

Ανάλυση Επιστημονικού Άρθρου

Το άρθρο το οποίο επιλέξαμε είναι το “Implementation of Explicit Wireless Loss Notification Using MAC-Layer Information”, του Bing Zhang και του Mahdad. N. Shirazi.

Η ασύρματες τεχνολογίες κατέχουν όλο ένα και περισσότερο σημαντικό ρόλο στην παγκόσμια υποδομή του διαδικτύου, τα πρωτόκολλα του οποίου όμως δεν δημιουργήθηκαν λαμβάνοντας υπόψη τα ασύρματα δίκτυα. Ένα από αυτά τα πρωτόκολλα είναι το TCP. Το TCP είναι ένα πρωτόκολλο επιπέδου μεταφοράς από άκρο σε άκρο, το οποίο σχεδιάστηκε για να επίλυση τα ήδη υπάρχοντα προβλήματα των συνδεδεμένων δικτύων. Όταν χάνεται ένα πακέτο το TCP το αναμεταδίδει και ταυτόχρονα μειώνει το μέγεθος του παραθύρου συμφόρησης και μεταβάλλει το χρονόμετρο αναμετάδοσης. Το TCP όμως θεωρεί πως η συμφόρηση είναι ο κύριος λόγος για χάσιμο πακέτων και καθυστερήσεων, πράγμα που το καθιστά ανίκανο να καταλάβει τα τυχαία συχνά λάθη που υπάρχουν στην ασύρματη επικοινωνία και γιαυτό τον λόγο ο μηχανισμός συμφόρησης οδηγεί σε αχρείαστες μειώσεις στην απόδοση. Έχουν δοθεί αρκετά σχήματα για επίλυση αυτού του προβλήματος με μερικά να αποκρύπτουν πληροφορίες από το TCP για να το χειριστούν άλλα πρωτόκολλα και μερικά να εισάγουν καινούργιους μηχανισμούς στο TCP για να μπορέσει να ξεπεράσει τα παραπάνω προβλήματα. Το άρθρο ασχολείται με την δεύτερη κατηγορία και συγκεκριμένα με το Explicit Wireless Loss Notification(EWLN). Το EWLN είναι ένας μηχανισμός στον οποίο ο λόγος για την απώλεια ενός πακέτου κοινοποιείται στον αποστολέα. Αυτό δίνει την δυνατότητα στον πομπό να καταλάβει αν η αιτία δεν συνδέεται με συμφόρηση, ώστε να μην ενεργοποιηθεί ο μηχανισμός συμφόρησης αν δεν χρειάζεται. Το EWLN έχει ήδη προταθεί για δύο συχνά σενάρια, όπου η πρόσβαση στις πληροφορίες γίνεται είτε από τερματικό μέσω ενσύρματης προς ασύρματη μετάδοση, είτε από κινητό σταθμό μέσω ασύρματης σε ενσύρματη σύνδεση. Στο πρώτο σενάριο ο δέκτης θέτει το EWLN bit σε κατάλληλη τιμή και το στέλνει πίσω στην πηγή. Στο δεύτερο σενάριο το σχήμα snooper χρησιμοποιεί το EWLN στο σταθμό βάσης. Ο παράγοντας snooper στον σταθμό βάσης παρακολουθεί τα τμήματα TCP για τυχόν τρύπες στον χώρο ακολουθίας και τις

καταγράφει. Όταν τα πακέτα ACK έρχονται από τον δέκτη, ο παράγοντας snoot συμβουλευεται την λίστα που έχει δημιουργήσει, θέτει ανάλογα το EWLN bit στο πακέτο ACK και κάνει τις απαραίτητες τροποποιήσεις στην λίστα του. Υπάρχουν όμως κάποια προβλήματα και με τα παραπάνω σενάρια, όπως το ότι υπάρχει ανομοιογένεια ανάμεσα στα 2 σενάρια, το ότι ο παράγοντας snoot χρειάζεται υψηλή υπολογιστική ισχύ, το ότι και τα 2 σενάρια δεν λαμβάνουν υπόψη την αλληλεπίδραση με τα σχήματα ανάκτησης σφαλμάτων επιπέδου σύνδεσης και ότι η προσέγγιση με τον παράγοντα δεν ταιριάζει στην πολυεπίπεδη δομή των πρωτοκόλλων του διαδικτύου. Κάποιες άλλες τεχνικές είναι η Explicit Congestion Notification(ECN), η οποία κάνει χρήση τεχνικών ενεργής διαχείρισης ουράς (active queue management), όπως η RED(Random Early Detection), για να αποφύγει περιττά packet drops και η Selective Acknowledgement(SACK) η οποία εισάγει την δυνατότητα στο TCP-Reno να διαχειριστεί πολλαπλές απώλειες σε ένα παράθυρο μετάδοσης, όμως η καθεμία έχει τα δικά της προβλήματα. Παρά την δυνατότητα του EWLN να ενημερώσει τον χρήστη ότι το πακέτο δεν χάθηκε λόγω συμφόρησης, δεν υπάρχει μέχρι στιγμής τρόπος ο TCP δέκτης αν μάθει τον λόγο που χάθηκε το πακέτο. Η λύση που προτείνει το άρθρο είναι η υλοποίηση του EWLN με πληροφορίες του επίπεδο MAC. Αρχικά ρυθμίζεται το EWLN bit μέσα στο πρωτόκολλα MAC. Σε ασύρματα LAN MAC πρωτόκολλα, όπως το IEEE802.11.b, χρησιμοποιείται η τεχνική Cyclic Redundancy Check(CRC) για να εξακριβωθεί αν τα πακέτα παραλήφθηκαν σωστά. Πρώτα ελέγχεται αν τελείωσε η τοπική αναμετάδοση μέσω μιας τεχνικής buffering. Αν ισχύει, το EWLN bit θέτεται κατάλληλα ανάλογα με την πληροφορία από το CRC και τυχόν λανθασμένα πακέτα που κανονικά θα απορριπτόταν από το CRC δίνονται είτε στο επίπεδο μεταφοράς σε κινητό τερματικό είτε δρομολογούνται στον επόμενο κόμβο σε σταθμό βάσης. Έπειτα ρυθμίζεται ο TCP δέκτης. Όταν ο δέκτης λαμβάνει το πακέτο με EWLN bit, το ελέγχει και στην περίπτωση που είναι ορισμένο, στέλνεται ένα ACK πακέτο στο αποστολέα το οποίο περιέχει το EWLN bit και τον αριθμό αλληλουχίας του αλλοιωμένου πακέτου. Αυτή η διαδικασία συνεχίζεται μέχρι να τελειώσει η αναμετάδοση ή μέχρι να έρθει καινούργιο αλλοιωμένο πακέτο. Για να μπορέσει να επιτευχθεί αυτή η διαδικασία ο TCP δέκτης πρέπει να κάνει κάποιους ελέγχους στα πακέτα τα οποία έρχονται και να θέσει ανάλογα με τους ελέγχους το EWLN bit στο ACK πακέτο, ώστε ο αποστολέας να καταλάβει ότι το υπήρχε σφάλμα bit. Τέλος πρέπει να ρυθμιστεί ο αποστολέας. Αφού το ACK πακέτο με το EWLN bit έχει επιστρέψει στο αποστολέα, ο

αποστολέας μπορεί να ξαναστείλει το πακέτο χωρίς να εκτελέσει την διαδικασία συμφόρησης αν δεν χρειάζεται. Για να εξακριβωθεί αν αυτή η τεχνική μπορεί να υλοποιηθεί σε αρκετά κοντινές στην πραγματικότητα καταστάσεις, εφαρμόστηκαν προσομοιώσεις με τον ns-2 προσομοιωτή δικτύου του Berkeley με κάποιες επεκτάσεις. Οι αξιολογήσεις έγιναν σε ρυθμίσεις προσομοίωσης, όπου οι απώλειες οφείλονται μόνο σε ασύρματα σφάλματα bit και όπου οι απώλειες οφείλονται και σε ασύρματα σφάλματα bit και σε συμφόρηση δικτύου, στις οποίες συγκρίνονται μέσα από 100 δοκιμές τα TCP-Reno με Reno-EWLN και TCP-SACK με SACK-EWLN. Στην πρώτη περίπτωση παρατηρήθηκε ότι τα EWLN βελτιώνουν αρκετά την απόδοση στις περιοχές SNR [8.5dB,9.5dB], με μέγιστη την τιμή 9dB. Στην δεύτερη περίπτωση παρατηρήθηκε ότι τα EWLN βελτιώνουν αρκετά την απόδοση στις περιοχές SNR [8.5dB,10dB], με μέγιστη την τιμή 9dB. Η μέθοδος που παρουσιάζεται στο άρθρο προσπαθεί να δώσει επίλυση στα προβλήματα τα οποία αναφέρθηκαν παραπάνω, βέβαια όπως αναφέρεται και στο άρθρο "TCP Issues in Mobile Ad Hoc Networks: CHallenges and Solutions" από τους Wei-Qiang Xu και Tie-Jun Wu, στο κεφάλαιο 3.1, η μέθοδος αυτή μπορεί να χρησιμοποιηθεί μόνο για εύρεση σφάλματος και όχι σαν στρατηγική για ανάκτηση. Αυτό σημαίνει πως για να εφαρμοστεί σε πραγματικά δίκτυα θα πρέπει να γίνει σε συνδυασμό με άλλες τεχνικές και μεθόδους. Άρα είναι εύκολο να γίνει κατανοητό πως ενώ τα αποτελέσματα της προσομοίωσης μπορεί να δείχνουν αρκετά ικανοποιητικά, τα αποτελέσματα από μια υλοποίηση σε ένα δίκτυο μπορεί να διαφέρουν αρκετά. Βέβαια η αβεβαιότητα αυτή επιτρέπει την παραπάνω μελέτη και δίνει την δυνατότητα σε επόμενους ερευνητές ,όπως οι Wei-Qiang Xu και Tie-Jun Wu, να προβληματιστούν και να καινοτομήσουν πάνω στο αντικείμενο.

Ερώτηση 1

Αρχικά κάθε φορά που στέλνεται μια αίτηση HTTP στον server(είσοδος χρήστη/client στο site), θα πρέπει να ελέγχεται αν στην αίτηση HTTP υπάρχει κάποιο cookie ID. Στην περίπτωση που δεν υπάρχει θα πρέπει ο server να στείλει στον χρήστη μήνυμα απόκρισης HTTP, το οποίο θα περιέχει και ένα μοναδικό cookie ID. Αυτό το cookie ID θα αποθηκεύεται τόσο στον browser του client, όσο και στον server σε ένα database. Έτσι κάθε φορά που ο χρήστης θα κάνει μια αγορά, θα συσχετίζονται τα στοιχεία της αγοράς με το συγκεκριμένο ID και με αυτόν τον τρόπο θα μπορεί η εταιρία να καταγράφει τις αγορές των πελατών.

Ερώτηση 2

Η web cache είναι μια οντότητα δικτύου που ικανοποιεί αιτήσεις HTTP εκ μέρους ενός εξυπηρετητή αρχετύπου και έχει τον δικό της χώρο αποθήκευσης, στον οποίο αποθηκεύει αντίγραφα των αντικειμένων που ζητήθηκαν πρόσφατα σε αυτόν τον χώρο. Έτσι, αν ο χρήστης ζητήσει ένα αντικείμενο που έχει η cache θα μπορεί να του το παρέχει κατευθείαν χωρίς να χρειάζεται να επικοινωνήσει με τον server, παρακάμπτοντας κατά αυτόν τον τρόπο τις έξτρα καθυστερήσεις που θα επέφερε η επικοινωνία με τον server. Όπως ειπώθηκε παραπάνω, η web cache αποθηκεύει τα αντίγραφα των πρόσφατων αναζητήσεων. Αυτό σημαίνει πως κάποια από τα αντικείμενα που ζητούνται από έναν χρήστη να μην βρίσκονται μέσα στην cache και η ίδια να χρειαστεί να επικοινωνήσει με τον server για να πάρει τα αντικείμενα αυτά. Το αποτέλεσμα της επικοινωνίας με τον server είναι να μην μειωθεί η καθυστέρηση λήψης αυτών των αντικειμένων, καθώς δεν παρακάμπτονται η καθυστερήσεις που επιφέρει αυτή η επικοινωνία.

Ερώτηση 3

Από την εκφώνηση καταλαβαίνουμε πως ο υπολογιστής A χρησιμοποιεί την θύρα 'χ' και ο υπολογιστής B την θύρα 'γ'. Άρα τα τμήματα που ταξιδεύουν από τον υπολογιστή B στον A, θα έχουν αριθμό θύρας προέλευσης 'γ' και αριθμό θύρας προορισμού 'χ'.

Ερώτηση 4

Ο λόγος που ένας προγραμματιστής θα επιλέξει να χρησιμοποιήσει το πρωτόκολλο UDP, έναντι του TCP, είναι καθαρά για λόγους ταχύτητας στην μεταφορά δεδομένων.

Το UDP, σε αντίθεση με το TCP, είναι ασυνδεδεστικό, δηλαδή δεν απαιτεί εγκαθίδρυση σύνδεσης(handshaking) μεταξύ του αποστολέα και του παραλήπτη. Επίσης δεν υποστηρίζει αναμετάδοση χαμένων πακέτων, επιτρέπει εκτός από Broadcasting και multicasting και έχει μικρότερο μήκος πακέτων και επικεφαλίδας. Όλα τα παραπάνω και μερικά ακόμα κάνουν το UDP να είναι ταχύτερο του TCP. Αυτό έχει ως αποτέλεσμα ένας προγραμματιστής να το επιλέξει για εφαρμογές που μας χρειάζεται αρκετά γρήγορη μετάδοση δεδομένων, όπως είναι το online gaming, το VoIP, και το DNS.

Ερώτηση 5

Το TCP πρωτόκολλο υπερτερεί του UDP σε ότι αφορά την ασφαλή μετάδοση δεδομένων. Αρχικά, στο TCP πρωτόκολλο, απαιτείται εγκαθίδρυση TCP σύνδεσης μεταξύ διεργασιών client και server, προσφέρεται αξιόπιστη μεταφορά δεδομένων μεταξύ διεργασιών αποστολής και παραλαβής. Παράλληλα, ο αποστολέας δεν υπερφορτώνει τον παραλήπτη σε συνδυασμό με το ότι γίνεται έλεγχος συμφόρησης, στοιχείο το οποίο υποδηλώνει ότι γίνεται επιβράδυνση ρυθμού αποστολής όταν το δίκτυο είναι υπερφορτωμένο. Τέλος, παρέχεται ελάχιστο bandwidth, το οποίο εγγυάται καθυστέρηση. Από την άλλη πλευρά, για την χρήση του UDP πρωτοκόλλου δεν απαιτείται handshaking πριν από την έναρξη μεταφοράς δεδομένων. Αυτό όμως το καθιστά αναξιόπιστο σε συνδυασμό με το γεγονός ότι δεν διαθέτει τα παραπάνω. Οπότε προτιμότερο για την μεταφορά δεδομένων τύπου φωνής και βίντεο είναι το TCP, καθώς παρέχεται καλύτερη αξιοπιστία και ασφάλεια των δεδομένων.

Internet applications, and transport protocols

application	application layer protocol	transport protocol
file transfer/download	FTP [RFC 959]	TCP
e-mail	SMTP [RFC 5321]	TCP
Web documents	HTTP 1.1 [RFC 7320]	TCP
Internet telephony	SIP [RFC 3261], RTP [RFC 3550], or proprietary	TCP or UDP
streaming audio/video	HTTP [RFC 7320], DASH	TCP
interactive games	WOW, FPS (proprietary)	UDP or TCP

Ερώτηση 6

Σε ότι αφορά το UDP πρωτόκολλο, γνωρίζουμε ότι δεν προσφέρεται αξιόπιστη μεταφορά δεδομένων, καθώς δεν πληρεί πολλά στοιχεία αξιοπιστίας όπως το TCP. Παρόλα αυτά, μπορεί να επιτευχθεί το αντίθετο σύμφωνα με τα παρακάτω. Αρχικά, βελτιώνοντας μηχανισμούς αναμετάδοσης για την επίτευξη σωστής μεταφοράς

πακέτων, ώστε να μην χάνεται κανένα πακέτο κατά την μετάδοση. Έπειτα, εύρεση σφαλμάτων κατά την μεταφορά αποφεύγοντας έτσι την ανάγκη για επαναποστολή των δεδομένων. Τέλος, θα μπορούσε να υπάρχει καλύτερη διαχείριση των πακέτων στοιχείο το οποίο θα βοηθήσει στην καλύτερη διανομή των πακέτων, αποφεύγοντας το overflow του Buffer.

Ερώτηση 7

Τα δύο τμήματα UDP από τους υπολογιστές A και B με τον αριθμό θύρας προορισμού 9876 θα κατευθυνθούν στην ίδια υποδοχή του υπολογιστή C. Η διαδικασία στον υπολογιστή C μπορεί να διαφοροποιήσει τα τμήματα από τους υπολογιστές A και B εξετάζοντας τις διευθύνσεις IP προέλευσης και τους αριθμούς των θυρών προέλευσης των ληφθέντων τμημάτων UDP. Η διαδικασία στον υπολογιστή C μπορεί να διατηρεί ένα αρχείο περιόδων επικοινωνίας ή καταστάσεων χρησιμοποιώντας το συνδυασμό source IP και source port για να χειρίζεται διαφορετικά τα μηνύματα με βάση τον αποστολέα. Με αυτόν τον τρόπο, παρόλο που όλα τα τμήματα κατευθύνονται στην ίδια υποδοχή UDP, η διαδικασία μπορεί να διακρίνει και να χειρίζεται κατάλληλα μηνύματα από διαφορετικές πηγές.

Ερώτηση 8

Επειδή στον υπολογιστή C εκτελείται εξυπηρετητής WEB, χρησιμοποιείται το TCP πρωτόκολλο μεταφοράς. Ένα TCP socket προσδιορίζεται από την διεύθυνση IP πηγής, τον αριθμό θύρας πηγής, την διεύθυνση IP δέκτη και τον αριθμό θύρας δέκτη, με όλες αυτές τις τιμές να καθορίζουν την αποπολύπλεξη του segment. Επειδή τα πακέτα

που έρχονται από τον A έχουν διαφορετικό IP από τα πακέτα που έρχονται από τον B, τότε θα χρησιμοποιηθεί και διαφορετικό socket. Επίσης και πακέτα που έρχονται από τον ίδια υπολογιστή(A ή B) αλλά με διαφορετικό αριθμό θύρας πηγής, θα έχουν διαφορετικό socket. Ο αριθμός θύρας προορισμού θα είναι και σε όλα τα segment το ίδιο (80), διότι εκεί εκτελείται ο εξυπηρετητής web. Αυτό δεν θα επηρεάσει καθόλου την αποπολύπλεξη, καθώς ο server θα είναι σε θέση να τα ξεχωρίσει λόγω των διαφορετικών τιμών στα τέσσερα πεδία που αναφέρθηκαν στην αρχή.

Ερώτηση 9

```
Windows IP Configuration

Host Name . . . . . : DESKTOP-QIQEMA9
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Description . . . . . : Realtek PCIe GBE Family Controller
Physical Address. . . . . : 10-7B-44-1A-16-D3
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv6 Address. . . . . : 2a02:2149:8b70:1e00:89:966c:a125:4fd5 (Preferred)
Temporary IPv6 Address. . . . . : 2a02:2149:8b70:1e00:38cd:eba2:853b:f8fb (Preferred)
Link-local IPv6 Address . . . . . : fe80::38db:a452:4bc:697b%6 (Preferred)
IPv4 Address. . . . . : 192.168.1.8 (Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Τετάρτη, 10 Ιουλίου 2024 1:59:23 μμ
Lease Expires . . . . . : Πέμπτη, 11 Ιουλίου 2024 1:59:31 μμ
Default Gateway . . . . . : fe80::1%6
                          192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DHCPv6 IAID . . . . . : 51411780
DHCPv6 Client DUID. . . . . : 00-01-00-01-21-CD-B4-6F-10-7B-44-1A-16-D3
DNS Servers . . . . . : fe80::1%6
                          192.168.1.1
                          fe80::1%6
NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Σύνδεση δικτύου Bluetooth:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Bluetooth Device (Personal Area Network)
Physical Address. . . . . : E8-48-B8-C8-20-00
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
```


Με βάση την παραπάνω εικόνα έχουμε:

- Διεύθυνση IP: 192.168.1.9
- Μάσκα υποδικτύου: 255.255.255.0
- Προεπιλεγμένος δρομολογητής: 192.168.1.1
- Διεύθυνση IP του DNS: 192.168.1.1

Ερώτηση 10

Γνωρίζουμε από την θεωρία πως οι επικεφαλίδες του IP είναι 20byte και το TCP είναι επίσης 20byte.

Οπότε έχουμε:

H = 40byte (Επικεφαλίδες)

D = 220byte (Δεδομένα)

T = 260byte (Συνολικό φορτίο)

Το ποσοστό του επιπλέον βάρους θα είναι:

$$\frac{H}{T} \cdot 100\% = \frac{40}{260} \cdot 100\% = 15.385\%$$

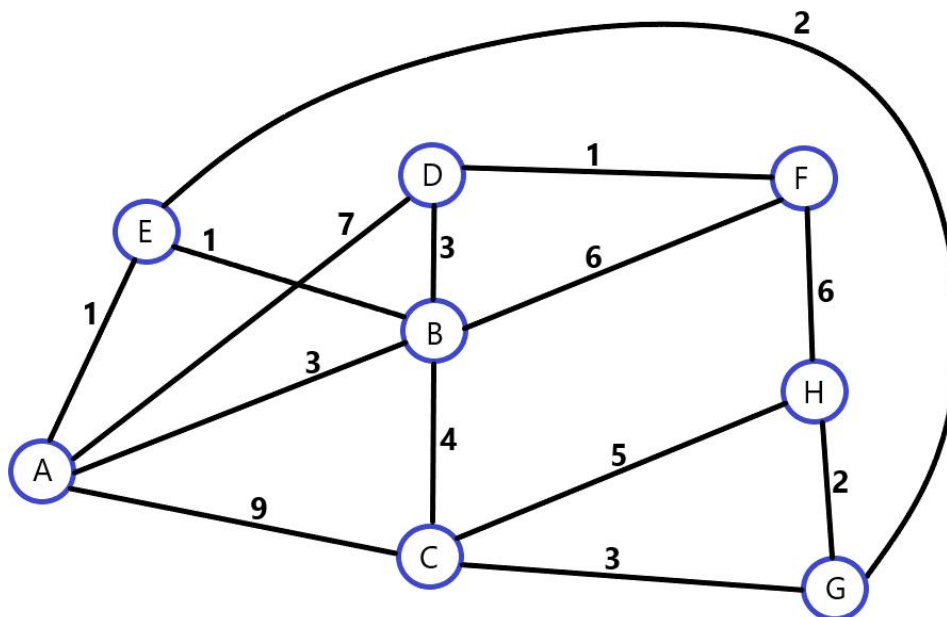
Το ποσοστό των δεδομένων εφαρμογής θα είναι:

$$\frac{D}{T} \cdot 100\% = \frac{220}{260} \cdot 100\% = 84.615\%$$

Άρα σε κάθε δεδομένογραμμα το 15.385% θα είναι το βάρος και το 84.615% θα είναι τα δεδομένα της εφαρμογής.

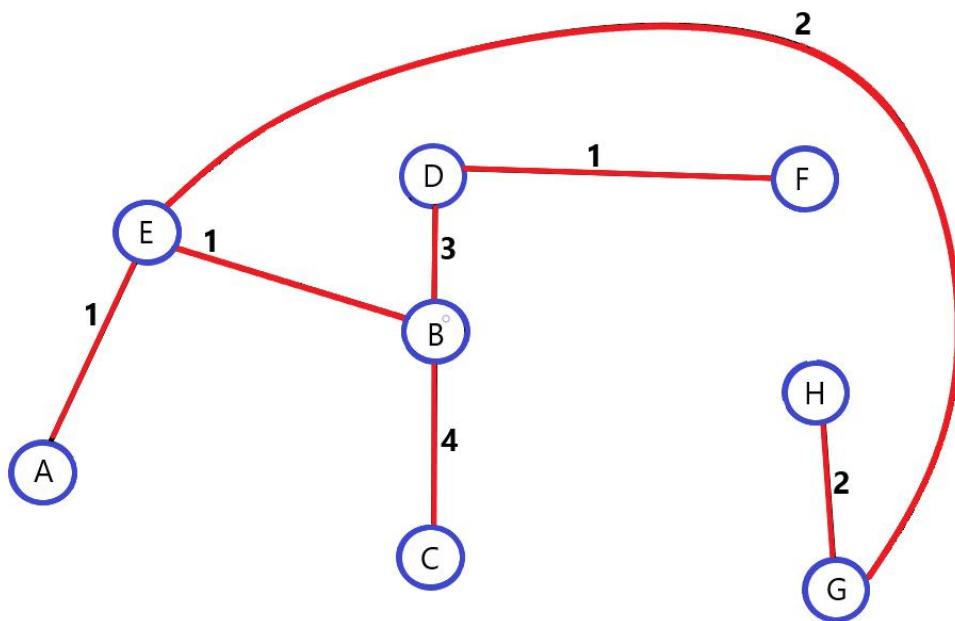
Άσκηση 1

1) Ο γράφος είναι ο εξής:



2) Ο πίνακας Κόστους ξεκινώντας από τον κόμβο A είναι ο εξής:

Επανάληψη	N'	D(B), ρ(B)	D(C), ρ(C)	D(D), ρ(D)	D(E), ρ(E)	D(F), ρ(F)	D(G), ρ(G)	D(H), ρ(H)
0	A	3,A	9,A	7,A	1,A	∞	∞	∞
1	AE	2,E	9,A	7,A		∞	3,E	∞
2	AEB		6,B	5,B		8,B	3,E	∞
3	AEBG		6,B	5,B		8,B		5,G
4	AEBGH		6,B	5,B		8,B		
5	AEBGHD		6,B			6,D		
6	AEBGHDF		6,B					
7	AEBGHDFC							



3) Η συντομότερη διαδρομή από τον router A σε κάθε άλλο router είναι η εξής:

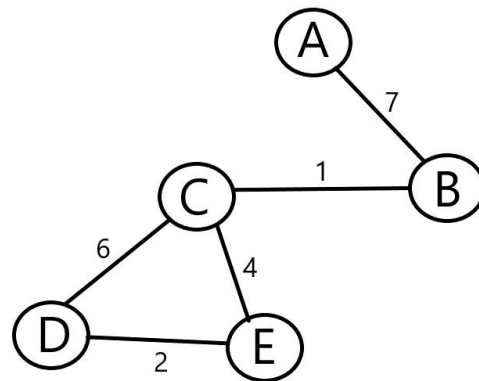
- AB: A → E → B, κόστος: 2
- AC: A → E → B → C, κόστος: 6
- AD: A → E → B → D, κόστος: 5
- AE: A → E, κόστος: 1
- AF: A → E → B → D → F, κόστος: 6
- AG: A → E → G, κόστος: 3
- AH: A → E → G → H, κόστος: 5

Άσκηση 2

Οι αρχικοί πίνακες δρομολόγησης($t=0$) για κάθε router είναι οι εξής:

DV in A:	DV in B:	DV in C:	DV in D:	DV in E:
$D_A(A)=0$	$D_B(A)=7$	$D_C(A)=\infty$	$D_D(A)=\infty$	$D_E(A)=\infty$
$D_A(B)=7$	$D_B(B)=0$	$D_C(B)=1$	$D_D(B)=\infty$	$D_E(B)=\infty$
$D_A(C)=\infty$	$D_B(C)=1$	$D_C(C)=0$	$D_D(C)=6$	$D_E(C)=4$
$D_A(D)=\infty$	$D_B(D)=\infty$	$D_C(D)=6$	$D_D(D)=0$	$D_E(D)=2$
$D_A(E)=\infty$	$D_B(E)=\infty$	$D_C(E)=4$	$D_D(E)=2$	$D_E(E)=0$

Δημιουργώντας τον γράφο μπορούμε να υπολογίσουμε πόσες επαναλήψεις χρειαζόμαστε.



Με βάση τον γράφο παρατηρούμε πως χρειαζόμαστε το μέγιστο 3 επαναλήψεις.

Επίσης με βάση την εκφώνηση μπορούμε να εξαγάγουμε τα ακόλουθα συμπεράσματα:

- Το A λαμβάνει πληροφορία μόνο από το B
- Το B λαμβάνει πληροφορία από το A και το C
- Το C λαμβάνει πληροφορία από το B, το D και το E
- Το D λαμβάνει πληροφορία από το C και το E
- Το E λαμβάνει πληροφορία από το D και το C

Για $t=1$

A:

- $D_A(B) = \min(c_{A,B} + D_B(B)) = 7+0 = 7$
- $D_A(C) = \min(c_{A,B} + D_B(C)) = 7+1 = 8$
- $D_A(D) = \min(c_{A,B} + D_B(D)) = 7+\infty = \infty$
- $D_A(E) = \min(c_{A,B} + D_B(E)) = 7+\infty = \infty$

Οπότε:

DV in A:
$D_A(A)=0$
$D_A(B)=7$
$D_A(C)=8$
$D_A(D)=\infty$
$D_A(E)=\infty$

B:

- $D_B(A) = \min(c_{B,A} + D_A(A), c_{B,C} + D_C(A)) = \min(7+0, 1+\infty) = 7$
- $D_B(C) = \min(c_{B,A} + D_A(C), c_{B,C} + D_C(C)) = \min(7+\infty, 1+0) = 1$
- $D_B(D) = \min(c_{B,A} + D_A(D), c_{B,C} + D_C(D)) = \min(7+\infty, 1+6) = 7$
- $D_B(E) = \min(c_{B,A} + D_A(E), c_{B,C} + D_C(E)) = \min(7+\infty, 1+4) = 5$

Οπότε:

DV in B:
$D_B(A)=7$
$D_B(B)=0$
$D_B(C)=1$
$D_B(D)=7$
$D_B(E)=5$

C:

- $D_C(A) = \min(c_{C,B} + D_B(A), c_{C,D} + D_D(A), c_{C,E} + D_E(A)) = \min(1+7, 6+\infty, 4+\infty) = 8$
- $D_C(B) = \min(c_{C,B} + D_B(B), c_{C,D} + D_D(B), c_{C,E} + D_E(B)) = \min(1+0, 6+\infty, 6+\infty) = 1$
- $D_C(D) = \min(c_{C,B} + D_B(D), c_{C,D} + D_D(D), c_{C,E} + D_E(D)) = \min(1+\infty, 6+0, 4+2) = 6$
- $D_C(E) = \min(c_{C,B} + D_B(E), c_{C,D} + D_D(E), c_{C,E} + D_E(E)) = \min(1+\infty, 6+2, 4+0) = 4$

Οπότε:

DV in C:
$D_C(A)=8$
$D_C(B)=1$
$D_C(C)=0$
$D_C(D)=6$
$D_C(E)=4$

D:

- $D_D(A) = \min(c_{D,C} + D_C(A), c_{D,E} + D_E(A)) = \min(6+\infty, 2+\infty) = \infty$
- $D_D(B) = \min(c_{D,C} + D_C(B), c_{D,E} + D_E(B)) = \min(6+1, 2+\infty) = 7$
- $D_D(C) = \min(c_{D,C} + D_C(C), c_{D,E} + D_E(C)) = \min(6+0, 2+4) = 6$
- $D_D(E) = \min(c_{D,C} + D_C(E), c_{D,E} + D_E(E)) = \min(6+4, 2+0) = 2$

Οπότε:

DV in D:
$D_D(A)=\infty$
$D_D(B)=7$
$D_D(C)=6$
$D_D(D)=0$
$D_D(E)=2$

E:

- $D_E(A) = \min(c_{E,C} + D_C(A), c_{E,D} + D_D(A)) = \min(4+\infty, 2+\infty) = \infty$
- $D_E(B) = \min(c_{E,C} + D_C(B), c_{E,D} + D_D(B)) = \min(4+1, 2+\infty) = 5$
- $D_E(C) = \min(c_{E,C} + D_C(C), c_{E,D} + D_D(C)) = \min(4+0, 2+4) = 4$
- $D_E(D) = \min(c_{E,C} + D_C(D), c_{E,D} + D_D(D)) = \min(4+6, 2+0) = 2$

Οπότε:

DV in E:
$D_E(A)=\infty$
$D_E(B)=5$
$D_E(C)=4$
$D_E(D)=2$
$D_E(E)=0$

Για $t=2$

A:

- $D_A(B) = \min(c_{A,B} + D_B(B)) = 7+0 = 7$
- $D_A(C) = \min(c_{A,B} + D_B(C)) = 7+1 = 8$
- $D_A(D) = \min(c_{A,B} + D_B(D)) = 7+7 = 14$
- $D_A(E) = \min(c_{A,B} + D_B(E)) = 7+5 = 12$

Οπότε:

DV in A:
$D_A(A)=0$
$D_A(B)=7$
$D_A(C)=8$
$D_A(D)=14$
$D_A(E)=12$

B:

- $D_B(A) = \min(c_{B,A} + D_A(A), c_{B,C} + D_C(A)) = \min(7+0, 1+8) = 7$
- $D_B(C) = \min(c_{B,A} + D_A(C), c_{B,C} + D_C(C)) = \min(7+8, 1+0) = 1$

- $D_B(D) = \min(c_{B,A} + D_A(D), c_{B,C} + D_C(D)) = \min(7+\infty, 1+6) = 7$
- $D_B(E) = \min(c_{B,A} + D_A(E), c_{B,C} + D_C(E)) = \min(7+\infty, 1+4) = 5$

Οπότε:

DV in B:
$D_B(A)=7$
$D_B(B)=0$
$D_B(C)=1$
$D_B(D)=7$
$D_B(E)=5$

C:

- $D_C(A) = \min(c_{C,B} + D_B(A), c_{C,D} + D_D(A), c_{C,E} + D_E(A)) = \min(1+7, 6+\infty, 4+\infty) = 8$
- $D_C(B) = \min(c_{C,B} + D_B(B), c_{C,D} + D_D(B), c_{C,E} + D_E(B)) = \min(1+0, 6+7, 6+5) = 1$
- $D_C(D) = \min(c_{C,B} + D_B(D), c_{C,D} + D_D(D), c_{C,E} + D_E(D)) = \min(1+7, 6+0, 4+2) = 6$
- $D_C(E) = \min(c_{C,B} + D_B(E), c_{C,D} + D_D(E), c_{C,E} + D_E(E)) = \min(1+5, 6+2, 4+0) = 4$

Οπότε:

DV in C:
$D_C(A)=8$
$D_C(B)=1$
$D_C(C)=0$
$D_C(D)=6$
$D_C(E)=4$

D:

- $D_D(A) = \min(c_{D,C} + D_C(A), c_{D,E} + D_E(A)) = \min(6+8, 2+\infty) = 14$
- $D_D(B) = \min(c_{D,C} + D_C(B), c_{D,E} + D_E(B)) = \min(6+1, 2+5) = 7$
- $D_D(C) = \min(c_{D,C} + D_C(C), c_{D,E} + D_E(C)) = \min(6+0, 2+4) = 6$
- $D_D(E) = \min(c_{D,C} + D_C(E), c_{D,E} + D_E(E)) = \min(6+4, 2+0) = 2$

Οπότε:

DV in D:
$D_D(A)=14$
$D_D(B)=7$
$D_D(C)=6$
$D_D(D)=0$
$D_D(E)=2$

E:

- $D_E(A) = \min(c_{E,C} + D_C(A), c_{E,D} + D_D(A)) = \min(4+8, 2+\infty) = 12$
- $D_E(B) = \min(c_{E,C} + D_C(B), c_{E,D} + D_D(B)) = \min(4+1, 2+7) = 5$
- $D_E(C) = \min(c_{E,C} + D_C(C), c_{E,D} + D_D(C)) = \min(4+0, 2+4) = 4$
- $D_E(D) = \min(c_{E,C} + D_C(D), c_{E,D} + D_D(D)) = \min(4+6, 2+0) = 2$

Οπότε:

DV in E:
$D_E(A)=12$
$D_E(B)=5$
$D_E(C)=4$
$D_E(D)=2$
$D_E(E)=0$

Για t=3

A:

- $D_A(B) = \min(c_{A,B} + D_B(B)) = 7+0 = 7$
- $D_A(C) = \min(c_{A,B} + D_B(C)) = 7+1 = 8$
- $D_A(D) = \min(c_{A,B} + D_B(D)) = 7+7 = 14$
- $D_A(E) = \min(c_{A,B} + D_B(E)) = 7+5 = 12$

Δεν υπάρχει ενημέρωση του πίνακα

B:

- $D_B(A) = \min(c_{B,A} + D_A(A), c_{B,C} + D_C(A)) = \min(7+0, 1+8) = 7$
- $D_B(C) = \min(c_{B,A} + D_A(C), c_{B,C} + D_C(C)) = \min(7+8, 1+0) = 1$
- $D_B(D) = \min(c_{B,A} + D_A(D), c_{B,C} + D_C(D)) = \min(7+14, 1+6) = 7$
- $D_B(E) = \min(c_{B,A} + D_A(E), c_{B,C} + D_C(E)) = \min(7+12, 1+4) = 5$

Δεν υπάρχει ενημέρωση του πίνακα

C:

- $D_C(A) = \min(c_{C,B} + D_B(A), c_{C,D} + D_D(A), c_{C,E} + D_E(A)) = \min(1+7, 6+14, 4+12) = 7$
- $D_C(B) = \min(c_{C,B} + D_B(B), c_{C,D} + D_D(B), c_{C,E} + D_E(B)) = \min(1+0, 6+7, 6+5) = 1$
- $D_C(D) = \min(c_{C,B} + D_B(D), c_{C,D} + D_D(D), c_{C,E} + D_E(D)) = \min(1+7, 6+0, 4+2) = 6$
- $D_C(E) = \min(c_{C,B} + D_B(E), c_{C,D} + D_D(E), c_{C,E} + D_E(E)) = \min(1+5, 6+2, 4+0) = 4$

Δεν υπάρχει ενημέρωση του πίνακα

D:

- $D_D(A) = \min(c_{D,C} + D_C(A), c_{D,E} + D_E(A)) = \min(6+8, 2+12) = 14$
- $D_D(B) = \min(c_{D,C} + D_C(B), c_{D,E} + D_E(B)) = \min(6+1, 2+5) = 7$
- $D_D(C) = \min(c_{D,C} + D_C(C), c_{D,E} + D_E(C)) = \min(6+0, 2+4) = 6$
- $D_D(E) = \min(c_{D,C} + D_C(E), c_{D,E} + D_E(E)) = \min(6+4, 2+0) = 2$

Δεν υπάρχει ενημέρωση του πίνακα

E:

- $D_E(A) = \min(c_{E,C} + D_C(A), c_{E,D} + D_D(A)) = \min(4+8, 2+14) = 12$
- $D_E(B) = \min(c_{E,C} + D_C(B), c_{E,D} + D_D(B)) = \min(4+1, 2+7) = 5$
- $D_E(C) = \min(c_{E,C} + D_C(C), c_{E,D} + D_D(C)) = \min(4+0, 2+4) = 4$
- $D_E(D) = \min(c_{E,C} + D_C(D), c_{E,D} + D_D(D)) = \min(4+6, 2+0) = 2$

Δεν υπάρχει ενημέρωση του πίνακα

Άρα οι τελικοί πίνακες είναι οι εξής:

DV in A:	DV in B:	DV in C:	DV in D:	DV in E:
$D_A(A)=0$	$D_B(A)=7$	$D_C(A)=8$	$D_D(A)=14$	$D_E(A)=12$
$D_A(B)=7$	$D_B(B)=0$	$D_C(B)=1$	$D_D(B)=7$	$D_E(B)=5$
$D_A(C)=8$	$D_B(C)=1$	$D_C(C)=0$	$D_D(C)=6$	$D_E(C)=4$
$D_A(D)=14$	$D_B(D)=7$	$D_C(D)=6$	$D_D(D)=0$	$D_E(D)=2$
$D_A(E)=12$	$D_B(E)=5$	$D_C(E)=4$	$D_D(E)=2$	$D_E(E)=0$

Ο τελικός πίνακας είναι ο εξής:

Από/Προς	A	B	C	D	E
A	0	7	8	14	12
B	7	0	1	7	5
C	8	1	0	6	4
D	14	7	6	0	2
E	12	5	4	2	0

Η συντομότερη διαδρομή από τον router A σε κάθε άλλο router, είναι η εξής:

Προορισμός	Κόστος	Διαδρομή
A	0	A
B	7	A-B
C	8	A-B-C
D	14	A-B-C-D
E	12	A-B-C-E

Άσκηση 3

Με μάσκα υποδικτύου /21 έχουμε:

- 11111111.11111111.11111000.00000000 (Μάσκα υποδικτύου σε δυαδικό)
- $n = 32 - 21 = 11$

- Πλήθος διευθύνσεων: $2^n = 2^{11} = 2048$

192.168.00010000.00000000 (Σε κόκκινο είναι τα bit τα οποία θα καθορίσουν το εύρος των διευθύνσεων υποδικτύου).

Άρα οι διευθύνσεις είναι από 192.168.00010000.00000000 έως 192.168.00010111.11111111, ή σε δεκαδική αναπαράσταση 192.168.16.0 έως 192.168.23.255 .

Επίσης για κάθε υποδίκτυο πρέπει να λάβουμε υπόψη την διεύθυνση υποδικτύου και την διεύθυνση εκπομπής(broadcast). Άρα για κάθε υποδίκτυο θεωρούμε πως χρειαζόμαστε $x + 2$ διευθύνσεις, όπου “x” ο αριθμός των απαιτούμενων διεπαφών.

Υποδύκτιο 1:

Γνωρίζουμε ότι $2^6 = 64$ και $2^5 = 32$. Για να μπορέσει το υποδίκτυο να υποστηρίξει τουλάχιστον 52 διευθύνσεις χρειαζόμαστε 6bit.

Υποδύκτιο 2:

Γνωρίζουμε ότι $2^7 = 128$ και $2^6 = 64$. Για να μπορέσει το υποδύκτιο να υποστηρίξει τουλάχιστον 82 διευθύνσεις χρειαζόμαστε 7bit.

Υποδύκτιο 3:

Γνωρίζουμε ότι $2^6 = 64$ και $2^5 = 32$. Για να μπορέσει το υποδύκτιο να υποστηρίξει τουλάχιστον(ίσως να χρειαστούν και άλλες) 32 διευθύνσεις χρειαζόμαστε 6bit.

Υποδύκτιο 4:

Γνωρίζουμε ότι $2^5 = 32$ και $2^4 = 16$. Για να μπορέσει το υποδύκτιο να υποστηρίξει τουλάχιστον 22 διευθύνσεις χρειαζόμαστε 5bit.

Οι συνολικές διευθύνσεις που θα χρειαστούν είναι:

$T = 64 + 128 + 64 + 32 \Leftrightarrow T = 288$ διευθύνσεις

Γνωρίζουμε ότι $2^9 = 512$ και $2^8 = 256$. Για τουλάχιστον 288 διευθύνσεις θα χρησιμοποιήσουμε 9bit και άρα θα χρησιμοποιήσουμε μάσκα /23, διότι $32 - x = 9 \Leftrightarrow x = 23$.

Άρα οι διευθύνσεις είναι από 192.168.00010000.00000000 έως 192.168.00010001.11111111, ή σε δεκαδική αναπαράσταση 192.168.16.0 έως 192.168.17.255.

Υποδύκτιο 1:

Οι διευθύνσεις θα είναι 192.168.16.0 - 192.168.16.63, επειδή όμως η πρώτη είναι η διεύθυνση υποδικτύου και η τελευταία η διεύθυνση εκπομπής (broadcast), οι διευθύνσεις που μπορούν να χρησιμοποιηθούν είναι από 192.168.16.1 έως 192.168.16.62.
Άρα 192.168.16.0/23.

Υποδύκτιο 2:

Οι διευθύνσεις θα είναι 192.168.16.64 - 192.168.16.191 επειδή η πρώτη είναι η διεύθυνση υποδυκτίου και η τελευταία η διεύθυνση εκπομπής (broadcast), οι διευθύνσεις που μπορούν να χρησιμοποιηθούν είναι από 192.168.16.65 έως 192.168.16.190.
Άρα 192.168.16.64/23.

Υποδύκτιο 3:

Οι διευθύνσεις θα είναι 192.168.16.192 - 192.168.16.255 επειδή η πρώτη είναι η διεύθυνση υποδυκτίου και η τελευταία η διεύθυνση εκπομπής (broadcast), οι διευθύνσεις που μπορούν να χρησιμοποιηθούν είναι από 192.168.16.193 έως 192.168.16.254.
Άρα 192.168.16.192/23.

Υποδύκτιο 4:

Οι διευθύνσεις θα είναι 192.168.17.0 - 192.168.17.21 επειδή η πρώτη είναι η διεύθυνση υποδυκτίου και η τελευταία η διεύθυνση εκπομπής (broadcast), οι διευθύνσεις που μπορούν να χρησιμοποιηθούν είναι από 192.168.17.1 έως 192.168.17.20.
Άρα 192.168.17.0/23.