Ονοματεπώνυμο: Ανδρέας Στάμος (03120***)	**Όνομα PC:** linux / Ubuntu 22.04.2 LTS
Ομάδα: 1	**Ημερομηνία:** 20/02/2024

Εργαστηριακή Άσκηση 2
Δικτύωση συστημάτων στο VirtualBox

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

Άσκηση 1

Προετοιμασία. Δεν υπάρχουν ερωτήσεις για απάντηση

Άσκηση 2

2.1 `ifconfig`

2.2 `ifconfig em0 down` και έπειτα `ifconfig em0 up`

2.3 `man tcpdump`, `man pcap` και `man pcap-filter`

2.4 `tcpdump -i em0 -n`

2.5 `tcpdump -i em0 -n -X`

2.6 `tcpdump -i em0 -n -e`

2.7 `tcpdump -i em0 -n -s 68`

2.8 `tcpdump -i em0 -n -e 'host 10.0.0.1'`

2.9 `tcpdump -i em0 -n 'host 10.0.0.1 and host 10.0.0.2'`

2.10 `tcpdump -i em0 -n 'net 1.1.0.0/16'`

2.11 `tcpdump -i em0 -n 'not net 10.0.2'`

2.12 `tcpdump -i em0 -n 'ip multicast or ip broadcast'`

2.13 `tcpdump -i em0 -n 'len > 576'`

2.14 `tcpdump -i em0 -n 'ip[8] < 5'`

2.15 `tcpdump -i em0 -n '(ip[0] & 0x0f) > 5'`

2.16 `tcpdump -i em0 -n 'icmp and src host 10.0.0.1'`

2.17 `tcpdump -i em0 -n 'tcp and dst host 10.0.0.2'`

2.18 `tcpdump -i em0 -n 'udp dst port 53'`

2.19 `tcpdump -i em0 -n 'tcp and host 10.0.0.10'`

2.20 `tcpdump -i em0 -n -w sample_capture 'tcp port 23'`

2.21 `tcpdump -i em0 -n 'tcp[tcpflags] & tcp-syn != 0'`

2.22 `tcpdump -i em0 -n 'tcp[tcpflags] & tcp-syn != 0 or (tcp[tcpflags] & tcp-syn != 0 and tcp[tcpflags] & tcp-ack != 0)'`

2.23 `tcpdump -i em0 -n 'tcp[tcpflags] & tcp-fin != 0 or tcp[tcpflags] & tcp-rst != 0'`

2.24 Το τετραπλάσιο του high nibble του 12ου byte (0-indexed) του TCP πακέτου, δηλαδή το μέγεθος της TCP επικεφαλίδας εκφρασμένο σε bytes. Αυτό συμβαίνει διότι το high nibble του 12ου byte είναι το πεδίο data offset, που είναι το μέγεθος της TCP επικεφαλίδας σε πλήθος 4-byte λέξεων και, επίσης, διότι το πέδιο Reserved, τουλάχιστον ως σήμερα, έχει τιμή 0.

2.25 `tcpdump -i em0 -n '(tcp[12:1] & 0xf0) >> 2 > 5'`

2.26 `tcpdump -i em0 -n -A 'tcp port 80'`

2.27 `tcpdump -i em0 -n 'tcp port 23 and dst host edu-dy.cn.ntua.gr'`

2.28 `tcpdump -i em0 -n 'ip6'`

Άσκηση 3

3.1 192.168.56.1

3.2 DHCP Server: 192.168.56.100 και εκχωρεί διευθύνσεις στην περιοχή [192.168.56.101, 192.168.56.254].

3.3 `dhclient em0` και στο PC1 και στο PC2

3.4 PC1: 192.168.56.101

PC2: 192.168.56.102

3.5 PC1: `ping 192.168.56.102` Επιτυγχάνει

PC2: `ping 192.168.56.101` Επιτυγχάνει

Συνεπώς τα δύο μηχανήματα επικοινωνούν μεταξύ τους.

3.6 Host: `ping 192.168.56.101` Επιτυγχάνει

Host: `ping 192.168.56.102` Επιτυγχάνει

3.7 `netstat -r` για να δούμε τους πίνακες δρομολόγησης.

3.8 Δεν υπάρχει. Ακόμα και αν ενεργοποιούσαμε το IP Forwarding στον host (σε Linux με `sysctl -w net.ipv4.ip_forward=1`) και αν βάζαμε ως default gateway την 192.168.56.1 δηλαδή τον Host, ναι μεν ο Host θα προωθούσε τα πακέτα από τον guest προς το εξωτερικό δίκτυο, όμως η διεύθυνση προέλευσης στο IP πακέτο θα ήταν 192.168.56.1 και έτσι ο τελικός παραλήπτης του πακέτου δεν θα μπορούσε να στείλει απάντηση προς τα πίσω. Συνεπώς θα μπορούσαμε να έχουμε μόνο μονόδρομη επικοινωία προς το εξωτερικό δίκτυο. Στην πραγματικότητα, στην περίπτωση αυτή θα έπρεπε να ρυθμίσουμε τον host να υλοποιεί NAT. Επειδή σκοπός του host-only network δεν είναι τίποτα από αυτά, δεν τίθεται default gateway (ο DHCP server δεν τον δίνει).

3.9 Όχι. Το FreeBSD βγάζει σφάλμα `no route to host` καθώς δεν έχει εγγραφή στους πίνακες δρομολόγησης για πρόθεμα εκτός του 192.168.56.0/24 και 127.0.0.1.

Πάραυτα, αν ενεργοποιήσουμε το IP forwarding στον host και χειροκίνητα προσθέσουμε default gateway με `route add default 192.168.56.1` το ping προς την IPv4 διεύθυνση της φυσικής κάρτας του host επιτυγχάνει. Ο λόγος που θα συμβεί αυτό (εν αντιθέσει με όσα λέχθηκαν στην ερώτηση 3.8) είναι πως ο host όταν λάβει το IP πακέτο με διεύθυνση προέλευσης 192.168.56.1, γνωρίζει πως ο guest βρίσκεται στην διεύθυνση αυτή (εν αντιθέσει με οποιονδήποτε άλλον κόμβο εκτός του Host – προσβάσιμο μέσω της φυσικής του κάρτα δικτύοθ).

3.10 Με `hostname` βλέπουμε ότι είναι `PC.ntua.lab` και στο PC1 και στο PC2.

3.11 PC1: `hostname PC1`

PC2: `hostname PC2`

3.12 Το prompt του φλοίου είναι:

PC1: `lab@PC1:~ %`

PC2: `lab@PC2:~ %`

3.13 Όχι. Αν γίνει επανεκκίνηση το hostname θα είναι `PC.ntua.lab`

3.14 Στο `/etc/rc.conf` αντικαθιστούμε το `PC.ntua.lab` με `PC1` και `PC2` στα PC1 και PC2 αντίστοιχα.

3.15 Προσθέτουμε και στο PC1 και στο PC2 στο αρχείο `/etc/hosts` τις γραμμές:

```
192.168.56.101 PC1
192.168.56.102 PC2
```

3.16 Στο PC1: `ping PC2`

3.17 Το μήκος είναι σταθερά 1 byte.

PC2 → PC1: TTL = 64

Host (192.168.56.1) → PC1: TTL = 64

DHCP εξυπηρετητής (192.168.56.100) → PC1: TTL = 255

3.18 `tcpdump -v -e -n -i em0 'host PC2'`

3.19 84 bytes μήκος πακέτου και TTL=64

3.20 `tcpdump -v -e -n -i em0 'icmp'`

3.21 Ίδιο είναι, αλλά ο host τρέχει Linux. Ενδεχομένως ένας host που τρέχει Windows να έστελνε διαφορετικό ICMP Echo Request.

Γενικά εξαρτάται από την υλοποίηση του `ping` στον host (και βασικά από τα default settings του).

3.22 TTL = 64. Είναι ίδια με προηγουμένως.

Είναι λογικό ότι όλα τα μηχανήματα έχουν ίδιο TTL και ίσο με την αρχική τιμή του, καθώς βρίσκονται στο ίδιο υποδίκτυο, οπότε δεν παρεμβάλλεται κάποιος δρομολογητής για να μειώσει το TTL.

3.23 1ος τρόπος:

```
tcpdump -i em0 -l -w - | tee capture_file | tcpdump -r -
```

2ος τρόπος:

```
tcpdump -i em0 -l -w - >capture_file &
tail -f capture_file | tcpdump -r -
```

3.24 Όχι.

3.25 Όχι.

3.26 Το πλαίσιο με διεύθυνση προορισμού το PC2, καταγράφαται και στην καταγραφή του PC1.

Αυτό συμβαίνει διότι PC1 και PC2 βρίσκονται στο ίδιο υποδίκτυο και διότι, λόγω του promiscuous mode, το VirtualBox προωθεί όλα τα πλαίσια του υποδικτύου σε όλα τα μηχανήματα.

Άσκηση 4

4.1 Αρχικά, τερματίζουμε τον DHCP πελάτη και στο PC1 και στο PC2 με: `killall dhclient` και διαγράφουμε τις τρέχουσες διευθύνσεις με: `ifconfig em0 delete`

PC1: `ifconfig em0 192.168.56.101/24`

PC2: `ifconfig em0 192.168.56.102/24`

4.2 Δεν εμφανίστηκε λάθος. Όμως η εκφώνηση μάλλον αναφέρεται στο λάθος που θα βλέπαμε αν δεν ορίζαμε το τμήμα υποδικτύου `/24`, οπότε το `ifconfig` μας ενημερώνει πως επέλεξε με κάποιο default τρόπο το τμήμα υποδικτύου, αλλά πως αυτό είναι deprecated. (λογικό αφού το τμήμα υποδικτύου μπορεί γενικά να έχει αυθαίρετο μέγεθος)

4.3 `tcpdump -i em0 -n -e -v`

4.4 Όχι.

4.5 Ναι τα ARP Requests του host για την μετάφραση της IPv4 διεύθυνσης 192.168.56.102 σε MAC διεύθυνση.

4.6 Όχι.

4.7 Όχι.

4.8 Ναι. Εκτελώντας στο PC1 `ping PC2` επιτυγχάνει και εκτελώντας στο PC2 `ping PC1` επιτυγχάνει επίσης.

4.9 Όχι. Τα PC1, PC2 είναι συνδεδεμένα στο ίδιο εικονικό υποδίκτυο με όνομα `LAN` οπότε μπορούν να επικοινωνήσουν μεταξύ τους. Όμως, το εικονικό υποδίκτυο αυτό είναι απομονωμένο ακόμα και από τον host, είναι ορατό μόνο από τα εικονικά μηχανήματα που είναι συνδεδεμένα σε αυτό.

4.10 `tcpdump -i em0 -n -e -v`

4.11 Παράγει ARP Requests για την μετάφραση της IPv4 192.168.56.1 σε MAC διεύθυνση.

4.12 Σε κανένα ARP Requests δεν έρχεται απάντηση, οπότε μετά από λίγο χρονικό διάστημα το ΛΣ χαρακτηρίζει το μηχάνημα με την IPv4 διεύθυνσή αυτή ως down, αφού δεν απαντά.

4.13 PC1: `ifconfig em0 10.11.12.61/26`

PC2: `ifconfig em0 10.11.12.62/26`

4.14 Ναι επικοινωνούν.

Άσκηση 5

5.1 Εκτελώ στα PC1, PC2, PC3: `dhclient em0`

5.2 Και στα τρία μηχανήματα PC1, PC2, PC3 έχει αποδοθεί η IPv4 διεύθυνση: 10.0.2.15 από DHCP εξυπηρετητή με IPv4 διεύθυνση 10.0.2.2

5.3 Με `netstat -r` βλέπουμε ότι και στα τρία PC1, PC2, PC3 η προεπιλεγμένη πυλή είναι η IPv4 διεύθυνση: 10.0.2.2

5.4 Το `/etc/resolv.conf` έχει περιεχόμενο:

```
search ntua.gr
nameserver 10.0.2.3
```

5.5 Στο `/var/db/dhclient.leases.em0`

5.6 Ναι και από τα τρία.

5.7 Εκτελούμε `ping google.com` και επιτυγχάνει. Αυτό που συμβαίνει είναι πως ο δρομολογητής του VirtualBox στα IP πακέτα που εξέρχονται από το δίκτυο αλλάζει την διεύθυνση προέλευσης στην φυσική διεύθυνσή του ενώ στα πακέτα που εισέρχονται αλλάζει την διεύθυνση προορισμού στην διεύθυνση του μηχανήματος για το οποίο προορίζονται. Προκειμένου να ξεχωρίσει το NAT τα πακέτα προς ποιο μηχάνημα απευθύνονται, στην πραγματικότητα λειτουργεί και στο στρώμα μεταφορά και αλλάζει κατάλληλα τον αριθμό θύρας σημειώνοντας ποιος αριθμός θύρας αντιστοιχεί σε ποιο μηχάνημα. Στο ICMP, αντί για αριθμό θύρας χρησιμοποιείται το Identifier ως αναγνωριστικό.

5.8 Απάντανε οι εξής, που έχουν και τους ακόλουθους ρόλους:

10.0.2.2 Προκαθορισμένη πύλη και DHCP εξυπηρετητής

10.0.2.3 DNS εξυπηρετητής

10.0.2.4 TFTP εξυπηρετητής για δικτυακό boot του guest αν αυτό θεωρείται επιθυμητό.

5.9 Όχι. Κάθε εικονικό μηχάνημα έχει το δικό του NAT δίκτυο με το VirtualBox να αναλαμβάνει να κάνει την μετάφραση χωριστά για κάθε δίκτυο μηχανήματος. Εξάλλου, ακόμα και αν θέλαμε να δοκιμάσουμε αν τυχόν επικοινωνούσαν τα μηχανήματα, δεν θα μπορούσαμε διότι έχουν όλα την ίδια IPv4 διεύθυνση — βέβαια θα μπόρουσε κάποιος να ισχυριστεί ότι θα μπορούσαμε να δοκιμάσουμε με τις MAC, αλλά αυτό δεν θα είχε νόημα διότι ακόμα και αν μια τέτοια επικοινωνία θεωρητικά επιτύγχανε, που θα αποτύγχανε δηλαδή, το στρώμα δικτύου θα αδυνατούσε μετά και πάλι να λειτουργήσει.

5.10 **-I** Χρησιμοποιεί ICMP Echo Requests.

-n Εκτυπώνει IP διεύθυνσεις χωρίς να τις μετατρέψει σε ονόματα με reverse DNS lookup.

-q 1 Για κάθε τιμή TTL (δηλαδή για κάθε "άλμα/βήμα/hop δρομολόγησης") στέλνει 1 πακέτο.

5.11 Ο τύπος ICMP μηνύματος που παράγεται από την `traceroute` είναι Echo Request και η διεύθυνση IPv4 της πηγής είναι 10.0.2.15

5.12 Ο τύπος ICMP μηνύματος που παράγεται από την `traceroute` είναι Echo Request και η διεύθυνση IPv4 της πηγής είναι η IPv4 διεύθυνση της φυσικής κάρτας δικτύου του host.

5.13

1. 192.168.1.1
2. 80.106.125.100
3. 79.128.226.161
4. 79.128.226.2
5. 176.126.38.118

5.14 Η IPv4 διεύθυνση της φυσικής κάρτας δικτύου του host.

5.15
1. 192.168.1.1
2. 80.106.125.100
3. 79.128.226.161
4. 79.128.226.2
5. 176.126.38.118

5.16 10.0.2.15

5.17 Όχι. Στο PC3 υπάρχει ένα έξτρα ICMP Time Exceeded μήνυμα από την IPv4 διεύθυνση 10.0.2.2 (που αναλαμβάνει την δρομολόγηση και την μεταφράση διευθύνσεων στο δίκτυο του VirtualBox για το PC3)

5.18 Ο γράφων χρησιμοποιεί Linux, οπότε τρέχει την ίδια εντολή με πριν: `traceroute -I -n -q 1 9.9.9.9`

Παρατηρούμε πως στον host είναι 1 λιγότερο hop σε σχέση με τον guest.

Αυτό συμβαίνει καθώς στον guest απαιτείται ένα παράπανω hop, εκείνο του δρομολογητή του VirtualBox για το δίκτυο του PC3.

Άσκηση 6

6.1 10.0.2.0/24

6.2 Εκτελώ και στο PC1 και στο PC2:

```
ifconfig em0 delete
rm /var/db/dhclient.leases.em0
```

6.3 Εκτελώ και στο PC1 και στο PC2:

```
dhclient em0
```

6.4 PC1: 10.0.2.4

PC2: 10.0.2.5

Ναι διαφέρουν.

6.5 10.0.2.3

6.6 Το `/etc/resolv.conf` έχει περιεχόμενο:

```
search ntua.gr
nameserver 10.0.2.1
```

6.7 Με `netstat -r` βλέπουμε ότι η προεπιλεγμένη πύλη είναι: 10.0.2.1

6.8 Ναι και από τα δύο μηχανήματα επιτυγχάνει.

6.9 Ναι και από τα δύο μηχανήματα επιτυγχάνει.

6.10 Από τον ARP πίνακα (τον βλέπουμε με `arp -a`) παρατηρούμε ότι στην IPv4 διεύθυνση 10.0.2.2 απαντά η ίδια διεπαφή δικτύου με εκείνη που απαντά στην IPv4 διεύθυνση 10.0.2.1, που γνωριζούμε ότι είναι η προκαθορισμένη πύλη, δηλαδή ο δρομολογητής του VirtualBox για το NAT δίκτυο με όνομα `NatNetwork`.

6.11 Εκτελούμε `ping google.com` και επιτυγχάνει. Αυτό που συμβαίνει είναι πως ο δρομολογητής του VirtualBox στα IP πακέτα που εξέρχονται από το δίκτυο αλλάζει την διεύθυνση προέλευσης στην φυσική διεύθυνσή του ενώ στα πακέτα που εισέρχονται αλλάζει την διεύθυνση προορισμού στην διεύθυνση του μηχανήματος για το οποίο προορίζονται. Προκειμένου να ξεχωρίσει το NAT τα πακέτα προς ποιο μηχάνημα απευθύνονται, στην πραγματικότητα λειτουργεί και στο στρώμα μεταφορά και αλλάζει κατάλληλα τον αριθμό θύρας σημειώνοντας ποιος αριθμός θύρας αντιστοιχεί σε ποιο μηχάνημα. Στο ICMP, αντί για αριθμό θύρας χρησιμοποιείται το Identifier ως αναγνωριστικό.

6.12 Ναι, το ping στο PC1 προς την IPv4 διεύθυνση του PC2 επιτυγχάνει και αντίστροφα.

6.13 Δεν μπορούμε να κάνουμε ping διότι βρίσκονται σε διαφορετικά NAT και έτσι οι εσωτερικές διευθύνσεις του ενός δεν αντιστοιχούν στις εσωτερικές διευθύνσεις του άλλου.

Φαινομενικά, βλέπουμε στο PC3 το ping προς την IPv4 διεύθυνση 10.0.2.4 να επιτυγχάνει, οπότε κάποιος φαινομενικά θα μπορούσε να νομίσει ότι το PC1 απαντά στο PC2. Όμως η διεύθυνση 10.0.2.4 αντιστοιχεί σε διαφορετικά μηχανήματα στα δύο δίκτυα, το οποίο γίνεται διότι βρίσκονται σε διαφορετικό NAT. Πράγματι με `arp -a` στο PC3 βλέπουμε την MAC διεύθυνση που αντιστοιχεί στην IPv4 10.0.2.4 που απαντά στο PC3 και με `ifconfig` στο PC1 βλέπουμε την MAC διεύθυνση της διεπαφής του PC1. Παρατηρούμε πως πράγματι είναι διαφορετικές.

Το ping από το PC3 προς την IPv4 10.0.2.5 του PC2 αποτυγχάνει, όχι απλά διότι δεν μπορούν τα πακέτα του PC3 να φθάσουν στο PC2, αλλά διότι η IPv4 διεύθυνση 10.0.2.5 για τον PC3 σημαίνει ένα μηχάνημα εντός του NAT δικτύου του με αυτή την διεύθυνση, δηλαδή άλλο μηχάνημα από το PC2.

6.14 Εξηγήθηκε στο 6.13. Βλέπουμε με `arp -a` την MAC διεύθυνση του μηχανήματος που απαντά και την συγκρίνουμε με την διευθύνση του μηχανήματος που περιμένουμε ότι απαντά.