

Ερώτηση 1

Ποια είναι η διαφορά μεταξύ των λειτουργιών της προώθησης και της δρομολόγησης του επιπέδου δικτύου;

1) Με τον όρο προώθηση εννοούμε την τοπική μεταφορά (απο τον δρομολογητή) ενός datagram απο μία διεπαφή εισερχόμενης ζεύξης προς μία διεπαφή εξερχόμενης ζεύξης . Με τον όρο δρομολόγηση εννοούμε την συνολική εργασία ολόκληρου του δικτύου να καθορίσει τις point-to-point routes ,

σύμφωνα με τις οποίες τα datagrams πηγαίνουν απ την μία άκρη του δικτύου στη άλλη. Οι αλγόριθμοι που τα καθορίζουν αυτα είναι οι routing algorithms.

Παραδείγματος χάρη, έστω ότι κάποιος θέλει να ταξιδέψει οδικώς από το Ηράκλειο στη Κνωσό .

Η προώθηση μοιάζει με τα πιθανά σταυροδρόμια που θα συναντήσει στη διάρκεια της διαδρομής-καθορίζει ποιον δρόμο θα ακολουθήσει μετέπειτα , ενώ η δρομολόγηση είναι ουσιαστικά ο σχεδιασμός του ταξιδιού,δηλαδή όλες οι πιθανές διαδρομές και σε πόση ώρα θα φτάσει εκεί

Ερώτηση 2

Συγκρίνετε τους αλγορίθμους δρομολόγησης κατάστασης ζεύξης (link-state) και απόστασης διανύσματος (distance-vector)

Οι αλγόριθμοι distance vector και link-state έχουν πολύ παρόμοιους τρόπους δρομολόγησης.Στον link state ο κάθε κόμβος επικοινωνεί με τους υπόλοιπους αλλά δίνει μόνο τα απευθείας κόστη ζευξης και γνωρίζει την τοπολογία του δικτύου ενώ στον distance vector ο κόμβος μπορεί να επικοινωνήσει μόνο με 1 άλλο κατευθείαν συνδεδεμένο σε αυτον κόμβο, παρ'ολα αυτά προσφέρει σε όλους τους γειτονικούς κόμβους του την εκτίμηση κόστους του.

Κάποιες αξιόλογες διαφορές είναι οι εξής:

Ταχύτητα:ο ls αλγόριθμος χρειάζεται (αριθμός κόμβων*πλήθος ζευξεων) μηνύματα να στείλει μέχρι να ολοκληρωθεί. Αντίθετα,ο dv συγκλίνει με πιο αργό ρυθμό και ίσως περιέχει βρόχους.Επίσης μπορεί να κάνει count to infinity (μέτρημα μέχρι το άπειρο)

Στιβαρότητα: ο κάθε κόμβος που τρέχει ls υπολογίζει μόνο τους δικούς του routing tables ,οπότε οι υπολογισμοί είναι ξεχωριστοί και άρα έχουμε μια μορφή στιβαρότητας . Στους dv algorithms ο κόμβος έχει τη δυνατότητα να διαφημίσει λανθάνοντες διαδρομές προς όλους τους κόμβους του δικτύου ή μερίδα αυτών .Σαν αποτέλεσμα αυτού, ίσως

διαδοθεί λάθος πληροφορία αναμεταξύ των κόμβων.

Πολυπλοκότητα μηνύματος:Είδαμε ότι ο ls αλγόριθμος θέλει(αριθμος κομβων *αριθμος ζευξεων) μηνύματα ώστε κάθε κόμβος να ξέρει το κόστος ζεύξης με κάθε άλλο κόμβο. Ακόμα, όταν αλλάζει το κόστος κάποιας ζεύξης αυτό πρέπει να σταλεί σε όλους τους κόμβους, ενώ αντίθετα ο dn θα διαδώσει τα αποτελέσματα του αλλαγμένου κόστους ζεύξης μόνο αν το νέο αλλάζει την διαδρομή ελάχιστου κόστους για ένα κόμβο που είναι σε αυτή τη ζεύξη.

Οι dn algorithms ζητούν ανταλλαγή μηνυμάτων ανάμεσα σε γειτονικούς συνδεδεμένους κόμβους επαναληπτικά κάθε φορά, κάτι που επηρεάζει τη σύγκλιση του αλγορίθμου.

Ερώτηση 3

Εξηγήστε τι προβλήματα δρομολόγησης προκύπτουν λόγω του μεγάλου πλήθους συσκευών στο Διαδίκτυο που επικοινωνούν μεταξύ τους.

Όπως γνωρίζουμε στο διαδίκτυο εμφανίζοντα συχνά προβλήματα λόγω του μεγάλου πλήθους των συσκευών. Ένα από αυτά είναι το πρόβλημα της κινητικότητας το οποίο πηγάζει από το IP addressing. Για παράδειγμα αν ένας κόμβος με μοναδική IP ,η οποία καθορίζει και το δίκτυο που ανήκει,σταματήσει να δέχεται μηνύματα και δεν ενημερώσει τους υπόλοιπους κόμβους,αυτοί θα στέλνουν μηνύματα τα οποία δεν θα δρομολογούνται σωστά. Το ίδιο και αν για κάποιο λόγο αλλάξει subnet ο κόμβος,συνεπώς και IP address

Επίσης λόγω του μεγάλου πλήθους υπάρχει μεγάλο κόστος στην κατασκευή των routing tables,και συνεπώς στην εύρεση των shortest paths.

Ερώτηση 4

Γιατί η δρομολόγηση στο διαδίκτυο γίνεται με τη χρήση των IP διευθύνσεων και όχι των MAC διευθύνσεων;

Γνωρίζουμε για τις IP και τις MAC addresses τε εξής:

IP->μοναδική για κάθε υπολογιστή που ανήκει σε ένα συγκεκριμένο υποδίκτυο και μεταβλητή ανάλογα το υποδίκτυο. Η “δομή” μιας IP διεύθυνσης είναι ιεραρχική και δίνεται από το ίδιο το δίκτυο.

MAC->μοναδική για κάθε υπολογιστή,σε όποιο υποδίκτυο και αν ανήκει και δίνεται απο τον κατασκευαστή της συσκευής. Σε όποιο δίκτυο και αν ανήκει μια συσκευή έχει σταθερή mac address.

Συνεπώς από τα παραπάνω μπορούμε να συμπεράνουμε τα εξής.

Για να φτάσει ένα πακέτο σε κάποιο τερματικό θα πρέπει προφανώς να

δρομολογηθεί. Οι διευθύνσεις IP λόγω της ιεραρχίας δείχνουν το πως ακριβώς θα γίνει αυτή η δρομολόγηση. Πως δηλαδή από τον κοντινότερο γείτονα(που θα έχει και αυτός δική του μοναδική IP) θα φτάσουμε στο άλλο τερματικό και ποιο θα είναι το shortest path. Αν γνωρίζαμε μόνο την MAC address και όχι σε ποιο δίκτυο ανήκει η συγκεκριμένη συσκευή θα έπρεπε να ενημερωθούν ΌΛΟΙ οι routers για το που βρίσκεται. Η απάντηση λοιπόν της ερώτησης είναι ότι η δρομολόγηση γίνεται μέσω IP γιατί αυτές δίνονται στον καθένα ανάλογα με το δίκτυο στο οποίο ανήκει και γιατί αυτές χρησιμοποιούν αυτό το ιεραρχικό σύστημα για καλύτερη οργάνωση του Διαδικτύου και συνεπώς πιο ορθή δρομολόγηση.

Ερώτηση 5.

Έχουν IP διευθύνσεις οι δρομολογητές; Αν ναι, πόσες;

Ναι έχουν IP διευθύνσεις, τόσες όσες και τα δίκτυα στα οποία ανήκει κάθε ένας από αυτούς τους δρομολογητές. Κάθε μια από αυτές τις διευθύνσεις έχει διαφορετικό prefix για να μπορεί να προωθήσει ο router το πακέτο στη σωστή διεύθυνση την οποία βρίσκει μέσω του routing table. Αυτό υποδεικνύει στον δρομολογητή ποιος θα είναι ο επόμενος γείτονας router που θα πάρει το πακέτο, ανάλογα με την IP που διάβασε. Ένας δρομολογητής που συνδέει N δίκτυα, έχει N IP διευθύνσεις.

Ερώτηση 6.

Είναι αλήθεια ότι όλα τα αυτόνομα συστήματα χρησιμοποιούν τον ίδιο intra-AS αλγόριθμο δρομολόγησης;

Όχι, κάτι τέτοιο δεν ισχύει. Αυτό που ισχύει είναι ότι όλοι οι δρομολογητές στο εσωτερικό ενός AS τρέχουν τον ίδιο intra-AS αλγόριθμο δρομολόγησης. Όσον αφορά διαφορετικά AS τότε το καθένα μπορεί να τρέχει το δικό του intra-AS αλγόριθμο. Αυτό εξαρτάται κυρίως από τους administrators(π.χ ISP's), οι οποίοι διαλέγουν τον αλγόριθμο δρομολόγησης που θα χρησιμοποιηθεί. Δρομολογητές σε διαφορετικά AS μπορεί να τρέχουν διαφορετικά intra-AS πρωτόκολλα δρομολόγησης.

Ερώτηση 7

Περιγράψτε πως το BGP χρησιμοποιεί το NEXT-HOP χαρακτηριστικό και πως το AS-PATH;

NEXT-HOP: Κάθε router που τρέχει BGP πρωτόκολλο, αφού διαβάσει την IP address στην οποία απευθύνεται το πακέτο, το επόμενο βήμα του είναι να το προωθήσει στον αμέσως επόμενο δρομολογητή ώστε να φτάσει στον

προορισμό του.Αυτός ο “επόμενος δρομολογητής” είναι και το next hop. Στο BGP λοιπόν θέλουμε να υπάρχει μια σύνδεση ανάμεσα σε eBGP και iBGP

Σε αυτό βοηθάει η διεύθυνση IP που ονομάζεται next-hop.Όταν ένας router που τρέχει ebgp πρωτοκόλλο,θέλει να διαφημίσει μια διαδρομή(AS-PATH+next_hop το οποίο είναι η διεύθυνση διεπαφής αυτών των 2 routers) του σε κάποιον δρομολογητή που τρέχει επίσης ebgp πρωτόκολλο,ο δευτερος αυτός δρομολογητής θα φροντίσει να ενημερώσει τους routers του δικού του AS μέσω ibgp για αυτή τη διαδρομή έτσι ώστε να τροποποιήσουν κατάλληλα τα routing table τους.

AS-PATH:Όταν ένας δρομολογητής ,ανακοινώνει κάποιο πρόθεμα που έλαβε τότε μαζί με αυτό ανακοινώνει και την διαδρομή που ακολούθησε αυτή η ανακοίνωση.Δηλαδή αν ένα πρόθεμα έχει περάσει από τα AS 1,2 τότε αν ακούσει το AS 3 που είναι συνδεδεμένο με το 2 και το 4,το 4 θα λάβει αυτή την ανακοίνωση μαζί με την διαδρομή 1-> 2-> 3->4.Αυτός είναι ένας τρόπος να αποφευχθούν επαναλήψεις στις ανακοινώσεις αυτές. Πιο συγκεκριμένα αν ένας δρομολογητής του AS δει ότι στο ASPATH που έλαβε υπάρχει και το AS στο οποίο ανήκει τότε κατευθείαν θα απορρίψει την διαφήμιση αυτή,γιατί πολύ απλά μπορεί να την κάνει κατευθείαν ο ίδιος.Αν μια διαδρομή έχει ίδιο μήκος AS-PATH τότε επιλέγεται αυτή με το πλησιέστερο δρομολογητή next-hop.(χαμηλοτερο κοστος)

Ερώτηση 8.

IP DATAGRAM:3000bytes with header→ 2980 without it.

MTU:500 bytes→480 payload.

FRAGMENTS→ $2980/480=6.2$ άρα 7 τεμάχια.

1ο πακέτο:0-479	(480) bytes	Offset=0
2ο πακέτο:480-959	(480)	Offset=480
3ο πακέτο:960-1339	(480)	Offset=960
4ο πακέτο:1340-1919	(480)	Offset=1340
5ο πακέτο:1920-2399	(480)	Offset=1920
6ο πακέτο:2400-2879	(480)	Offset=2400
7ο πακέτο:2880-2980	(100)bytes	Offset=2880

Ερώτηση 9

Αν θέλουμε να έχουμε ένα υποδίκτυο (subnet) το οποίο θα επιτρέπει την ύπαρξη τουλάχιστον 8000 διευθύνσεων για κόμβους, ποια μάσκα διεύθυνσης (subnet mask) θα ήταν η καλύτερη επιλογή;

Η καλύτερη επιλογή θα ήταν μια subnet mask που θα είχε ελεύθερα τα 13 τελευταία bits για διευθύνσεις κόμβων. Άρα η εξής:
11111111.11111111.11100000.00000000

ΕΡΩΤΗΣΗ 10

a) $D_x(w) = \min \{c(x,w) + D_w(w), c(x,y) + D_y(w)\} = \min \{2+0, 5+2\} = 2$

$D_x(y) = \min \{c(x,y) + D_y(y), c(x,w) + D_w(y)\} = \min \{5+0, 2+2\} = 4$

$D_x(w) = \min \{c(u,w) + D_x(w), c(x,u) + D_u(w)\} = \min \{5+2, 6+4\} = 7.$

b) i) Αν το $c(x,w)$ πάρει τιμή μεγαλύτερη του 6 τότε η διαδρομή για να φτάσει στον κόμβο u θα είναι μέσω του y άρα θα ενημερώσει τους γείτονες του x και θα ξανατρέξει τον αλγόριθμο distance vector.

ii) Για να μην ενημερώσει τους γείτονες για κάποιο νέο μονοπάτι τότε χρειάζεται απλά μια αλλαγή στο $c(x,y)$, είτε είναι μεγαλύτερο του 5 ή μικρότερο του 5 (και μεγαλύτερο από 1) δεν θα χρειαστεί να ενημερώσει για shortest path προς τον κόμβο u .

Ερώτηση 11.

a)

Χρησιμοποιούμε iBGP όταν έχουμε πολλούς δρομολογητές οι οποίοι χρησιμοποιούν το BGP για να ανταλλάσσουν πληροφορίες δρομολόγησης με τους ISPs τους. (διαφορετικά AS)

Με μια σύνδεση από iBGP γείτονες, στην περίπτωση των δρομολογητών άκρου, κάθε δίκτυο που αντιμετωπίζει πρόβλημα στο δρομολογητή μπορεί να μάθει διαδρομές από άλλο router και να αποφασίσει αν ο άλλος έχει μια καλύτερη διαδρομή για να φτάσουμε σε κάποιους προορισμούς στο Διαδίκτυο. Χωρίς το iBGP connection, οι δρομολογητές δεν έχουν κανέναν τρόπο να γνωρίζουν αν ο άλλος δρομολογητής έχει μια καλύτερη διαδρομή.

Υπό κανονικές συνθήκες, μία eBGP διαδρομή προτιμάται πάντα σε σχέση με μια διαδρομή iBGP, καθώς μία διαδρομή που μαθεύτηκε από έναν iBGP δεν θα διαφημιστεί σε μια άλλη iBGP by default. Ακόμη, οι iBGP γείτονες δεν χρειάζεται να συνδεθούν άμεσα και παράμετροι όπως η τοπική προτίμηση αποστέλλονται στους iBGP γείτονες αλλά όχι σε eBGP

γείτονες.

Η μόνη απαίτηση είναι η προσβασιμότητα προς τον απομακρυσμένο σταθμό με κατάλληλη διαμόρφωση eBGP-multihop.

Τέλος, το iBGP σχηματίζεται μεταξύ γειτόνων, στο ίδιο AS ενώ eBGP σχηματίζεται μεταξύ γειτόνων σε διαφορετικά AS.

b)

i.eBGP

ii. iBGP (αφού ο 3a ανήκει στο ίδιο AS με τον 3c που είναι border router)

iii.eBGP

iv. iBGP

ΑΣΚΗΣΗ 1

a)Step	Nprime	D(a),p(a)	D(c),p(c)	D(d),p(d)	D(b),p(b)	D(t),p(t)
0	u	18,u	15,u	20,u	inf	inf
1	uc	17,c		20,u	29,c	inf
2	uca				27,a	56,a
3	ucad					54,d
4	ucadb					53,b

b) i) $u \rightarrow c \rightarrow a \rightarrow b$

ii) $u \rightarrow d$

iii) $u \rightarrow c \rightarrow a \rightarrow b \rightarrow t$

c) Forwarding table u

Destination	Link
a	(u,c)
c	(u,c)
b	(u,c)
d	(u,d)
t	(u,c)

ΑΣΚΗΣΗ 2

a) 149.53.82.0/24

10010101.00110101.01010010.00000000

Αφού δεσμεύουμε τα 24 τότε ισχύει

10010101.00110101.01010010.00000000

Επίσης θέλουμε να δεσμεύσουμε 5 ακόμα bits για να έχουμε 20 subnets. Τα 12 subnets που θα περισσέψουν είναι για ενδεχόμενη μελλοντική χρήση. Άρα σύνολο έχουμε δεσμεύσει 29 bits.

$32-29=3$ bits ελεύθερα για host addresses. Άρα $2^3=8$ hosts σε κάθε δίκτυο. Βέβαια το 000 και 111 έχουν ήδη δεσμευθεί από το πρωτόκολλο, άρα 6 hosts.

Υποδίκτυο 0:	10010101.00110101.01010010.00000000	149.53.82.0
Υποδίκτυο 1:	10010101.00110101.01010010.00001000	149.53.82.8
Υποδίκτυο 2:	10010101.00110101.01010010.00010000	149.53.82.16
Υποδίκτυο 3:	10010101.00110101.01010010.00011000	149.53.82.24
Υποδίκτυο 4:	10010101.00110101.01010010.00100000	149.53.82.32
Υποδίκτυο 5:	10010101.00110101.01010010.00101000	149.53.82.40
Υποδίκτυο 6:	10010101.00110101.01010010.00110000	149.53.82.48
Υποδίκτυο 7:	10010101.00110101.01010010.00111000	149.53.82.56
Υποδίκτυο 8:	10010101.00110101.01010010.01000000	149.53.82.64
Υποδίκτυο 9:	10010101.00110101.01010010.01001000	149.53.82.72
Υποδίκτυο 10:	10010101.00110101.01010010.01010000	149.53.82.80
Υποδίκτυο 11:	10010101.00110101.01010010.01011000	149.53.82.88
Υποδίκτυο 12:	10010101.00110101.01010010.01100000	149.53.82.96
Υποδίκτυο 13:	10010101.00110101.01010010.01101000	149.53.82.104
Υποδίκτυο 14:	10010101.00110101.01010010.01110000	149.53.82.112
Υποδίκτυο 15:	10010101.00110101.01010010.01111000	149.53.82.120
Υποδίκτυο 16:	10010101.00110101.01010010.10000000	149.53.82.128
Υποδίκτυο 17:	10010101.00110101.01010010.10001000	149.53.82.136
Υποδίκτυο 18:	10010101.00110101.01010010.10010000	149.53.82.144
Υποδίκτυο 19:	10010101.00110101.01010010.10011000	149.53.82.152

ερώτημα β

i) 149.53.32.0

Σε δεκαδικό σύστημα θα γραφτεί ως εξής

10010101.00110101.00100000.00000000

Από την εκφώνηση είναι γνωστό ότι δεσμεύονται τα πρώτα 19 bits.

10010101.00110101.00010000.00000000

Άρα για να το χωρίσουμε σε 4 μικρότερα μπλοκ διευθύνσεων θα προσθέσουμε 2 bits στα 19 που έχουν ήδη δεσμευθεί. Σύνολο 21. Άρα $32-21=11$ bits

10010101.00110101.00100000.00000000

Φίλος 0: 10010101.00110101.00100000.00000000 149.53.32.0
 Φίλος 1: 10010101.00110101.001010000.00000000 149.53.40.0
 Φίλος 2: 10010101.00110101.001100000.00000000 149.53.48.0
 Φίλος 3: 10010101.00110101.001110000.00000000 149.53.56.0

ii) 21 bits μέχρι στιγμής για κάθε διεύθυνση. Αφού τώρα θέλουμε 24 bits
 24-21=3bits ακόμα, άρα $2^3=8$ άρα 8 subnets για τον καθένα.

ΑΣΚΗΣΗ 3.

	interface
0.0.0.0\0=00000000.00000000.00000000.00000000\0	0
149.53.128.0/17=(10010101.00110101.1)0000000.00000000\17	1
149.53.128.0/19=(10010101.00110101.100)00000.00000000\19	2
149.53.160.0/19=(10010101.00110101.100)10110.00000000\19	3
149.53.192.0/19=(10010101.00110101.110)10110.00000000\19	4
149.53.214.50=(10010101.00110101.110)10110.00110010\19	4
149.53.168.36=(10010101.00110101.100)11110.00100100\19	2
149.53.155.40=(10010101.00110101.100)11011.00101000\19	2
149.53.199.111=(10010101.00110101.110)00111.01101111\19	4
149.53.208.42=(10010101.00110101.110)10000.00101010\19	4
149.53.224.200=(10010101.00110101.110)11100.110)00100\19	4
149.53.127.11=(10010101.00110101.110)11111.00001011\19	4
149.53.179.20=(10010101.00110101.1)0110011.00010100\17	1

ΑΣΚΗΣΗ 4

Μετατρέπουμε κάθε πεδίο του header σε δυαδικό σύστημα.

Το header το μετατρέπουμε σε δυαδικό αλλά μόνο για να το συγκρίνουμε με το τελικό αποτέλεσμα. Για την ώρα θα είναι ίσο με 0.

4500=0100010100000000

+003C =0000000000111100

453C = 010001010011110

+1C46=0001110001000110

0110000110000010 → 6182

+4000=0100000000000000

1010000110000010 → A182

+4006= 0100000000000110

1110000110001000→ E188

+0(checksum)

+AC10=1010110000010000

11000110110011000→ 18D98

+1(κρατουμενο)

1000110110011001→ 8D99

+0A63=0000101001100011

1001011111111100 → 97FC

+AC10 1010110000010000

10100010000001100→ 1440C

+1(κρατουμενο)

0100010000001101→440D

+0A0B 0000101000001011

0100111000011000→ 4E18

(αντιστρεφω) 1011000111100111→ **B1E7** το οποίο είναι διαφορετικό από B1E6 άρα το λαμβανόμενο πακέτο είναι λάθος.

BONUS.

Η netsat εντολή εμφανίζει στοιχεία για το πρωτόκολλο και τις τρέχουσες

συνδέσεις tcp/ip με το flag-r εμφανίζει το routing table. Εμφανίζεται η δική μας IP,μαζί με το subnet mask του δικτύου μας.

kostas@kostas-Satellite-L655: ~

kostas@kostas-Satellite-L655:~\$ netstat -r

Kernel IP routing table

Destination	Gateway	Genmask	Flags	MSS	Window	irrt	Iface
default	147.52.144.1	0.0.0.0	UG	0	0	0	wlan0
147.52.144.0	*	255.255.254.0	U	0	0	0	wlan0

kostas@kostas-Satellite-L655:~\$