

Εργαστήριο Δικτύων Υπολογιστών

Εργαστηριακή ασκηση **2**

Δικτyωση συστηaτων στο virtual box

Κουστένης Χρίστος | el20227 | 27/02/2024

# Άσκηση 1: Προετοιμασία

# Άσκηση 2: Ανάλυση δικτυακών πρωτοκόλλων με το TCPDUMP

### 2.1

**ifconfig -a**

### 2.2

**ifconfig em0 down** ---> Απενεργοποίηση

**ifconfig em0 up** ---> Ενεργοποίηση

### 2.3

**man tcpdump**

**man pcap**

**man pcap-filter**

### 2.4

**tcpdump -i em0 -n**

### 2.5

**tcpdump -i em0 -X**

### 2.6

**tcpdump -e** ---> τυπώνουμε επιπλέον την επικεφαλίδα ethernet, άρα με την εντολή

### 2.7

**tcpdump -i em0 -s 68**

### 2.8

**tcpdump ip and host 10.0.0.1 -v**

### 2.9

**tcpdump -i em0 host 10.0.0.1 and 10.0.0.2**

### 2.10

**tcpdump ip and net 1.1.0.0/16**

### 2.11

**tcpdump ip and not net 192.168.1.0/24 -e**

### 2.12

**tcpdump ip broadcast or multicast**

### 2.13

**tcpdump ip and greater 576**

### 2.14

**tcpdump ‘ip[8] < 5’**

### 2.15

**tcpdump ‘(ip[0] & 0x0f) > 5’**

### 2.16

**tcpdump icmp and src 10.0.0.1**

### 2.17

**tcpdump tcp and dst 10.0.0.2**

### 2.18

**tcpdump udp and dst port 53**

### 2.19

**tcpdump tcp and host 10.0.0.10**

### 2.20

**tcpdump tcp and host 10.0.0.10 and port 23 -w sample\_capture**

### 2.21

**tcpdump ‘(tcp[13] & 0x3f) = 0x02’** *(Tcp flags στο 13ο byte και φιλτράρουμε τα 6 τελευταία bits που είναι οι σημαίες και ελέγχουμε να υπάρχει μόνο η SYN.)*

### 2.22

**tcpdump ‘tcp[tcpflags] & ((tcp-syn) | (tcp-syn & tcp-ack)) != 0’**

### 2.23

**tcpdump ‘tcp[tcpflags] & (tcp-fin) != 0’**

### 2.24

Αρχικά, η παράσταση tcp[12:1] μας δίνει τα 8 bits του 13ου Byte μιας TCP επικεφαλίδας.

Στη συνέχεια, η έκφραση tcp[12:1] & 0xf0 μας 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 13ο byte το 01010001, τότε, από τα 4 αριστερότερα bits συμπεραίνουμε ότι το μήκος της επικεφαλίδας είναι 0101 = 510 \* 4 bytes = 20 bytes, ενώ αν εφαρμόσουμε το φίλτρο τότε το byte αυτό μετατρέπεται σε 00010100 = 2010.

### 2.25

**tcpdump ‘(tcp[12] & 0xf0) > 5’**

### 2.26

**tcpdump -A port 80**

### 2.27

**tcpdump port 23 and dst edu-dy.cn.ntua.gr**

### 2.28

**tcpdump ip6**

# Άσκηση 3: Δικτύωση Host-only

### 3.1

Host-Only adapter IPv4 Address : **192.168.145.1**

### 3.2

DHCP Server IPv4 Address : **192.168.145.2**

Lower Address Bound : **192.168.145.3**

Upper Address Bound : **192.168.145.254**

### 3.3

**dhclient em0**

### **3.4**

PC1 ---> **192.168.145.101**

PC2 ---> **192.168.145.102**

### **3.5**

Κάνουμε ping από το PC1 μηχάνημα στο PC2(και αντιστρόφως) και λαμβάνουμε απάντηση.

Στο PC1 : **ping -c 5 192.168.145.101**

Στο PC2 : **ping -c 5 192.168.145.102**

### **3.6**

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

**ping 192.168.145.101**

**ping 192.168.145.102**

### **3.7**

**netstat -r**

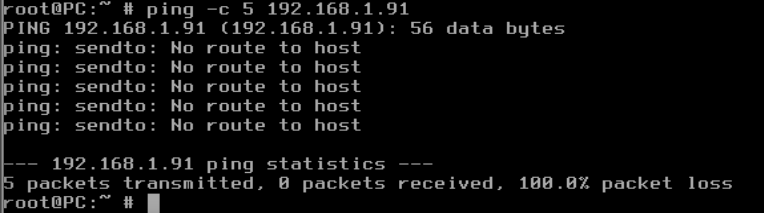
### 3.8



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

### 3.9

Επιχειρούμε να κάνουμε ping στη διεύθυνση IPv4 της φυσικής κάρτας δικτύου του υπολογιστή μας.



Όπως αναμενόταν, δε λαμβάνουμε απάντηση. Αυτό συμβαίνει γιατί τα VMs(PC1 και PC2) ανήκουν σε διαφορειτκό δίκτυο από τη φυσική μας κάρτα. Στο Host-Only δίκτυο που ανήκουν επικοινωνούν μόνο μεταξύ τους καθώς και με την εικονική κάρτα του Host. Αν το host machine θέλει να επικοινωνήσει με τα VMs το κάνει με χρήση της Virtual κάρτας δικτύου και όχι της φυσικής.

### 3.10

**hostname** ---> PC.ntua.lab (και για τα δύο μηχανήματα)

### 3.11

**hostname PC1** --->



**hostname PC2** --->



### 3.12

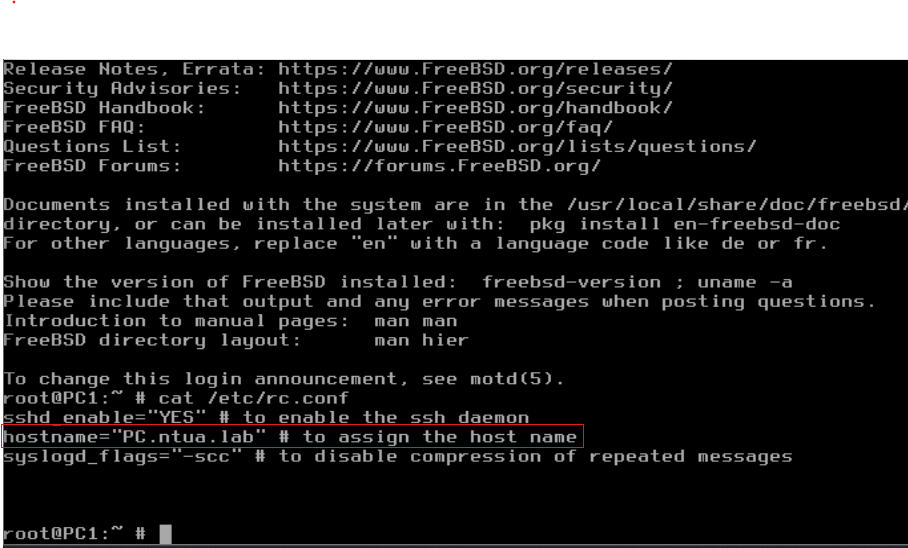
Η αλλαγή φαίνεται στο prompt





### 3.13

**cat /etc/rc.conf** --->



Όχι, δε το περιέχει, αντ’ αυτού περιέχει το « PC.ntua.lab », άρα αυτό θα είναι το όνομα του PC1 σε ενδεχόμενη επανεκκίνηση.

### 3.14

**vi /etc/rc.conf** ---> Διορθώνουμε το πεδίο « hostname ».

<ESC> ---> : ---> wq ---> <ENTER> ---> Αποθήκευση του αρχείου.

### 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 PC2 τη γραμμή « **192.168.56.101 PC1 myPC1.local** », ενώ στου PC1 τη γραμμή « **192.168.56.102 PC2 myPC2.local** ».

### 3.16

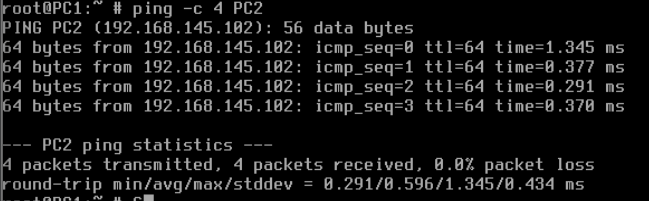
**ping -c 5 PC2** *(από το PC1)*

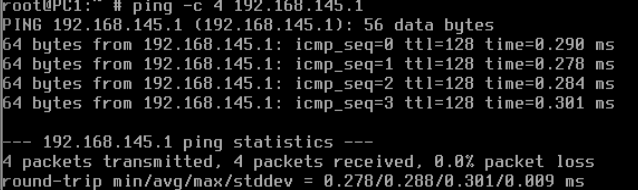
### 3.17

**ping -c 5 PC2** (TTL = 64)

**ping -c 5 192.168.145.1** (TTL = 128)

**ping -c 5 192.168.145.2**

****

****

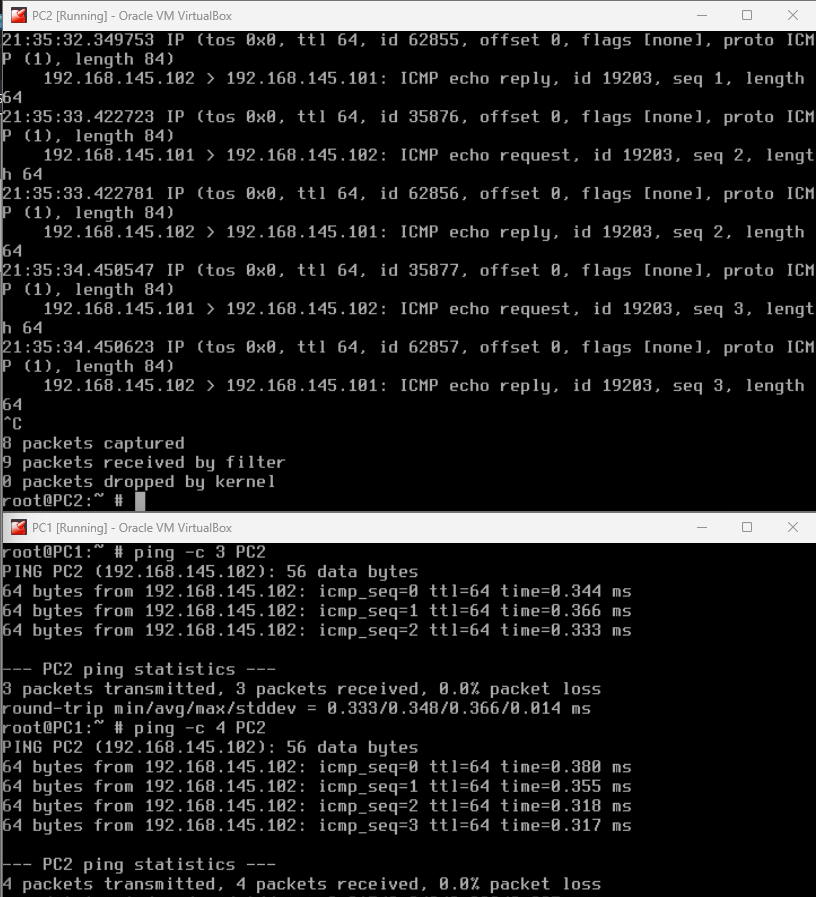
### 3.18

**tcpdump ip and host 192.168.145.101 -v -n**

### 3.19

Length : 64 bytes

TTL : 64



### 3.20

**tcpdump icmp -vvv**

### 3.21

Total Length : 40 bytes

Το φιλοξενούν μηχάνημα αναφέρει πως παράγει 32 bytes requests, 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

Request from host : TTL = 64

Reply from PC2 : TTL = 128

Nαι, συμφωνούν.

### 3.23

**tcpdump -l | tee dat**

**tcpdump -l > dat & tail-f dat**

### 3.24

Όχι.

### 3.25

Όχι.

### 3.26

Αυτή τη φορά, παρατηρούμε κίνηση σα να είμαστε το PC2 δηλαδή τα icmp πακέτα καταγράφονται κανονικά.

# Άσκηση 4: Δικτύωση Internal

### 4.1

ifconfig em0 <IPv4 address>/24

### 4.2

Ενημερώνει για την αποδέσμευση της δυναμικά καταχωρημένης διεύθυνσης IP από τον DHCP Server.

### 4.3

**tcpdump -vv**

### 4.4

Όχι.

### 4.5

Ναι.

### 4.6

Όχι.

### 4.7

Όχι.

### 4.8

Ναι, όπως διαπιστώνουμε έπειτα από επιτυχή pings μεταξύ τους.

### 4.9

Tο φιλοξενούν μηχάνημα αδυνατεί να επικοινωνήσει με οποιοδήποτε από τα μηχανήματα όπως και ήταν αναμενόμενο γεγονός που διαπιστώθηκε έπειτα από απόπειρα ping στις IPs καθενός από αυτά. Αυτό συμβαίνει, γιατί με τη δικτύωση Internal Network δημιουργούμε ένα εικονικό ιδιωτικό LAN δίκτυο για τα VMs μας, χωρίς να υπάρχει δυνατότητα επικοινωνίας με τον host ούτε με το διαδίκτυο, αφού η εικονική διεπαφή που διαθέτει ο host δεν είναι στο δίκτυο αυτό.

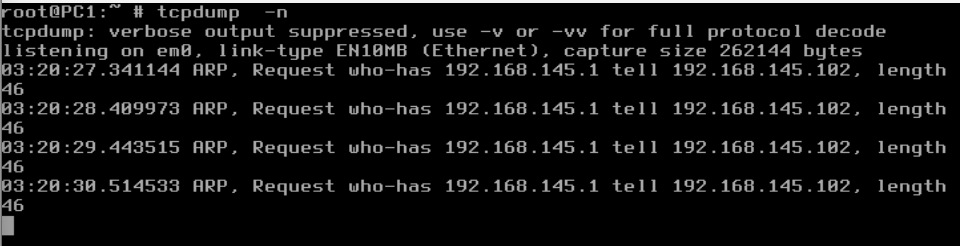
### 4.10

**tcpdump -n**

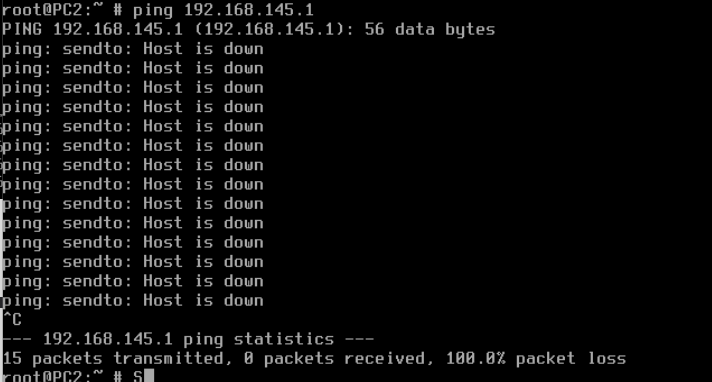
### 4.11

**arp -ad** --->Αδειάζουμε τον πίνακα arp του PC2.

**ping 192.168.145.1** ---> Παράγονται μηνύματα τύπου ARP request, δηλαδή ο PC2 ψάχνει την MAC address της διεύθυνσης 192.168.56.1.



### 4.12



Δεν επικοινωνεί το VM με τον host οπότε νομίζει ότι δεν είναι ενεργό.

### 4.13

Subnet Mask Decimal : 26 => Subnet Mask Binary Octets : 11111111 11111111 11111111 11000000

=> Subnet Mask Decimal Octets : 255 255 255 192

Για να βρούμε τη *διεύθυνση έναρξης* στην ακόλουθη μάσκα υποδικτύου, απλώς κάνουμε « ΑΝD » operation μεταξύ της διεύθυνσης IP και της μάσκας υποδικτύου.

Yπολογίζουμε την *τελευταία διεύθυνση IP* εφαρμόζοντας « OR » operation σε αυτήν με το δυαδικό αντίστροφο bit της μάσκας υποδικτύου στην πρώτη διεύθυνση IP.

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

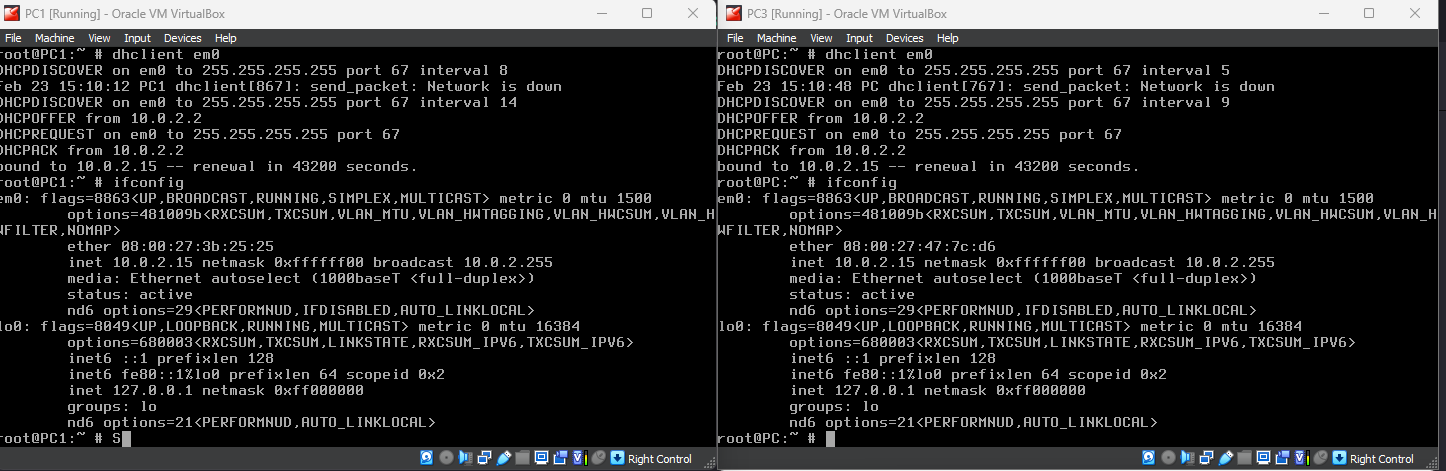
### 4.14

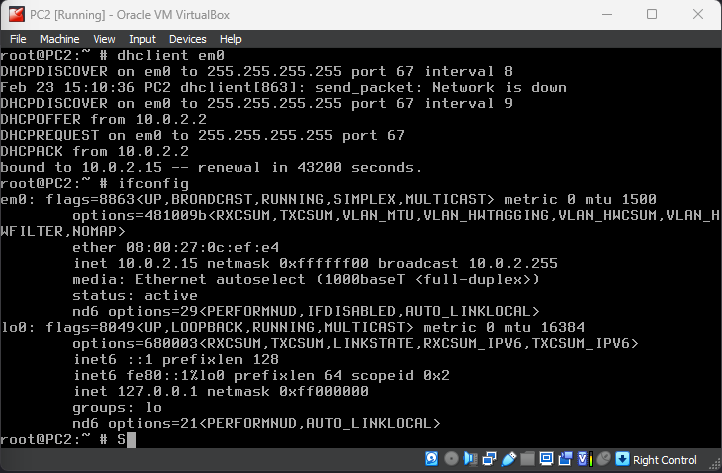
Τα μηχανήματα συνεχίζουν να επικοινωνούν κανονικά όπως διαπιστώνουμε με τη χρήση pings.

# Άσκηση 5: Δικτύωση NAT

### 5.1

**dhclient em0**





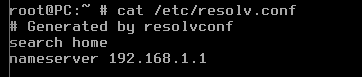
### 5.2

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

### 5.3

**netstat -r** --->προεπιλεγμένη πύλη είναι η 10.0.2.2

### 5.4



### 5.5

Στο [/var/db/dhclient.leases.em0](https://man.freebsd.org/cgi/man.cgi?query=dhclient.leases&sektion=5&manpath=FreeBSD+13.2-STABLE)

### 5.6

Ναι.

### 5.7

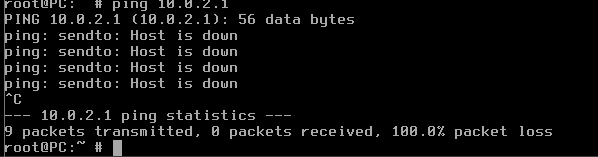
Ναι και το διαπιστώσαμε κάνοντας σε μια διεύθυνση εξωτερική του δικτύου από την οποία λάβαμε απάντηση.

**ping -c 4 www.chess.com**

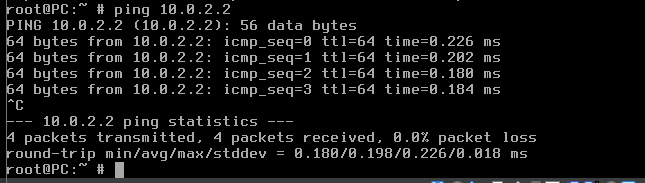
### 5.8

Παρατηρήσαμε τα εξής:

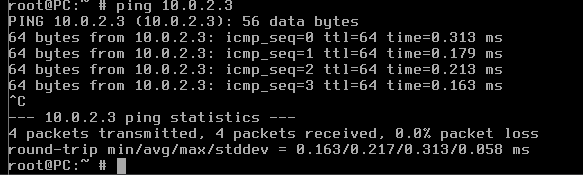
* 10.0.2.1 (δε λαμβάνουμε απάντηση)



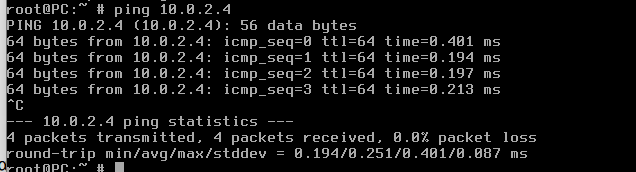
* 10.0.2.2 (λαμβάνουμε απάντηση – default gateway)



* 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

* -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 192.168.1.91, δηλαδή αυτή του υπολογιστή μας (host).

### 5.13

Sources of “TTL exceeded in transit” packets : (Wireshark)

* 192.168.1.1
* 80.106.125.100
* 79.128.230.51
* 79.128.230.32
* 79.128.226.2
* 176.126.38.118

### 5.14

Destination of “TTL exceeded in transit” packets : 192.168.1.1 (Wireshark)

### 5.15

Sources of “TTL exceeded in transit” packets : (tcpdump)

* 10.0.2.2
* 192.168.1.1
* 80.106.125.100
* 79.128.230.51
* 79.128.230.32
* 79.128.226.2
* 176.126.38.118

### 5.16

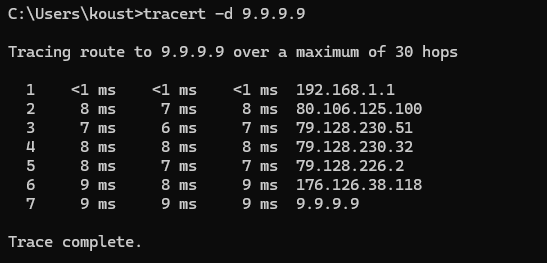
Destination of “TTL exceeded in transit” packets : 10.0.2.15 (tcpdump)

### 5.17

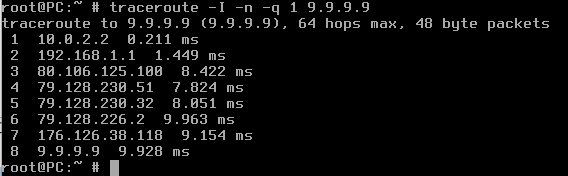
Δεν υπάρχει 1 προς 1 αντιστοίχηση, καθώς στο tcdump καταγράφηκε ένα επιπλέον τέτοιο μήνυμα από την 10.0.2.2 .

### 5.18

Φιλοξενούν μηχάνημα :



PC3 :



Παρατηρούμε ότι στο εικονικό μηχάνημα προκύπτει ένα hop παραπάνω.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

10.0.2.0/24

### 6.2

**ifconfig em0 delete**

**rm /var/db/dhclient.leases.em0**

### 6.3

**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

*nameserver 192.168.1.1*

### 6.7

**netstat -r** --->Προκαθορισμένη πύλη είναι η **10.0.2.1 .**

### 6.8

Ναι, μπορούμε.

### 6.9

Ναι, μπορούμε.

### 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

Ναι, γιατί λαμβάνουμε απάντηση στο **ping** **www.google.com**

### 6.12

Ναι επικοινωνούν.

### 6.13

Όχι, γιατί δεν έχουν τον ίδιο τρόπο δικτύωσης. Ωστόσο μπορούμε να κάνουμε ping στην IP address 10.0.2.4 δεν απαντά όμως το PC2 αλλά ο tftp server της NAT δικτύωσης του PC3.

### 6.14

Τρέχοντας tcpdump στα PC στα οποία κάνουμε ping για να δούμε αν λαμβάνουν τα αντίστοιχα ICMP πακέτα.