ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΓΚΟΥΜΕ ΛΑΟΥΡΕΝΤΙΑΝ
ΑΜ: 031 18 014
ΕΞΑΜΗΝΟ: 8^{ο}
ΟΜΑΔΑ: 3

ΕΡΓΑΣΤΗΡΙΟ ΔΙΚΤΥΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 2: ΔΙΚΤΥΩΣΗ ΣΥΣΤΗΜΑΤΩΝ ΣΤΟ VirtualBox

Άσκηση 1: Γνωριμία με το περιβάλλον εργασίας

1.1) – 1.15) Ακολουθούμε τα βήματα. (Εντολή “history -c” για διαγραφή ιστορικού αναζητήσεων” στο 1.11).

Άσκηση 2: Ανάλυση δικτυακών πρωτοκόλλων με το TCPDUMP

2.1) Με την εντολή “ifconfig”.

2.2) Εκτελώντας διαδοχικά την εντολή “ifconfig em0 down” για απενεργοποίηση και στη συνέχεια την εντολή “ifconfig em0 up” για ενεργοποίηση.

2.3) Με τις εντολές “man tcpdump”, “man pcap” και “man pcap-filter”.

2.4) Με την εντολή “tcpdump -n -i em0”. (Στις επομένες εντολές για χάριν συντομίας θα παραλείπουμε το όρισμα -i em0, καθώς το μηχάνημά μας έχει μία μόνο κάρτα δικτύου πέραν της loopback, οπότε by default η κίνηση ανιχνεύεται σε αυτήν).

2.5) Με την εντολή “tcpdump -X”.

2.6) Ουσιαστικά θέλουμε να τυπώσουμε επιπλέον την επικεφαλίδα ethernet, άρα με την εντολή “tcpdump -e”.

2.7) Με την εντολή “tcpdump -s 68”.

2.8) Με την εντολή “tcpdump ip host 10.0.0.1 -v ”.

2.9) Με την εντολή “tcpdump host 10.0.0.1 or 10.0.0.2”.

2.10) Με την εντολή “tcpdump net 1.1.0.0/16”.

2.11) Με την εντολή “tcpdump not net 192.168.1.0/24 -e”.

2.12) Με την εντολή “tcpdump ip broadcast”.

2.13) Με την εντολή “tcpdump ip and greater 576”.

2.14) Με την εντολή “tcpdump ‘ip[8] < 5’”.

2.15) Με την εντολή “tcpdump ‘(ip[0] & 0x0f)’ > 5”. Στο πρώτο byte της επικεφαλίδας IP έχουμε τα πρώτα 4 bits για το Version και άλλα 4 για το Header Length, το οποίο by default είναι 5 εκτός και αν έχουμε options. Επομένως, εκτελούμε bitwise and με το 1111, ώστε να πάρουμε τα τελευταία 4 bits και τα συγκρίνουμε με το 5.

2.16) Με την εντολή “tcpdump ‘icmp and src host 10.0.0.1’ ”.

2.17) Με την εντολή “tcpdump ‘tcp and dst host 10.0.0.2’ ”.

2.18) Με την εντολή “tcpdump ‘udp and dst port 53’ ”.

2.19) Με την εντολή “tcpdump ‘tcp and host 10.0.0.10’ ”.

2.20) Με την εντολή “tcpdump ‘tcp and host 10.0.0.10 and port 23’ -w sample_capture”.

2.21) Με την εντολή “tcpdump ‘(tcp[13] & 0x3f) = 0x02’ ”. Αρχικά εφαρμόζουμε κατάλληλη μάσκα (0011 1111), ώστε να πάρουμε τα τελευταία 6 bits, τα οποία και αφορούν τις σημαίες που μας ενδιαφέρουν. Στη συνέχεια συγκρίνουμε το αποτέλεσμα αυτό με το 0000 0010, το οποίο υποδεικνύει την μοναδικότητα του flag SYN.

2.22) Με την εντολή “tcpdump ‘tcp[tcpflags] & ((tcp-syn) | (tcp-syn & tcp-ack)) != 0’ ”.

2.23) Με την εντολή “tcpdump ‘tcp[tcpflags] & (tcp-fin) != 0’ ”.

2.24) Αρχικά, η παράσταση tcp[12:1] μας δίνει τα 8 bits του $13^{ου}$ Byte μιας TCP επικεφαλίδας. Στη συνέχεια, η έκφραση tcp[12:1] & 0xf0 μάς δίνει τις τιμές των τεσσάρων αριστερότερων bits, τα οποία και εκφράζουν την τιμή του πεδίου Data Offset (Header Length σε 32bitες λέξεις). Στη συνέχεια, με την τελική παράσταση που μας δίνεται, διαιρούμε ουσιαστικά το Data Offset ακέραια με το 4. Αυτό που προκύπτει τελικά είναι το πραγματικό μέγεθος της επικεφαλίδας σε bytes. Π.χ. αν είχαμε αρχικά ως $13^{ο}$ byte το 01010001, τότε, από τα 4 αριστερότερα bits συμπεραίνουμε ότι το μήκος της επικεφαλίδας είναι $0101 = 5_{10}$ * 4bytes = 20 bytes, ενώ αν εφαρμόσουμε το φίλτρο τότε το byte αυτό μετατρέπεται σε $00010100 = 20_{10}$.

2.25) Με την εντολή “tcpdump ‘(tcp[12] & 0xf0) > 5’ ”.

2.26) Με την εντολή “tcpdump -A port 80”.

2.27) Με την εντολή “tcpdump ‘port 23 and host edu-dy.cn.ntua.gr’ ”.

2.28) Με την εντολή “tcpdump ip6”.

Άσκηση 3: Δικτύωση Host-Only

3.1) IPv4 του Host-Only Ethernet Adapter: 192.168.56.1.

3.2) IPv4 του DHCP Server: 192.168.56.100 και περιοχή εκχώρησης διευθύνσεων: 192.168.56.101 έως 192.168.56.255.

3.3) Εκτελούμε την εντολή “dhclient” σε κάθε μηχάνημα.

3.4) Αποδίδεται η 192.168.56.103 στο PC1 και η 192.168.56.104 στο PC2.

3.5) Κάνουμε ping από το 1 μηχάνημα στο άλλο και λαμβάνουμε απάντηση (π.χ. ping -c 4 192.168.56.104 από το PC1).

3.6) Κάνοντας ping από το terminal του υπολογιστή μας σε κάθε μία από τις IPv4 διευθύνσεις που αποδόθηκαν παραπάνω.

3.7) Με την εντολή “netstat -r”.

3.8) Με την παραπάνω εντολή λαμβάνουμε το παρακάτω αποτέλεσμα:

```
root@PC:~ # netstat -r
Routing tables

Internet:
Destination        Gateway            Flags     Netif Expire
localhost          link#2             UH          lo0
192.168.56.0/24    link#1             U           em0
192.168.56.103     link#1             UHS         lo0

Internet6:
Destination        Gateway            Flags     Netif Expire
::/96              localhost          UGRS        lo0
localhost          link#2             UH          lo0
::ffff:0.0.0.0/96  localhost          UGRS        lo0
fe80::/10          localhost          UGRS        lo0
fe80::%lo0/64      link#2             U           lo0
fe80::1%lo0        link#2             UHS         lo0
ff02::/16          localhost          UGRS        lo0
```

Όπως είναι αναμενόμενο, δεν υπάρχει gateway μιας και στη Host-Only δικτύωση δεν επιτρέπεται σύνδεση με συσκευές εκτός του Host-Only δικτύου.

3.9) Δε μπορούμε να κάνουμε ping στην IPv4 διεύθυνση της φυσικής κάρτας δικτύου του host machine, καθώς για τα VMs ανήκει σε διαφορετικό δίκτυο, για αυτό και εάν ο host θέλει να επικοινωνήσει με τα VMs το κάνει με χρήση της Virtual κάρτας δικτύου και όχι της φυσικής.

3.10) Με την εντολή “hostname” βλέπουμε πως τα μηχανήματα ονομάζονται “PC.ntua.lab”.

3.11) Εκτελούμε την εντολή “hostname PC1” ή “hostname PC2” αντίστοιχα.

3.12) Η αλλαγή φαίνεται στο prompt:

3.13) Όχι, δε το περιέχει, αντ’ αυτού περιέχει το “PC.ntua.lab”, άρα αυτό θα είναι το όνομα του PC1 σε ενδεχόμενη επανεκκίνηση.

3.14) Διορθώνουμε την τιμή του πεδίου “hostname=” σε PC1 και PC2 αντίστοιχα με χρήση του vi (“vi /etc/rc.conf”).

3.15) Όπως διαβάζουμε από το manpage της hosts (“man hosts”), θα πρέπει για κάθε IPv4 διεύθυνση που επιθυμούμε να χρησιμοποιούμε όνομα αντί αυτής να προσθέσουμε μια γραμμή με τα παρακάτω: Internet Address, Official Host Name, Aliases. Επομένως, προσθέτουμε στο /etc/hosts του PC1 τη γραμμή “192.168.56.104 PC2 PC2.local”, ενώ στου PC2 τη γραμμή “192.168.56.103 PC1 PC1.local”.

3.16) Στο /etc/hosts είναι ορισμένο το “127.0.0.1 localhost localhost.my.domain”, επομένως αξιοποιούμε τη λειτουργία του αρχείου hosts με την εντολή “ping -c 4 localhost”.

3.17) Είτε με την εντολή “tcpdump icmp and host PC1 -l | tee capture” είτε με την εντολή “tcpdump icmp and host PC1 -l > capture & tail -f capture”.

3.18) Λαμβάνει απαντήσεις μήκους 64 bytes με TTL επίσης 64.

3.19) Κάνοντας “ping -c 2 192.168.56.1” λαμβάνουμε απαντήσεις με TTL = 128.

3.20) Χρησιμοποιήθηκε η εντολή “tcpdump -ve icmp”, ωστόσο εάν θέλαμε ακόμη περισσότερες πληροφορίες θα μπορούσαμε να εκτελέσουμε την ίδια εντολή, μόνο που αντί για -v θα είχαμε -vvv.

3.21) Το φιλοξενούν μηχάνημα αναφέρει πως παράγει 32bytes, τα οποία, ωστόσο αφορούν καθαρά το ICMP Payload, επομένως, το συνολικό ICMP μήνυμα εάν συμπεριλάβουμε την ICMP επικεφαλίδα είναι 40 bytes. Η διαφορά αυτή έγκειται στα λειτουργικά συστήματα των 2 μηχανημάτων, καθώς τα μεν Windows στέλνουν μηνύματα μήκους 40 bytes, ενώ τα δε unix* μηχανήματα 64 bytes.

3.22) Η τιμή είναι 128 για το πακέτο που στέλνει το φιλοξενούν στο PC2 και 64 για το reply που λαμβάνει ο host, συμφωνώντας με τις τιμές που βρήκαμε πριν.

3.23) Δε παρατηρείται τίποτα.

3.24) Αυτή τη φορά, παρατηρούμε κίνηση σα να είμαστε το PC2.

Άσκηση 4: Δικτύωση Internal

4.1) PC1: “ifconfig em0 192.168.56.103/24”, PC2: “ifconfig em0 192.168.56.104/24"

4.2) Λάβαμε το παρακάτω μήνυμα, το οποίο ενημερώνει για την αποδέσμευση της δυναμικά καταχωρημένης διεύθυνσης IP από τον DHCP Server:

```
root@PC1:~ # Mar 12 17:47:37 PC1 dhclient[689]: My address (192.168.56.103)
deleted, dhclient exiting
Mar 12 17:47:37 PC1 dhclient[673]: connection closed
Mar 12 17:47:37 PC1 dhclient[673]: exiting.
```

4.3) Εκτελούμε “tcpdump -ev”.

4.4) Όχι, δε μπορούμε.

4.5) Όχι, δε παρατηρούμε.

4.6) Όχι, επίσης δε μπορούμε.

4.7) Όχι, δε παρατηρούμε.

4.8) Ναι, τώρα επικοινωνούν κανονικά.

4.9) Το φιλοξενούν μηχάνημα αδυνατεί να επικοινωνήσει με οποιοδήποτε από τα μηχανήματα όπως και ήταν αναμενόμενο. Ο λόγος που αυτό συμβαίνει, είναι πως με τη δικτύωση Internal Network στην πραγματικότητα δημιουργούμε ένα εικονικό ιδιωτικό LAN δίκτυο για τα VMs μας, χωρίς να υπάρχει δυνατότητα επικοινωνίας με τον host, αφού η εικονική διεπαφή που διαθέτει ο host δεν είναι στο δίκτυο αυτό.

4.10) Εκτελούμε “tcpdump -n” στο PC1.

4.11) Αδειάζουμε τον πίνακα arp του PC2 με την εντολή “arp -ad”. Παράγονται τα εξής μηνύματα τύπου ARP request, δηλαδή ο PC2 ψάχνει την MAC address της διεύθυνσης 192.168.56.1:

```
root@PC1:~ # tcpdump -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em0, link-type EN10MB (Ethernet), capture size 262144 bytes
18:35:36.107624 ARP, Request who-has 192.168.56.1 tell 192.168.56.104, length 46
18:35:37.112303 ARP, Request who-has 192.168.56.1 tell 192.168.56.104, length 46
18:35:38.119419 ARP, Request who-has 192.168.56.1 tell 192.168.56.104, length 46
18:35:39.130801 ARP, Request who-has 192.168.56.1 tell 192.168.56.104, length 46
```

4.12) Το μήνυμα “host is down” υποδεικνύει πως δε γνωρίζουμε τη διαδρομή για τη διεύθυνση που κάναμε ping.

4.13) Οι τελευταίες διαθέσιμες διευθύνσεις IP του υποδικτύου είναι οι 10.11.12.61 και 10.11.12.62 αντίστοιχα (η 10.11.12.63 δε θεωρείται διαθέσιμη καθώς προορίζεται για broadcast). Επομένως, εισάγουμε τις εντολές:
PC1: ifconfig em0 10.11.12.61 netmask 255.255.255.192 broadcast 10.11.12.63
PC2: ifconfig em0 10.11.12.62 netmask 255.255.255.192 broadcast 10.11.12.63

4.14) Τα μηχανήματα συνεχίζουν να επικοινωνούν κανονικά.

Άσκηση 5: Δικτύωση NAT

5.1) Εκτελούμε σε κάθε μηχάνημα “dhclient em0”.

5.2) Αποδόθηκε στο καθένα από αυτά η IP 10.0.2.15 από τη διεύθυνση 10.0.2.2.

5.3) Εκτελώντας “netstat -r” βλέπουμε πως προεπιλεγμένη πύλη είναι η 10.0.2.2.

5.4) Το περιεχόμενο του αρχείο /etc/resolv.conf φαίνεται παρακάτω:

```
root@PC2:~ # cat /etc/resolv.conf
# Generated by resolvconf
nameserver 62.217.126.164
nameserver 194.177.210.210
```

5.5) Στο αρχείο /var/db/dhclient.leases.em0.

5.6) Ναι, μπορούμε να κάνουμε “ping -c 4 10.0.2.2”.

5.7) Το νέο εικονικό μηχάνημα επικοινωνεί κανονικά με το internet, μιας και διατίθεται για αυτό προκαθορισμένη πύλη, στην οποία και θα αποσταλούν τα όποια πακέτα έχουν προορισμό σε εξωτερικό δίκτυο για να δρομολογηθούν. Αν π.χ. εκτελέσουμε “ping -c 2 www.google.com” λαμβάνουμε κανονικά απάντηση.

5.8) Παρατηρήσαμε τα εξής:

- 10.0.2.1 (δε λαμβάνουμε απάντηση)
- 10.0.2.2 (λαμβάνουμε απάντηση – default gateway)
- 10.0.2.3 (λαμβάνουμε απάντηση – proxy DNS server)
- 10.0.2.4 (λαμβάνουμε απάντηση – TFTP Server)

5.9) Το κάθε VM βλέπει τον εαυτό του σαν μοναδικό στο δίκτυό του και επικοινωνεί με το δικό του gateway router, το οποίο με τη σειρά του επικοινωνεί με τη φυσική κάρτα δικτύου του host. Επομένως, δεν υπάρχει τρόπος να δρομολογηθεί ένα πακέτο από το PC3 στο PC1 ή στο PC2, διότι θα έχει ως αποδέκτη την IP διεύθυνση 10.0.2.15, επομένως θα στέλνει στην πραγματικότητα πακέτα στον εαυτό του.

5.10) Για το κάθε όρισμα που χρησιμοποιήθηκε έχουμε τα παρακάτω:

- -I: Επιβάλει χρήση ICMP Echo μηνυμάτων αντί για UDP datagrams
- -n: Εμφανίζει μόνο τις διευθύνσεις από τις οποίες περνάνε τα πακέτα χωρίς να κάνει resolve σε ονόματα.
- -q: Καθορίζει το πόσα πακέτα θα σταλούν ανά request (το default είναι 3, εμείς στέλνουμε 1)
- 1.1.1.1: Η τελική διεύθυνση των πακέτων μας

5.11) Διεύθυνση IPv4 πηγής: 10.0.2.15
Τύπος μηνυμάτων που παράγει η traceroute: ICMP Echo request.

5.12) Από το Wireshark ως διεύθυνση πηγής εμφανίζεται η 10.3.20.24, δηλαδή αυτή του υπολογιστή μας (host).

5.13) Καταγράφηκαν κατά σειρά οι εξής διευθύνσεις:

- 10.3.20.1
- 62.217.77.8
- 176.126.38.5

5.14) Διεύθυνση προορισμού είναι η “1.1.1.1”. (Wireshark)

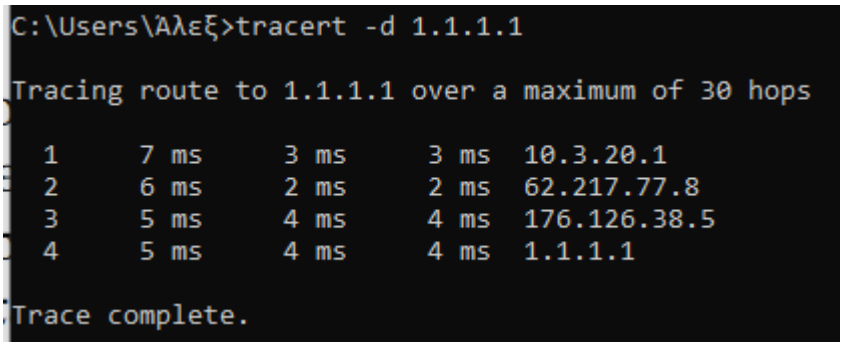
5.15) Εμφανίζονται κατά σειρά οι εξής:

- 10.0.2.2
- 10.3.20.1
- 62.217.77.8
- 176.126.38.5

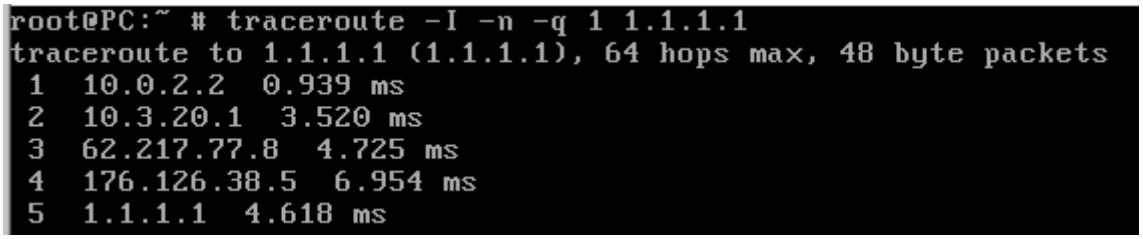
5.16) Διεύθυνση προορισμού είναι η “1.1.1.1”. (tcpdump)

5.17) Δεν υπάρχει 1 προς 1 αντιστοίχηση, καθώς στο tcdump καταγράφηκε ένα επιπλέον τέτοιο μήνυμα από την 10.0.2.2 .

5.18) Από το φιλοξενούν μηχάνημα βλέπουμε τις παρακάτω 4 αναπηδήσεις:



Αντίστοιχα, από το εικονικό μηχάνημά βλέπουμε τις εξής 5 αναπηδήσεις:



Η διαφορά οφείλεται στο γεγονός ότι από το εικονικό μηχάνημα τα πακέτα θα πρέπει να περάσουν πρώτα από το gateway του εικονικού μηχανήματος και στη συνέχεια από το gateway του φιλοξενούντος, ενώ στο φιλοξενούν δεν υπάρχει αυτό το επιπλέον hop.

Άσκηση 6: Δικτύωση NAT Network

6.1) Έχει ορισθεί η 10.0.2.0/24.

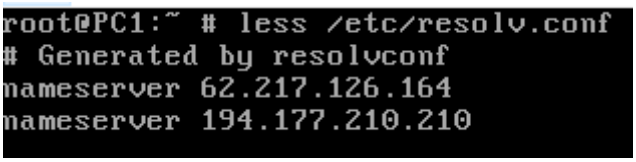
6.2) Σε καθένα από τα μηχανήματα εκτελούμε την εντολή “ifconfig em0 delete” και “rm /var/db/dhclient.leases.em0”.

6.3) Εκτελούμε “dhclient em0”.

6.4) Αποδόθηκαν στο PC1 και PC2 οι 10.0.2.15 και 10.0.2.4 αντίστοιχα, η μεν πρώτη ίδια με πριν, ενώ η δεύτερη διαφορετική.

6.5) DHCP IPv4: 10.0.2.3.

6.6) Για το κάθε μηχάνημα, το περιεχόμενο φαίνεται παρακάτω:

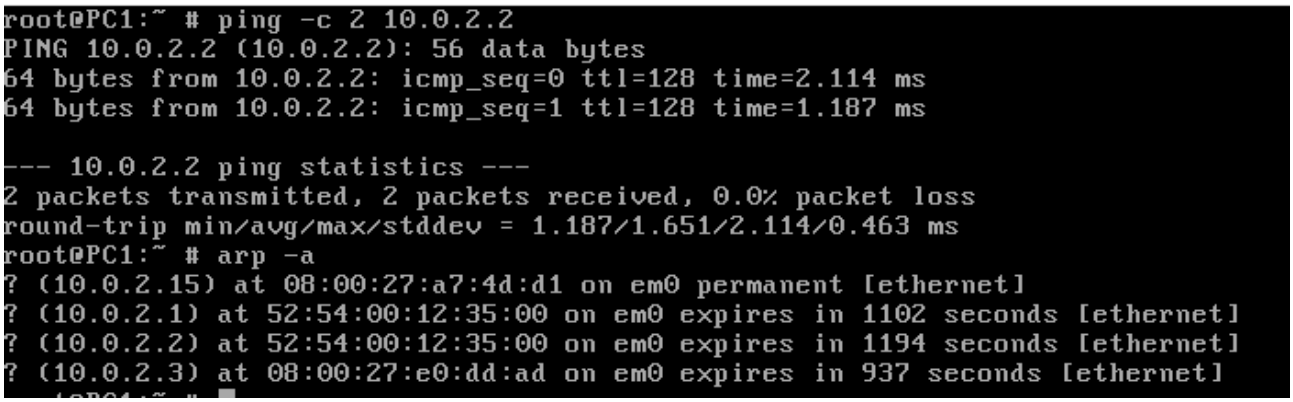


6.7) Προκαθορισμένη πύλη είναι η “10.0.2.1”.

6.8) Ναι, μπορούμε κανονικά.

6.9) Επίσης μπορούμε κανονικά.

6.10) Μπορούμε να κάνουμε κανονικά ping στην “10.0.2.2”. Μάλιστα, παρατηρούμε πως πρόκειται στην πραγματικότητα για την “συσκευή” που αποτελεί την προκαθορισμένη πύλη, αφού από τον πίνακα arp βλέπουμε πως η 10.0.2.1 και 10.0.2.2 έχουν ίδιες MAC διευθύνσεις.



6.11) Τα μηχανήματα επικοινωνούν κανονικά με το Internet (π.χ. ping www.google.com), πράγμα λογικό μιας και έχουν gateway router για να κάνει τις απαραίτητες δρομολογήσεις.

6.12) Ναι, επικοινωνούν.

6.13) Δοκιμάζοντας να κάνουμε ping στο PC2, φαίνεται πως λαμβάνουμε κανονικά απάντηση. Το PC1 εν προκειμένω έχει ίδια IPv4 με το PC3 επομένως δε μπορούμε να το εξετάσουμε.

6.14) Στην πραγματικότητα,βλέποντας τη MAC που είναι αποθηκευμένη στον ARP πίνακα για τη διεύθυνση 10.0.2.4 (PC2), παρατηρούμε πως είναι διαφορετική από αυτή που πραγματικά έχει το PC2, δε μπορούμε από το PC3 να κάνουμε Ping στο PC2 (και λογικά ούτε στο 1).