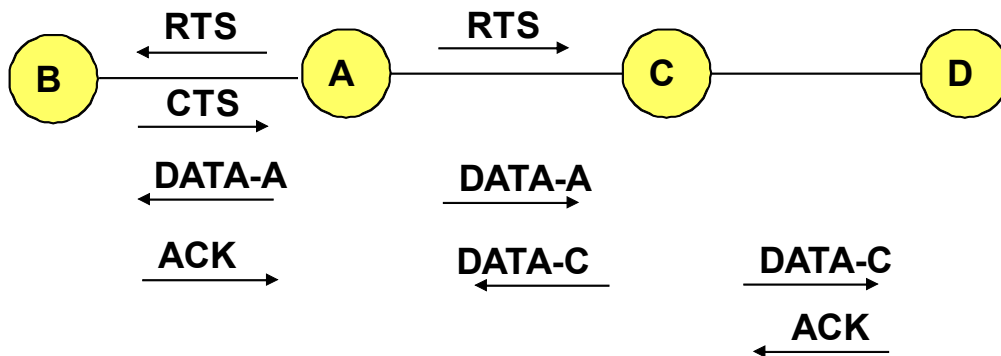
(η) 64 (84-20) byte

(7)

(α)



(β)



Υπάρχει ενδεχόμενο η αποστολή DATA-C να συμπέσει με την αποστολή ACK του B προς τον A.

(8)

(α)

Το μέγεθος του παραθύρου θα φθάσει στη μέγιστη τιμή του όταν η ζεύξη είναι πλήρης:

$$250ms \times 10Mb/s = 2.5Mbit = 312.500 \text{ bytes.}$$

(β)

$$1/2 \times 312.500 = 156.250 \text{ bytes.}$$

(γ)

Σε κάθε RTT το παράθυρο μεγαλώνει 125 byte, οπότε θα απαιτηθεί χρόνος

$$[(W_{max} / 2) / 125 \text{ byte}] \times RTT = 312.5 \text{ sec.}$$

(δ)

$$\bar{R} = [(3/4) \times W_{max}] / RTT = (0.75 \times 312500) / 0.25 = 937.500 \text{ byte/sec}$$

(ε)

Υποθέτοντας ότι η αργή αρχή αρχίζει με ένα πακέτο, πρέπει να βρούμε πόσος χρόνος θα χρειαστεί να φθάσει το παράθυρο τα 312.500/2 bytes, αν το μέγεθος του αυξάνει κατά 125bytes κάθε φορά που λαμβάνεται μια ACK. Στην ουσία, το παράθυρο διπλασιάζεται κάθε RTT, οπότε ο απαιτούμενος χρόνος είναι:

$$\log_2 \left(\frac{312,500}{2 \times 125} \right) \times 250ms \approx 2.5 \text{ seconds}$$

(9)

(α) Για 20 υπολογιστές ανά δίκτυο χρειαζόμαστε 5 bit.

Άρα απομένουν 3 bit για τα υποδίκτυα, δηλαδή, επιπλέον 3 άσσοι, 8 υποδίκτυα, μάσκα: /27

(β) Για το έβδομο υποδίκτυο, τα τρία πρώτα bit από το τελευταίο byte θα είναι 110, δηλαδή 192.

Η διεύθυνσή του θα είναι 200.35.1.192/27

(γ) Η διεύθυνση εκπομπής για το υπόψη δίκτυο θα έχει τα 5 τελευταία bit άσσους. Δηλαδή, το τελευταίο byte θα είναι το 11011111 = 223.

Άρα, η διεύθυνση εκπομπής για το 7^ο δίκτυο είναι: 200.35.1.223.

(10)

(α) ΟΧΙ

(β) ΟΧΙ, λόγω TTL

(11)

(α) $\log_2 32 = 5$, επιπλέον 5 άσσοι, 32 υποδίκτυα, μάσκα: /29

(β) $2^{32} - 2^{29} = 8$

(γ) Πρώτο υποδίκτυο.

Η πρώτη διεύθυνση είναι η αρχική διεύθυνση του μπλοκ: 211.17.180.0

Η τελευταία διεύθυνση: 211.17.180.7

(δ) Τελευταίο υποδίκτυο.

Η πρώτη διεύθυνση είναι: 211.17.180.248

Η τελευταία διεύθυνση: 211.17.180.255

(12)

Ροή A

$1 + 2 + 4 + 8 + 9 + 10 + 11 + 12 + 13 + 14 + 15 = 99$

Ροή B

$1 + 2 + 4 + 8 + 9 + 10 = 34$

(13)

Total time = DIFS + RTS + PROP + SIFS + CTS + PROP + SIFS + PREAMBLE + TF + PROP + SIFS + ACK + PROP

TF = Data packet / Transmission Rate = $1100 \times 8 / (11 \times 10^6) = 800\mu s$

$4 \times \text{PROP} = 4 \times 750m / 3 \times 10^8 = 10\mu s$

DIFS = 50μs

RTS + CTS + ACK = $3 \times 200 = 600\mu s$

PREAMBLE = 200μs

$$\text{SIFS} \times 3 = 10\mu\text{s} \times 3 = 30\mu\text{s}$$

$$\text{Total time} = 800\mu\text{s} + 10\mu\text{s} + 50\mu\text{s} + 600\mu\text{s} + 200\mu\text{s} + 30\mu\text{s} = 1690\mu\text{s}$$

$$\text{Efficiency} = 800/1690 = 0.473$$

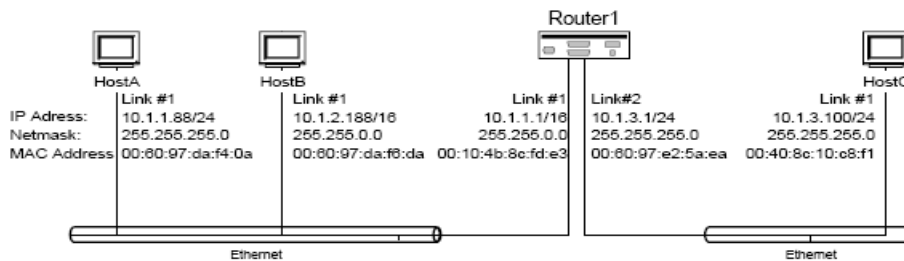
$$\text{Throughput} = 0.4 \times 11\text{Mbps} = 5.2\text{Mbps}$$

(14)

Η μέγιστη τιμή για τον αύξοντα αριθμό είναι $2^{32} - 1$.

Αν αρχίσουμε το μέτρημα από το 7000 θα μεσολαβήσουν $[(2^{32} - 1) - 7000] / 10^6 = 4295 \text{ sec} = 1\text{h } 11\text{min } 35\text{sec}$.

(15)



Host A		Host B		Router 1		Host C	
Προορισμός	Πύλη	Προορισμός	Πύλη	Προορισμός	Πύλη	Προορισμός	Πύλη
0.0.0.0	10.1.1.1	0.0.0.0	10.1.1.1	10.1.0.0/16	Link #1	0.0.0.0	10.1.3.1
10.1.1.0/24	Link#1	10.1.0.0/16	Link#1	10.1.1.0/24	Link #1	10.1.3.0/24	Link#1
				10.1.3.0/24	Link #2		

α)

Μετά την εφαρμογή της μάσκας του A στη διεύθυνση 10.1.2.188 προκύπτει ότι ο 10.1.2.188 δεν είναι στο ίδιο LAN, οπότε το πακέτο ICMP θα πρέπει να σταλεί στην πύλη 10.1.1.1.

Στέλνεται πακέτο ARP για να μαθευτεί η φυσική διεύθυνση της πύλης 10.1.1.1.

Address Resolution Protocol (request)

Hardware type: Ethernet (0x0001)
 Protocol type: IP (0x0800)
 Hardware size: 6 (06)
 Protocol size: 4 (04)
 Opcode: request (0x0001)
 Sender MAC address: 00:60:97:da:f4:0a
 Sender IP address: 10.1.1.88
 Target MAC address: ff:ff:ff:ff:ff:ff
 Target IP address: 10.1.1.1

Με την απάντηση της πύλης 10.1.1.1 στο ARP γίνεται γνωστή η φυσική της διεύθυνση 00:10:4b:8c:fd:e3 και στη συνέχεια ο A προωθεί το πακέτο ICMP Echo request (type=8), ενθυλακωμένο σε πακέτο IP και σε πλαίσιο Ethernet, στη φυσική διεύθυνση της πύλης 10.1.1.1.

```
Address Resolution Protocol (response)
  Hardware type: Ethernet (0x0001)
  Protocol type: IP (0x0800)
  Hardware size: 6 (06)
  Protocol size: 4 (04)
  Opcode: response (0x0002)
  Sender MAC address: 00:10:4b:8c:fd:e3
  Sender IP address: 10.1.1.1
  Target MAC address: 00:60:97:da:f4:0a
  Target IP address: 10.1.1.88
```

Ο δρομολογητής 1 εφαρμόζοντας τη μάσκα του στη διεύθυνση 10.1.2.188 διαπιστώνει ότι είναι στο ίδιο LAN (link#1). Στέλνει πακέτο ARP για να μάθει τη φυσική διεύθυνση του 10.1.2.188 και λαμβάνει ως απάντηση την 00:60:97:da:f6:da.

```
Address Resolution Protocol (request)
  Hardware type: Ethernet (0x0001)
  Protocol type: IP (0x0800)
  Hardware size: 6 (06)
  Protocol size: 4 (04)
  Opcode: request (0x0001)
  Sender MAC address: 00:10:4b:8c:fd:e3
  Sender IP address: 10.1.1.1
  Target MAC address: ff:ff:ff:ff:ff:ff
  Target IP address: 10.1.2.188
```

```
Address Resolution Protocol (response)
  Hardware type: Ethernet (0x0001)
  Protocol type: IP (0x0800)
  Hardware size: 6 (06)
  Protocol size: 4 (04)
  Opcode: response (0x0002)
  Sender MAC address: 00:60:97:da:f6:da
  Sender IP address: 10.1.2.188
  Target MAC address: 00:10:4b:8c:fd:e3
  Target IP address: 10.1.1.1
```

Στη συνέχεια ο δρομολογητής προωθεί το πακέτο ICMP Echo request (type=8), ενθυλακωμένο σε πακέτο IP και πλαίσιο Ethernet, στην υπόψη φυσική διεύθυνση.

β)

Μετά την εφαρμογή της μάσκας του Β στη διεύθυνση 10.1.3.100 προκύπτει ότι ο 10.1.3.100 είναι στο ίδιο υποδίκτυο (link#1), οπότε στέλνει ARP για να μάθει τη φυσική του διεύθυνση.

```
Address Resolution Protocol (request)
  Hardware type: Ethernet (0x0001)
  Protocol type: IP (0x0800)
  Hardware size: 6 (06)
  Protocol size: 4 (04)
```

Opcode: request (0x0001)
Sender MAC address: 00:60:97:da:f6:da
Sender IP address: 10.1.2.188
Target MAC address: ff:ff:ff:ff:ff:ff
Target IP address: 10.1.3.100

Στην ερώτηση ARP απαντάει ο proxy ARP του Router1 και στέλνει τη φυσική διεύθυνση του, δηλαδή, 00:10:4b:8c:fd:e3.

Address Resolution Protocol (response)
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6 (06)
Protocol size: 4 (04)
Opcode: response (0x0002)
Sender MAC address: 00:10:4b:8c:fd:e3
Sender IP address: 10.1.3.100
Target MAC address: 00:60:97:da:f6:da
Target IP address: 10.1.2.188

Στη συνέχεια ο B προωθεί το πακέτο ICMP Echo request (type=8), ενθυλακωμένο σε πακέτο IP και πλαίσιο Ethernet, στην ανωτέρω φυσική διεύθυνση.

Ο Router1, όταν λάβει το πακέτο ICMP διαπιστώνει, εφαρμόζοντας τη μεγαλύτερη μάσκα, ότι το 10.1.3.100 βρίσκεται στο link#2. Στέλνει λοιπόν ARP στο link#2 για να μάθει τη φυσική διεύθυνση του 10.1.3.100.

Address Resolution Protocol (request)
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6 (06)
Protocol size: 4 (04)
Opcode: request (0x0001)
Sender MAC address: 00:60:97:e2:5a:ea
Sender IP address: 10.1.3.1
Target MAC address: ff:ff:ff:ff:ff:ff
Target IP address: 10.1.3.100

Ο C απαντάει στην ερώτηση ARP στέλνοντας τη φυσική του διεύθυνση, δηλαδή, 00:40:8c:10:c8:f1.

Address Resolution Protocol (response)
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6 (06)
Protocol size: 4 (04)
Opcode: response (0x0002)
Sender MAC address: 00:40:8c:10:c8:f1
Sender IP address: 10.1.3.100
Target MAC address: 00:60:97:e2:5a:ea
Target IP address: 10.1.3.1

Τέλος, ο router1 προωθεί το πακέτο ICMP Echo request (type=8), ενθυλακωμένο σε πακέτο IP και πλαίσιο Ethernet, στη φυσική διεύθυνση του host C.

(16)

16 byte δεδομένα / 24 byte συνολικού μήκους = 0.666

16 byte δεδομένα / 72 byte ελάχιστου μήκους πλαισίου = 0.222

(17)

28 byte

72 byte (με το απαιτούμενο παραγέμισμα). Τούτο περιλαμβάνει 8 byte προοίμιο και SFD.

(18)

45 00 00 54 00 03 00 00 20 06 00 00 7C 4E 03 02 B4 0E 0F 02

Δεν υπάρχουν επιλογές.

Το πακέτο δεν είναι θρυμματισμένο.

Το πακέτο μεταφέρει $84 - 20 = 64$ byte δεδομένων.

Το άθροισμα ελέγχου δεν χρησιμοποιείται.

Το πακέτο μπορεί να επισκεφθεί 32 ακόμη δρομολογητές.

Ο αριθμός ταυτότητας του πακέτου είναι 3.

Ο τύπος υπηρεσίας είναι normal.

(19)

(α)

$$W = (R/8) \times RTT$$

$$\begin{aligned} W &= [(20 \times 10^6) / 8] \times 2 \times (150000000 / 300000) = \\ &= 2.5 \times 2 \times 500 \times 10^6 = 2.5 \text{ Gbytes} \end{aligned}$$

$$2.5 \text{ Gbytes} / (10 \text{ kbyte/packet}) = 250000 \text{ packets}$$

(β)

Το TCP slow start είναι πολύ αργό. Αυξάνει από ένα πακέτο σε 250000 πακέτα ανά ACK.

Χρειάζονται περίπου 18 RTTs και κάθε RTT είναι 1000 sec.

Άρα απαιτούνται περίπου 18000 sec = 5h 16min 40sec.

(20)

$$2^{15} > 32000$$

$$\chi = 8 + 15 = 23$$

$\Rightarrow 10.0.0.0/23$.

Έστω ότι επιλέγουμε το υποδίκτυο $10.202.216.0/23$

Η διεύθυνση εκπομπής στο δίκτυο αυτό θα είναι:

$10.202.217.255$ ή $00001010\ 11001010\ 11011001\ 11111111$

(21)

(α) $(12 \times 5 \times 10^9) / (250 \times 8) = 30 \times 10^6$ πακέτα/sec

(β) $(12 \times 250 \times 8) / (0.75 \times 10^6) = 0.032$ sec = 32ms

(γ) 30×10^6 πακέτα/sec / 12 = 2.5×10^6 ροές/sec

$$1/32\text{ms} = 1000/32 = 31.25 \text{ ροές/sec}$$

$$(2.5 \times 10^6 \text{ ροές/sec}) / 31.25 \text{ ροές/sec} = 80000 \text{ ροές}$$

(δ) $(30 \times 10^6 \text{ πακέτα/sec}) \times (64 \text{ byte}) \times (24\text{h/day}) \times (3600\text{sec/h}) = 165.89 \text{ Tbyte/day}$

(22)

Αν υπάρχουν 1000 απερχόμενες συνδέσεις και δεν συμβαίνει τίποτε άλλο στον εξυπηρετητή, θα χρησιμοποιούνται 1001 υποδοχές – η welcoming υποδοχή και οι 1000 για την επικοινωνία του server με τους client.

Ο επιπλέον αριθμός θύρας που θα χρησιμοποιείται στον εξυπηρετητή είναι ο αριθμός θύρας που σχετίζεται με την welcoming υποδοχή, δηλαδή, η θύρα 80 για τον εξυπηρετητή ιστού.

(23)

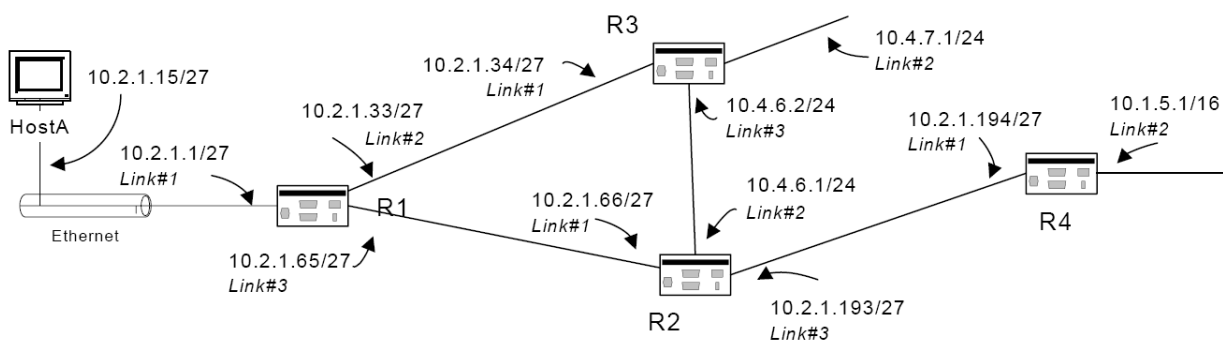
(α)

Θα φθάσει ένα δεδομένογραμμα στην οντότητα UDP του δέκτη. Το στρώμα IP θα θρυμματίσει κτο δεδομένογραμμα UDP στην πλευρά της εκπομπής αλλά θα φροντίσει, επίσης, να το επανασυναρμολογήσει στην πλευρά της λήψης, πριν το παραδώσει στο UDP.

(β)

$$\text{Απόδοση} = \text{Data} / (\text{Data} + \text{UDP header} + \text{IP header}) = 28 / (28 + 8 + 20) = 0.5.$$

(24)



Πίνακας δρομολόγησης R1		Πίνακας δρομολόγησης R2		Πίνακας δρομολόγησης R3		Πίνακας δρομολόγησης R4	
Προορισμός	Πύλη	Προορισμός	Πύλη	Προορισμός	Πύλη	Προορισμός	Πύλη
10.2.1.0/27	Link #1	10.2.1.64/27	Link #1	10.2.1.32/27	Link #1	10.2.1.192/27	Link#1
10.2.1.32/27	Link#2	10.4.6.0/24	Link#2	10.4.7.0/24	Link#2	10.1.0.0/16	Link#2
10.2.1.64/27	Link#3	10.2.1.192/27	Link#3	10.4.6.0/24	Link#3	10.4.0.0/16	10.2.1.193
10.0.0.0/8	10.2.1.34	10.4.7.0/24	10.4.6.2	10.1.0.0/16	10.4.6.1	10.2.1.0/24	10.2.1.193
10.4.7.1/32	10.2.1.66	10.1.0.0/16	10.2.1.194	192.0.0.0/8	10.4.6.1		
10.2.1.192/27	10.2.1.66	10.2.1.0/27	10.2.1.65				

(α) και (β)

ICMP Echo Request

Μετά την εφαρμογή της μάσκας υποδικτύου στον host A προκύπτει ότι η διεύθυνση 10.1.5.1 ανήκει σε άλλο δίκτυο, οπότε το πακέτο ICMP οδηγείται στην **προεπιλεγμένη πύλη 10.2.1.1**.

Στον δρομολογητή R1, σύμφωνα με την εγγραφή **10.0.0.0/8 ⇒ 10.2.1.34**, το πακέτο ICMP οδηγείται στον **R3**.

Στον R3, σύμφωνα με την εγγραφή **10.1.0.0/16 ⇒ 10.4.6.1**, το πακέτο ICMP οδηγείται στον **R2**.

Στον R2, σύμφωνα με την εγγραφή **10.1.0.0/16 ⇒ 10.2.1.194**, το πακέτο ICMP οδηγείται στον **R4**.

Στον R4, σύμφωνα με την εγγραφή **10.1.0.0/16 ⇒ Link#2**, το πακέτο ICMP οδηγείται στο **Link#2**.

ICMP Echo Reply

Στον R4, σύμφωνα με την εγγραφή **10.2.1.0/24 ⇒ 10.2.1.193**, το πακέτο ICMP οδηγείται στον **R2**.

Στον R2, σύμφωνα με την εγγραφή **10.2.1.0/27 ⇒ 10.2.1.65**, το πακέτο ICMP οδηγείται στον **R1**.

Στον R1, σύμφωνα με την εγγραφή **10.2.1.0/27 ⇒ Link#1**, το πακέτο ICMP οδηγείται στον **Host A**.

(25)

(α)

Η πιθανότητα να πέσει ο αύξων αριθμός μέσα στο παράθυρο μετάδοσης είναι:

$$p = 2^{13}/2^{32} = 2^{-19}$$

Άρα, ο αναμενόμενος αριθμός προσπαθειών μέχρι κάποιο τεμάχιο να πετύχει τον σκοπό του είναι:

$$A = 1/p = 2^{19} = 524288 \text{ προσπάθειες.}$$

(β)

Στις 524288 προσπάθειες θα παραχθούν 524288 RST bit ή

$$(26 \text{ eth} + 20 \text{ IP} + 20 \text{ TCP} + 6 \text{ pad}) \times 8 \times 524288 \text{ bit.}$$

Ο χρόνος για την αποστολή τους είναι:

$$(26 + 20 + 20 + 6) \times 8 \times 524288 \text{ bit} / 10^8 \text{ bps} \approx 3.02 \text{ sec.}$$