



## Εργαστήριο Δικτύων Υπολογιστών

Εργαστηριακή άσκηση 3 (Τοπικά δίκτυα και μεταγωγείς LAN)

Τσάκωνας Παναγιώτης (03119610)

Ομάδα: 2

Ακαδημαϊκό Έτος: 2023-2024

### Άσκηση 1: Γέφυρα - Διασύνδεση δύο LAN

- 1.1) PC1 → `ifconfig em0 inet 192.168.1.1` και PC2 → `ifconfig em0 inet 192.168.1.2`
- 1.2) Οι διεπαφές του B1 θα ενεργοποιηθούν με τις εντολές: `ifconfig em0 up` και `ifconfig em1 up`.
- 1.3) Τα μηχανήματα δεν επικοινωνούν καθώς δεν υπάρχει τρόπος να φτάσουν τα πακέτα από το PC1 στο PC2.
- 1.4) Δεν βλέπουμε πακέτα ICMP επειδή στο ARP request δεν έρχεται ποτέ απάντηση.
- 1.5) `ifconfig bridge0 create` → `ifconfig bridge0 addm em0 addm em1` → `ifconfig bridge0 up`
- 1.6) Ναι τώρα επικοινωνούν.
- 1.7) Φαίνεται να απέχουν 1 βήμα.
- 1.8) Ο πίνακας ARP του αντίστοιχου VM περιέχει τη δική του εγγραφή και την διεύθυνση MAC του άλλου VM, ενώ δεν υπάρχει σε κανένα από τα 2 VMs η διεύθυνση MAC της γέφυρας.
- 1.9) Χρησιμοποιώντας τις εντολές: `tcpdump -i -vvn em0 -e` και `tcpdump -i -vvn em1 -e`, καθώς και στη συνέχεια κάνοντας `ping` από το PC1 στο PC2, επιβεβαιώνουμε ότι το B1 προωθεί τα πλαίσια Ethernet μεταξύ των LAN1 και LAN2.
- 1.10) Όχι, δεν αλλάζει τις διευθύνσεις τους.
- 1.11) Όχι, δεν φαίνεται να αλλάζει κάτι.
- 1.12) Όχι, καθώς ένα bridge απλώς προωθεί το πακέτο χωρίς να κάνει καμία αλλαγή σε αυτό.
- 1.13) PC1 → `ping 192.168.1.2` και στο B1 → `tcpdump -e -vvn -i em1`
- 1.14) PC2 → `ifconfig em0 inet 192.168.2.1` . Η γέφυρα τα προωθεί και δεν σταματάει να στέλνει ICMP echo requests.
- 1.15) Όχι δεν είναι επιτυχές, διότι έχει αλλάξει η ip του PC2 και ο προορισμός του `ping` από το PC1 είναι η παλιά ip.

- 1.16) Όχι, δεν μπορώ, γιατί η διεπαφή em2 δεν έχει προστεθεί στο bridge0.
- 1.17) `ifconfig bridge0 addm em2, ifconfig em2 up`
- 1.18) Ναι λαμβάνω απάντηση από το PC1.
- 1.19) Όχι δεν εμφανίζονται επειδή το bridge ξέρει που να προωθήσει τα πακέτα μεταξύ PC1 και PC3 και αυτή η διαδρομή δεν περιλαμβάνει το LAN2.
- 1.20) Καταγράφεται ένα arp request το οποίο προέρχεται από το PC1 και ζητά ποιος έχει τη διεύθυνση του PC3.
- 1.21) Με χρήση της εντολής: `ifconfig bridge0`, μπορώ να δω ποιες διεπαφές είναι μέλη της γέφυρας bridge0.
- 1.22) Το περιεχόμενο του πίνακα προώθησης της γέφυρας bridge0 μπορώ να το δω με χρήση της εντολής: `ifconfig bridge0 addr`
- 1.23) Αντιστοιχούν στους υπολογιστές που βρίσκονται στο αντίστοιχο LAN (π.χ. em0 → PC1 κ.ο.κ.)
- 1.24) `ifconfig bridge0 flush`
- 1.25) `ifconfig bridge0 deletem em2`
- 1.26) `ifconfig bridge0 destroy`
- 1.27) Για κάθε ένα από τα 3 PCs εκτελούμε την εντολή: `ifconfig em0 delete`

## Άσκηση 2: Αυτο-εκπαίδευση γεφυρών

- 2.1) PC1 → `ifconfig em0 inet 192.168.1.1`, PC2 → `ifconfig em0 inet 192.168.1.2`, PC3 → `ifconfig em0 inet 192.168.1.3` και PC4 → `ifconfig em0 inet 192.168.1.4`
- 2.2) `ifconfig bridge1 create` → `ifconfig bridge1 addm em0 em1` → `ifconfig em0 up` → `ifconfig em1 up` → `ifconfig bridge1 up`
- 2.3) `ifconfig bridge2 create` → `ifconfig bridge2 addm em0 em1` → `ifconfig em0 up` → `ifconfig em1 up` → `ifconfig bridge2 up`
- 2.4) `ifconfig bridge3 create` → `ifconfig bridge3 addm em0 em1` → `ifconfig em0 up` → `ifconfig em1 up` → `ifconfig bridge3 up`

- 2.5)** PC1 → 08:00:27:b9:35:ab, PC2 → 08:00:27:5b:98:e4, PC3 → 08:00:27:d1:27:ab και PC4 → 08:00:27:5e:53:fc . Άδειασμα με χρήση της εντολής: `arp -d -a`
- 2.6)** `ifconfig bridge1 flush`, `ifconfig bridge2 flush` και `ifconfig bridge3 flush` για κάθε γέφυρα
- 2.7)** –
- 2.8)** Αφού βεβαιώθηκε ότι οι πίνακες προώθησης των B1, B2 και B3 είναι κενοί τότε: `bridge1 → PC1` και `PC2`, `bridge2 → PC1` και `PC2` και `bridge3 → PC1`
- 2.9)** Μέσω του ARP Request του PC1 ενημερώνονται όλες οι γέφυρες για τον PC1. Μέσω του ARP Reply θα ενημερωθούν μόνο οι B1 B2 αφού το πακέτο προωθείται στο LNK1 που συνδέει αυτές τις δύο. Τα ICMP Request και ICMP Reply δεν προσθέτουν κάποια επιπλέον πληροφορία στις γέφυρες και απλά ακολουθούν την σωστή διαδρομή από τον PC1 στον PC2.
- 2.10)** Όχι δεν υπήρξαν αλλαγές, διότι οι γέφυρες θα οδηγήσουν τα μηνύματα προς τον PC1 από το σωστό μονοπάτι και έτσι η B3 δεν θα μάθει ποτέ για τα στοιχεία του PC2.
- 2.11)** `bridge1 → PC1, PC2 και PC3`, `bridge2 → PC1, PC2 και PC3` και `bridge3 → PC1, PC2 και PC3`
- Ο πίνακας προώθησης του B1 περιέχει εγγραφή για το PC4 γιατί όταν το ARP reply από το PC4 εκπέμπεται από το B2 στη θύρα `em0` το λαμβάνει ο B1 και αν και δεν το προωθεί σε άλλη θύρα, το αποθηκεύει στο πίνακα προώθησης
- 2.12)** `bridge1 → PC1, PC2, PC3 και PC4`, `bridge2 → PC1, PC2, PC3 και PC4` και `bridge3 → PC1, PC2, PC3 και PC4`
- Το ARP request από το PC3 λαμβάνεται από τα B2, B3 και επειδή το B2 προωθεί το πακέτο στη θύρα `em0` λαμβάνεται και από το B1. Έτσι, πλέον και οι 3 γέφυρες έχουν εγγραφές για όλα τα PC.
- 2.13)** –
- 2.14)** Σταματάει για λίγο διότι πλέον το B3 δεν βρίσκει το PC2 στη διεπαφή που ήξερε ότι βρίσκεται. Μετά από λίγο ανανεώνεται το routing table του και το προωθεί σωστά οπότε το ping συνεχίζεται.
- 2.15)** Το ping από το PC1 σταματάει γιατί η γέφυρα B2 δεν προωθεί ποτέ το πακέτο πέρα από αυτό το LAN (λόγω του παλιού πίνακα προώθησης) και επομένως το icmp request δεν φτάνει ποτέ στο PC2.
- 2.16)** Το ping αρχίζει και πάλι να λειτουργεί, διότι με το ping από το PC2 ο πίνακας προώθησης του B2 ανανεώθηκε.
- 2.17)** Θα έπρεπε να περιμένουμε μέχρι να λήξει η παλιά εγγραφή στα routing tables και να γίνει καινούρια.

### Άσκηση 3: Καταιγίδα πλαισίων εκπομπής

- 3.1) `ifconfig bridge1 create → ifconfig bridge1 addm em0 em1 → ifconfig em0 up → ifconfig em1 up → ifconfig bridge1 up`
- 3.2) `ifconfig bridge2 create → ifconfig bridge2 addm em0 em2 → ifconfig em0 up → ifconfig em1 up → ifconfig bridge2 up`
- 3.3) PC1 → 08:00:27:b9:35:ab, PC2 → 08:00:27:5b:98:e4 και PC3 → 08:00:27:d1:27:ab. Άδειασμα με χρήση της εντολής: `arp -d -a`.
- 3.4) Εμφανίζει το ARP request καθώς προωθείται το πακέτο αυτό σε όλο το δίκτυο και το λαμβάνει και το PC1 μέσω του B1.
- 3.5) PC3 → `ping 192.168.1.1`
- 3.6) `ifconfig bridge1 addm em2 → ifconfig em2 up → ifconfig bridge 2 addm em1 → ifconfig em1 up`
- 3.7) με χρήση της εντολής: `ifconfig bridge1 addr` και `ifconfig bridge2 addr` βλέπουμε ότι η γέφυρα 1 έχει εγγραφές για τα PC1 και 3 και η γέφυρα 2 έχει εγγραφές για όλα τα PC.
- 3.8) B1 → PC1: em0, PC3: em1 και B2 → PC1: em1, PC3: em0
- 3.9) `tcpdump -e`
- 3.10) Όχι δεν είναι επιτυχές το ping.
- 3.11) Στην em2 εμφανίζονται και οι δύο. Αυτό συνέβη λόγω του broadcast storm (τα πακέτα παγιδεύτηκαν μεταξύ των B1 και B2)
- 3.12) ARP request: who has 192.168.1.1 και ARP reply: 192.168.1.1 is at <08:00:27:b9:35:ab >
- 3.13) Η MAC διεύθυνση πηγής τους είναι: PC3 → 08:00:27:d1:27:ab
- 3.14) Τα πακέτα ARP Request επαναλαμβάνονται συνεχώς σε αμφότερες τις καταγραφές, διότι δημιουργήθηκε βρόχος λόγω της ατέρμονης προώθησης πακέτων εκπομπής από θύρα σε θύρα από τις γέφυρες.
- 3.15) Τα ARP Reply στο LAN1 δεν προωθούνται στο LAN2, διότι στο πίνακα προώθησης του B2 έχει καταγραφεί ότι το PC3 βρίσκεται στη θύρα em2 → LNK2 και έτσι δεν στέλνει ποτέ το ARP reply στη σωστή θύρα.

## Άσκηση 4: Συνάθροιση ζεύξεων

- 4.1) `ifconfig bridge1 destroy → ifconfig em0 down → ifconfig em1 down → ifconfig em2 down → ifconfig bridge1 create`
- `ifconfig bridge2 destroy → ifconfig em0 down → ifconfig em1 down → ifconfig em2 down → ifconfig bridge2 create`
- 4.2) `ifconfig em0 up → ifconfig em1 up → ifconfig em2 up → ifconfig lagg0 create`
- 4.3) `ifconfig lagg0 up laggport em1 → ifconfig lagg0 up laggport em2`
- 4.4) `ifconfig em0 up → ifconfig em1 up → ifconfig em2 up → ifconfig lagg0 create → ifconfig lagg0 up laggport em1 → ifconfig lagg0 up laggport em2`
- 4.5) `ifconfig bridge1 addm em0 addm lagg0 → ifconfig bridge1 up`
- 4.6) `ifconfig bridge2 addm em0 addm lagg0 → ifconfig bridge2 up`
- 4.7) Ναι, εμφανίζεται κανονικά κίνηση.
- 4.8) –
- 4.9) `arp -d -a → ping -c 1 192.168.1.1` . Ναι, είναι επιτυχές και παρατηρώ 1 ARP request και 1 reply.
- 4.10) Εμφανίζεται κίνηση μόνο στο LNK1, καθώς η ζεύξη LNK2 δεν είναι active. Αυτό οφείλεται στο πρωτόκολλο failover που είναι default και ορίζει μόνο τη μία από τις δύο ζεύξεις ως master και την άλλη ως backup.
- 4.11) Η ping εξακολουθεί να επιτυγχάνεται και παρατηρούμε ότι η κίνηση μεταφέρθηκε από το LNK1 στο LNK2.
- 4.12) Η κίνηση επέστρεψε και πάλι στη ζεύξη LNK1.

## Άσκηση 5: Αποφυγή βρόχων

- 5.1) `ifconfig lagg0 destroy → ifconfig bridge1 destroy → ifconfig bridge2 destroy`
- 5.2) `ifconfig bridge1 create → ifconfig bridge1 addm em0 addm em1 addm em2 up`
- 5.3) `ifconfig bridge2 create → ifconfig bridge2 addm em0 addm em1 addm em2 up`
- 5.4) `ifconfig bridge1 stp em0 stp em1 stp em2`
- 5.5) `ifconfig bridge2 stp em0 stp em1 stp em2`
- 5.6) B1 → 08:00:27:35:06:a3 και B2 → 08:00:27:10:c8:c7
- 5.7) Αφού και οι δύο γέφυρες έχουν την ίδια προτεραιότητα, επιλέγεται αυτή με τη μικρότερη MAC διεύθυνση αρά η B2.
- 5.8) `role designated` και `state forwarding` για όλες τις διεπαφές της B2.
- 5.9) Η ριζική θύρα είναι αυτή στο LNK1.
- 5.10) Η κατάσταση και ο ρόλος της άλλης εκ των δύο διεπαφών είναι `alternate` και `discarding`.
- 5.11) Η κατάσταση και ο ρόλος της διεπαφής στο LAN1 ή LAN2 της γέφυρας που δεν είναι η ρίζα είναι `designated` και `forwarding`.
- 5.12) `tcpdump -i em1 -e -vnn` και διαπιστώνουμε ότι κάθε 2” εκπέμπονται BPDUs.
- 5.13) IEEE 802.3
- 5.14) 08:00:27:b8:b3:11 → 01:80:c2:00:00:00
- 5.15) Στην `em0`
- 5.16) Multicast
- 5.17) `root-id: 8000.08.00.27.14.7a.b5    bridge-id: 8000.08.00.27.14.7a.b5    root-pathcost: 0`
- 5.18) Το μέρος πριν την τελειά.
- 5.19) Το δεύτερο μέρος είναι η MAC διεύθυνση.
- 5.20) Τα πρώτα 4 bits είναι το `port priority` και τα επόμενα 12 είναι το `port_id`. Εδώ το `port priority` είναι 8 και το `port id` είναι 2.
- 5.21) Όχι

**5.22)** Στην em0 που είναι designated.

**5.23)** root-id: 8000.08.00.27.14.7a.b5    bridge-id: 8000.08.00.27.14.7a.b5    root-pathost: 0

**5.24)** Ναι είναι επιτυχές

**5.25)** Περίπου 6” και είναι αναμενόμενο

**5.26)** Υπάρχει μια ανεπαίσθητη διακοπή

## **Άσκηση 6: Ένα πιο πολύπλοκο δίκτυο με εναλλακτικές διαδρομές**

**6.1)** ifconfig em0 up → ifconfig em1 up → ifconfig em2 up → ifconfig em3 up → ifconfig bridge1 create  
→ ifconfig bridge1 addm em0 addm em1 addm em2 addm em3 up → ifconfig bridge1 stp em0 stp  
em1 stp em2 stp em3

**6.2)** ifconfig em0 up → ifconfig em1 up → ifconfig em2 up → ifconfig em3 up → ifconfig bridge2 create  
→ ifconfig bridge2 addm em0 addm em1 addm em2 addm em3 up → ifconfig bridge2 stp em0 stp  
em1 stp em2 stp em3

**6.3)** ifconfig bridge3 create → ifconfig bridge3 addm em0 addm em1 addm em2 up → ifconfig bridge3  
stp em0 stp em1 stp em2

**6.4)** Ναι είναι επιτυχές

**6.5)** ifconfig bridge1 priority 0

**6.6)** ifconfig bridge2 → path cost 20000 για όλες, το οποίο είναι το default για 1 Gbps ζεύξη.

**6.7)** Με tcpdump στα αντίστοιχα interfaces προκύπτει root-pathcost 0 για το LNK3 και 20000 για το LNK4,  
κάτι το οποίο είναι λογικό αφού το B1 είναι η ρίζα και το pathcost προς το B2 είναι 20000.

**6.8)** Η ριζική θύρα της bridge3 είναι η LNK3 καθώς αυτή οδηγεί προς τη ρίζα συντομότερα.

**6.9)** Η B3 είναι alternate και discarding ενώ η B2 είναι designated και forwarding.

**6.10)** 20000

**6.11)** ping 192.168.1.3

**6.12)** `ifconfig bridge3 ifpathcost em1 50000` → Επιλέξαμε για κόστος την τιμή 50000 , διότι πρέπει η LNK3 να έχει μεγαλύτερο κόστος από το άθροισμα των άλλων επιμέρους διαδρομών.

**6.13)** Περίπου 3-4”

**6.14)** Η B3 είναι alternate και discarding ενώ η B2 είναι designated και forwarding.

**6.15)** Όχι δεν υπήρξε κάποια διαφορά.

**6.16)** Πλέον στο LAN3 το root-pathcost έγινε 40000.

**6.17)** Περίπου 7”

**6.18)** Περίπου 3-4”

**6.19)** Η προϋπάρχουσα είναι designated και forwarding και η νέα είναι backup και discarding.

**6.20)** Επιλέγω για cost αυτό που είχε πριν (10000) έτσι ώστε root cost < 40000.

## **Άσκηση 7: Εικονικά τοπικά δίκτυα (VLAN)**

**7.1)** `ifconfig em0.5 create up` → `ifconfig em0.6 create up` → `ifconfig em0.5 192.168.5.1 netmask 255.255.255.0` → `ifconfig em0.6 192.168.6.1 netmask 255.255.255.0`

**7.2)** `ifconfig em0.5 create up` → `ifconfig em0.6 create up`

**7.3)** `ifconfig em1.6 create up` → `ifconfig em3.5 create up`

**7.4)** `ifconfig em0.6 create up` → `ifconfig em0.6 192.168.6.2 netmask 255.255.255.0`

**7.5)** `ifconfig em0.6 create up` → `ifconfig em1.6 create up`

**7.6)** `ifconfig em0.5 create up` → `ifconfig em0.5 192.168.5.3 netmask 255.255.255.0`

**7.7)** `ifconfig em0.5 create up` → `ifconfig em1.5 create up`

**7.8)** Όχι, δεν μπορώ να κάνω ping στις διευθύνσεις 192.168.6.2 και 192.168.5.3 από το PC1.

**7.9)** `ifconfig bridge1 -stp em0`

**7.10)** `tcpdump -i em0 -vvv -e -xx`

**7.11)** ARP → 0x0806 και IPv4 → 0x0800



- 7.12)** Τα νέα πλαίσια τώρα έχουν ετικέτα 802.1Q καθώς στη θέση του ethertype βλέπουμε τη τιμή 0x8100.
- 7.13)** Τα πακέτα ξεχωρίζουν από το ethertype. Σε VLAN πακέτα έχει τη τιμή 0x8100.
- 7.14)** Στο ethertype εμφανίζεται η πληροφορία για το VLAN, 4 bytes μετά το πεδίο length.
- 7.15)** `tcpdump -i em0.5 -vvv -e -xx`
- 7.16)** ARP → 0x0806 και IPv4 → 0x0800 και δεν φαίνεται κάποιο πεδίο σχετικό με το VLAN.
- 7.17)** `ifconfig bridge1 stp em0`
- 7.18)** Τα πλαίσια Ethernet είναι διαφορετικά από πριν, πλέον είναι IEEE 802.3, οπότε το Ethertype αντιστοιχεί στο μήκος του payload
- 7.19)** Για να μην συλλαμβάναμε πλαίσια BPDU θα έπρεπε να χρησιμοποιήσουμε το φίλτρο `not stp` στις προηγούμενες καταγραφές στο PC1.