

תקשורת מחשבים ואלגוריתמים מבוזרים

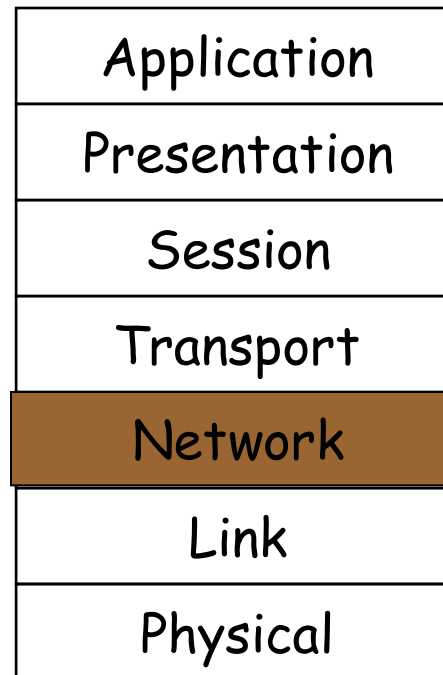


קורס מס' 202-2-1131

מתרגל: ד"ר גיא לשם leshemg@cs.bgu.ac.il

הרצאה רביעית – שכבת הרשת

חיקוק שכבת השכבת הרשת באודף 7 השכבות



The 7-layer OSI Model

שכבת הרשת

- **מבוא**
 - מעגלים וירטואלים וחבילות המידע ברשת
 - מה יש בתוך הנתב (router)
 - פרוטוקלי אינטרנט
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6
- **אלגוריתמי ניתוב:**
 - Link state
 - Distance Vector
 - Hierarchical routing
- **ניתוב באינטרנט:**
 - RIP
 - OSPF
 - BGP
- **ניתוב Broadcast ו-multicast**

תפקידי שכבת הרשת-מלא

□ תפקיד של שכבת הרשת

■ להעביר חבילות מידע (packets) מהמחשב (host) השולח למחשב המקבל.

□ פונקציות עקריות:

1. שליחת מידע ממחשב למחשב (Host-to-Host delivery).

□ מיעון (Addressing)

□ ניתוב (Routing)

■ (Routing Algorithms) אלגוריתמי ניתוב → (Path determination) קביעת נתיב

■ בחירת הממשק (port) היוצא הנכון → (Forwarding) העברה הלאה

2. קריאה להגדרת תצורה (Call Setup):

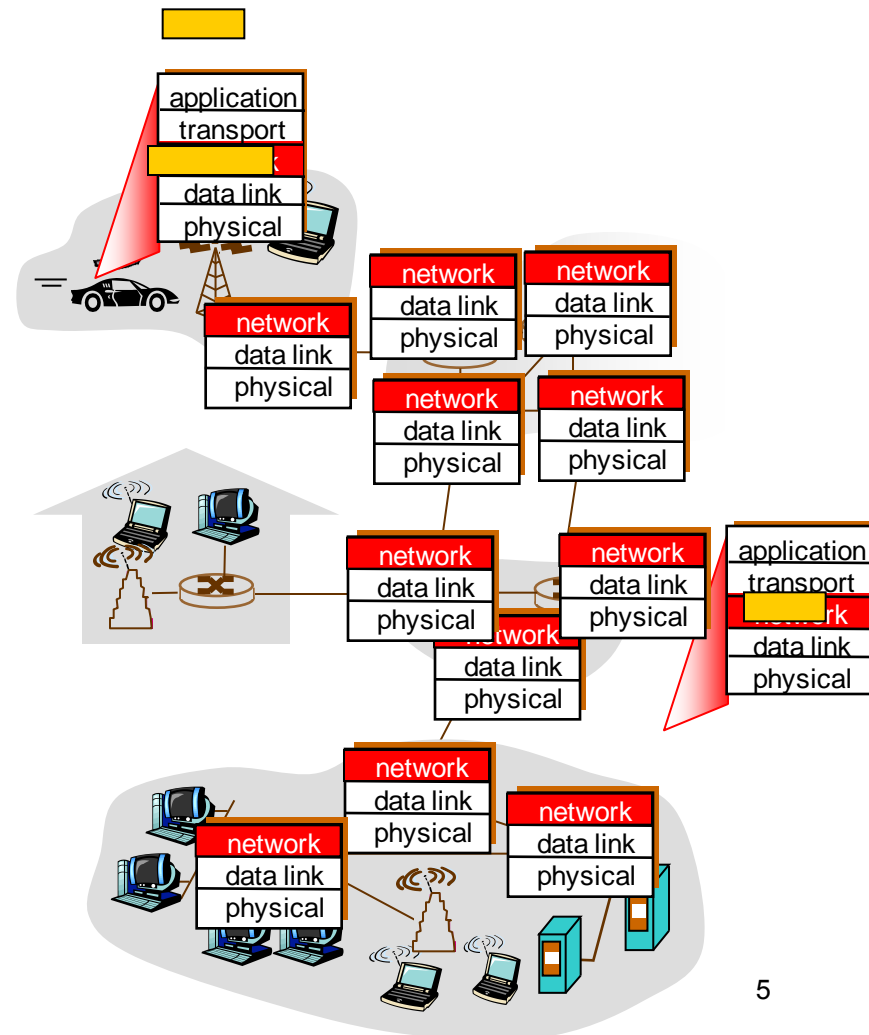
□ מספר ארכיטקטורות שכבת הרשת דורשות שהנתבים (routers) לאורך הנתיב הנבחר יסכימו בניהם על הנתיב.

3. יצירת חבילות מידע (Packetizing).

4. שבירה למקטעים (Fragmenting).

תפקידי שכבת הרשת-מבוא (המשך)

- שכבת הרשת מעבירה "מקטע (segment)" מהמחשב השולח למקבל.
- פרוטוקול שכבת הרשת קיים בכל מחשב, ובכל **נתב** (router).
- מהצד השולח שכבת הרשת עוטפת את המקטע לתוך חבילת מידע.
- מהצד המקבל שכבת הרשת מעבירה את המקטע לשכבת התעבורה.
- הנתב בוחן את שדה "הראש" בכול חבילת IP ומעביר אותה.



לפונקציות שכבת הרשת שני "מפתחות"

אנלוגיה: טיול

□ **העברה:** תהליך של קבלת נתיב דרך צומת בודדת.

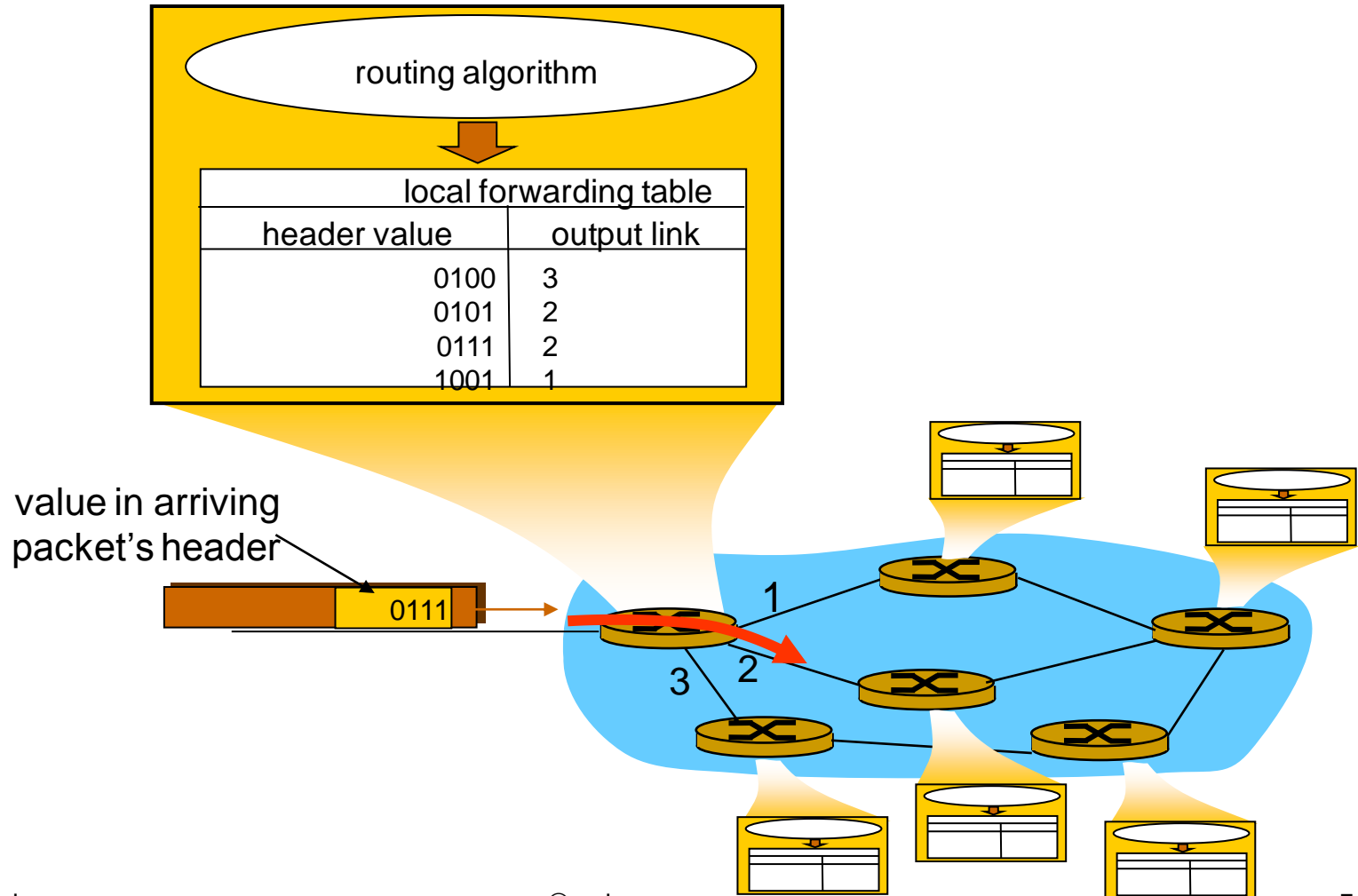
□ **ניתוב:** תהליך של תכנון טיול מהמוצא ליעד.

פרוטוקולי תקשורת

□ **העברה:** מעביר חבילות מהנתב הקלט לנתב הפלט המתאים.

□ **ניתוב:** קובע את הנתיב הנלקח ע"י החבילה מהמקור ליעד.
■ ע"י אלגוריתם ניתוב (routing algorithms).

פאולה הדדית בין העברה הלאה לניתוח



הקמת ההתקשרות

□ לדוגמא שלוש פונקציות חשובות בכמה ארכיטקטורות רשת:

■ ATM (Asynchronous Transfer Mode) - פרוטוקול להעברת נתונים בקצב גבוה.

■ frame relay - הוא פרוטוקול תקשורת לשינוע מהיר של חבילות מידע מעל גבי חיבורים וירטואליים לחיבור אל התקני רשת.

■ X.25 - מורכבת ממכלול פרוטוקולים תקיניים למיתוג מנות ברשת wan. כלומר מגדירה כיצד מוקמת ומתוחזקת תקשורת בין תחנות קצה לרכיבי רשת וציוד תקשורת נתונים.

□ לפני שחבילת המידע נשלחת, 2 נקודות קצה ונתבים מפרידים מקימים חיבור וירטואלי.

■ הנתבים נעשים מעורבים.

□ הרשת דרך שכבת התעבורה מקשרת שרותים:

■ **רשת:** בין שני מחשבים (כאשר יתכן שמערבים גם נתבים במקרה של חיבור וירטואלי).

■ **תעבורה:** בין שני תהליכים.

מודל השירות ברשת

שאלה: מהו "מודל השירות" עבור ערוץ תעבורת חבילות מהשולח למקבל?

דוגמא לשירות עבור חבילה

בודדת:

- הבטחה לשליחה.
- הבטחה לשליחה עם עיכוב הקטן מ-40 msec.

דוגמא לשירות עבור זרימה של

חבילות:

- סדר תקין של שליחת החבילות.
- הבטחה לשימוש במינימום רוחב פס בשליחה.
- הגבלה על השינויים במרחב החבילה הפנימי.

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

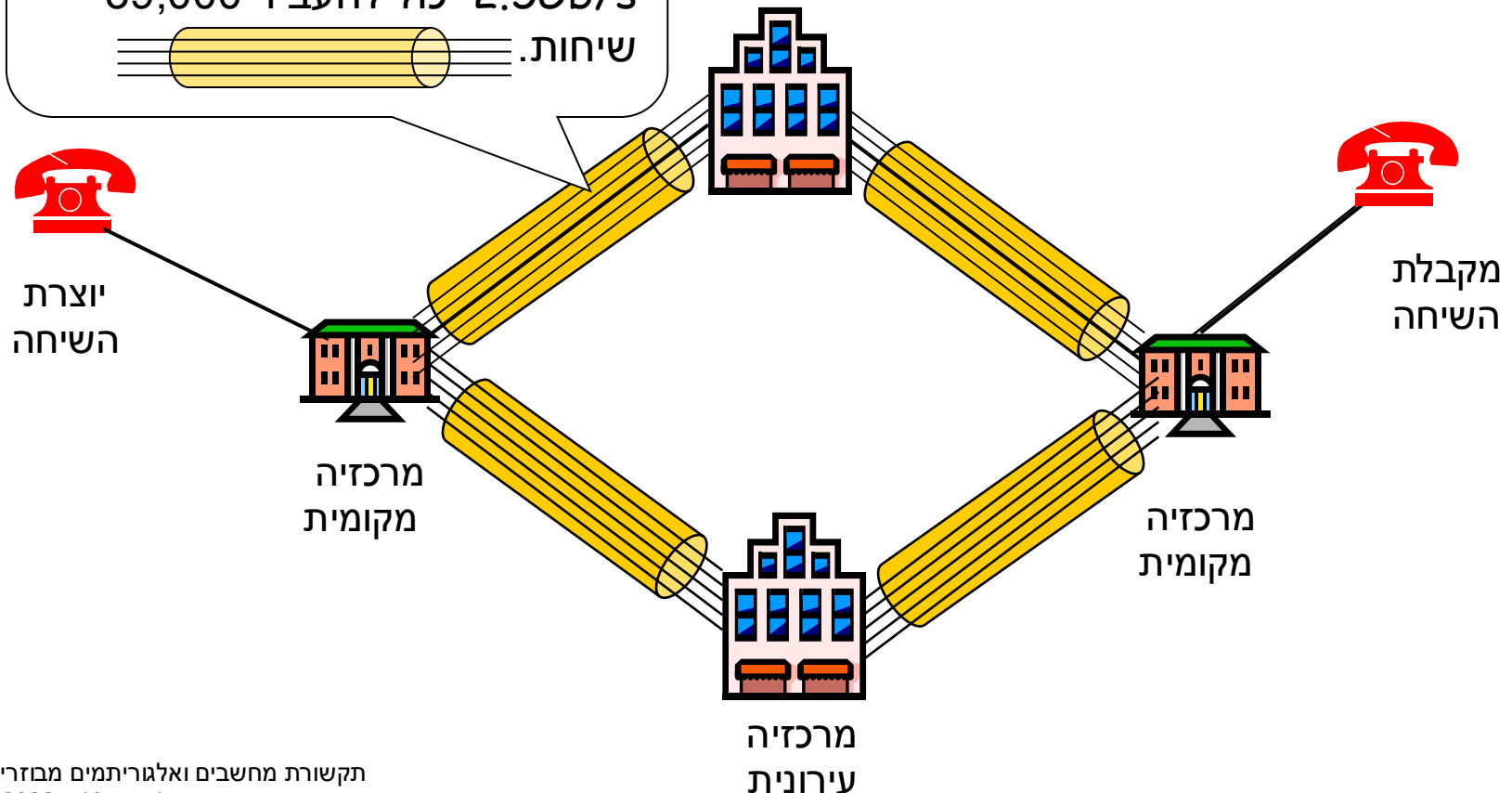
שירות fe שכבת הרשת

- רשת של מעגלים וירטואלים (VC) מספקת שירות שכבת רשת עם קישור.
- רשת של חבילות מידע מספקת שירות שכבת רשת ללא קישור.
- מקביל לשירות שכבת התעבורה, אבל:
 - **שירות:** מחשב למחשב.
 - **ללא בחירה:** הרשת מספקת לאחד או לאחרים.
 - **יישום:** בליבת הרשת.

Virtual circuits – *מיתוח מצלפים*

נסתכל תחילה אך פועלת
רשת טלפוניה

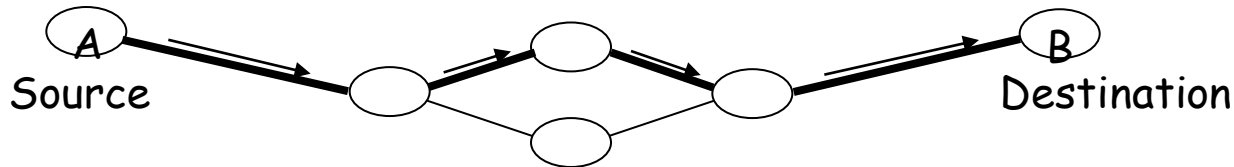
לכל שיחת טלפון מוקצה 64kb/s ,
לכן קו תקשורת עם רוחב פס של
 2.5Gb/s יכול להעביר 39,000
שיחות.



חיתוך מצלמים

"הנתיב מהמקור ליעד מתנהג כמו מעגל טלפוני"

■ הרשת פועלת לאורך הנתיב מהמקור ליעד



■ זוהי שיטה שבה משתמשת ברשת הטלפוניה, לקריאה יש מספר פאזות:
1. קביעת (בחירה) המעגל מקצה לקצה - התקשרות עצמה לפני שנתונים יכולים לזרום.

2.1 ההתקשרות עצמה – כל חבילה נושאת זיהוי של חיבור וירטואלי (VC) ולא כתובת של מחשב היעד.

2.2 כל נתב בנתיב של מקור-יעד מתחזק "מצב" עבור כל חבילה העוברת בקשר הזה.

2.3 הערוץ, משאבי הנתב (רוחב פס, זכרון זמני (buffers)) חייבים להיות מוקצים לחיבור הוירטואלי (משאבים הדבקים במטרה = שירות הניתן לחיזוי).

3. סגירת המעגל.

ייוס fe האצא הוירטואלי (VC)

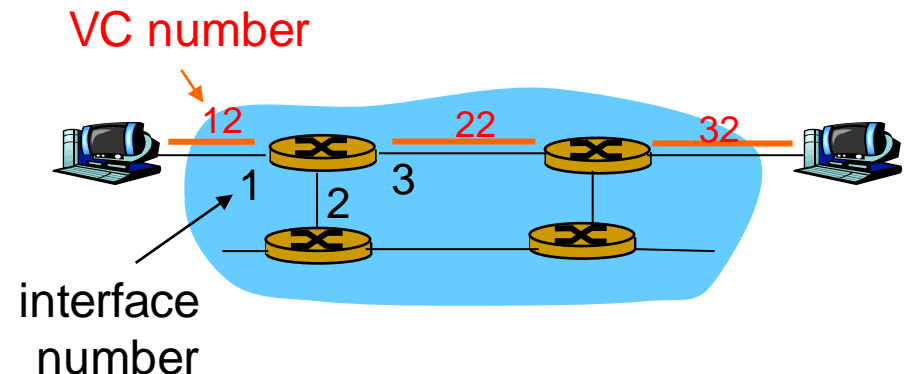
החיבור הוירטואלי מורכב מ:

1. נתיב מהמקור ליעד.
2. מיספור של המעגל הוירטואלי, מספר אחד לכל ערוץ לאורך הנתיב.
3. גישה לטבלת השליחות (forwarding) בנתבים לאורך הנתיב.

- חבילה שייכת למעגל וירטואלי נושאת את מספר המעגל הוירטואלי ולא את כתובת היעד.
- מספר המעגל הוירטואלי יכול להיות שונה בכל ערוץ.
- מספר מעגל וירטואלי חדש מגיע מטבלת השליחות (forwarding).

Forwarding table טבלת הפניה

Forwarding table in northwest router:



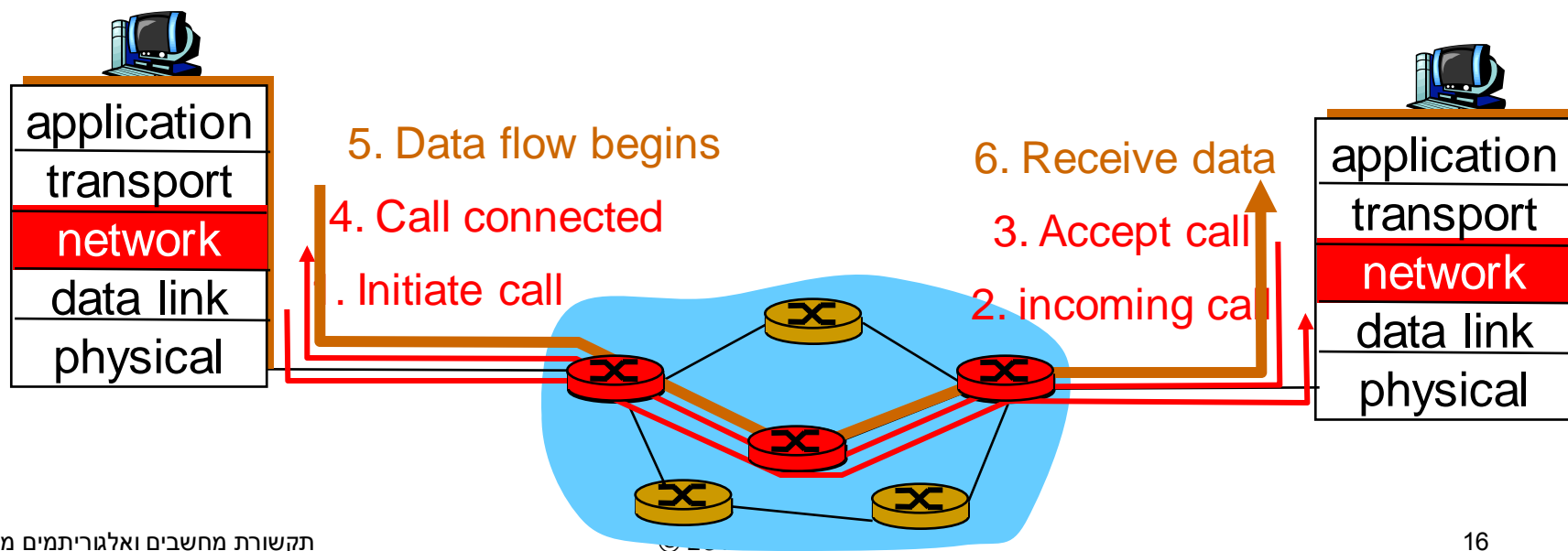
Incoming interface	Incoming VC #	Outgoing interface	Outgoing VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
...

Routers maintain connection state information!

צוואר תעבורה וירטואלי

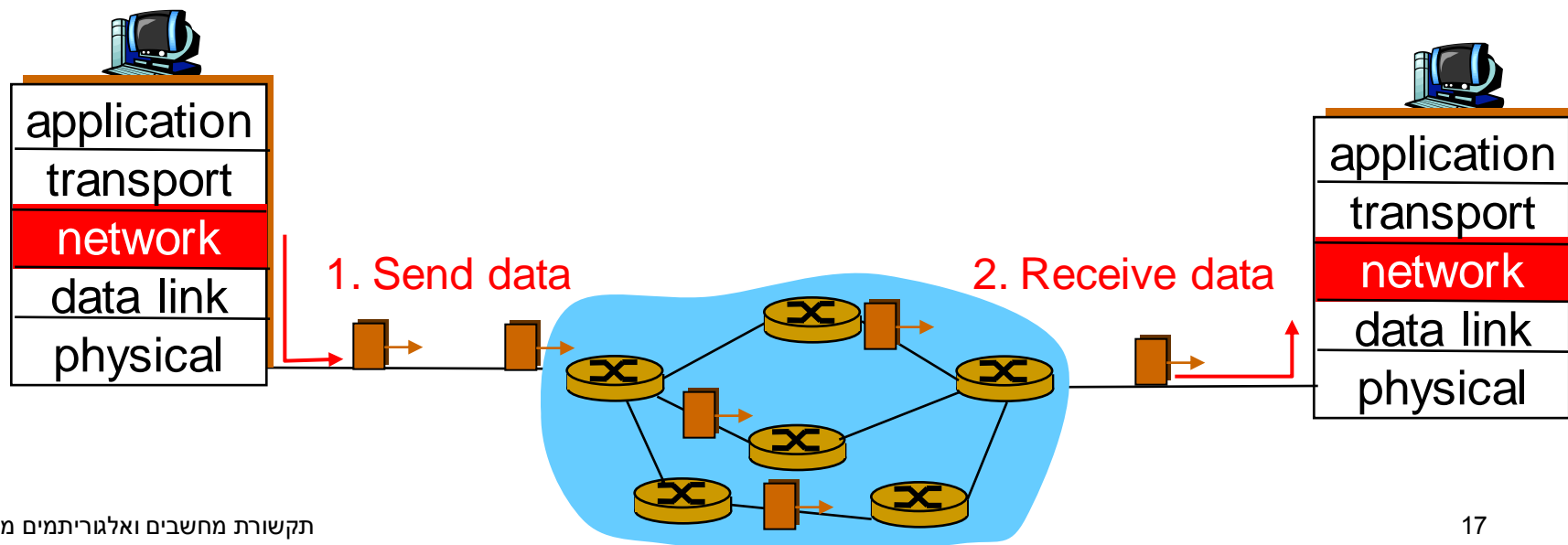
סיכום לשימוש במעגל וירטואלי (Virtual circuits):

- משמש להקמת תקשורת, תחזוקה, הריסת מעגלים וירטואליים.
- משתמש בפרוטוקולים: ATM, frame-relay, X.25.
- לא שימושי כיום ברשת האינטרנט.



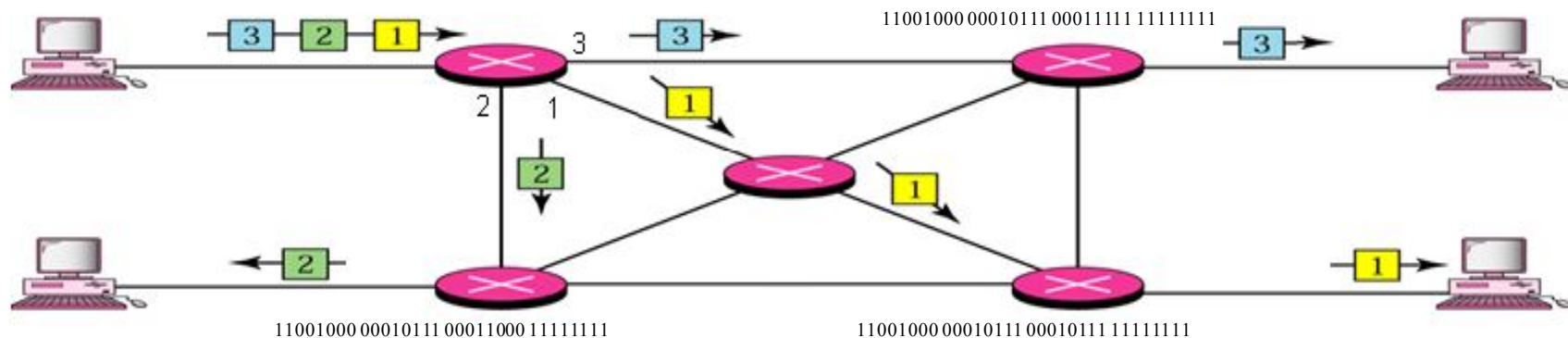
אינטרנט חבילות המידע Datagram networks

- השיטה בה משתמשים באינטרנט.
- כל חבילה בנפרד נשלחת לדרכה ובוחרת את נתיבה, כך חבילה אחר חבילה, כאשר הן משתמשות בטבלת הניתוב המקומית של הנתב (ראוטר-router).
- הנתב מתחזק לא לפי מצב זרימה מסוים.
- חבילות שונות יתכן ויבחרו נתיבים שונים.
- מספר חבילות יכולות להגיע בערוץ מסוים באותו זמן, לכן לגישת חבילת המידע יש חוצץ (מקום אחסון זמני בזיכרון).



Forwarding table *ጠ/ከ'ቺ ክፍሪ*

<u>Destination Address Range</u>	<u>Link Interface</u>
11001000 00010111 00010000 00000000	1
11001000 00010111 00011000 00000000	2
11001000 00010111 00011001 00000000	3



אישת חבילות המידע או אישת מצלם ורטואלי ?

ATM (מעגל ורטואלי):

- התפתח מרשת הטלפוניה.
- בדומה לשיחה אנושית:
 - דורש תזמון מושלם ואמינות.
 - נדרש להבטחת השירות.
- קצה המערכת "אילם-טיפש":
 - לכן נצטרך לטלפן.
 - רשת מסובכת מבפנים.

אינטרנט (חבילות מידע)

- מידע מוחלף בין המחשבים.
 - שירות גמיש לא נדרש תזמון מושלם.
- קצה המערכת "חכם" (מחשב):
 - יכול לעבד נתונים, לבצע בקרה, להתאושש משגיאות.
 - רשת פשוטה מבפנים, מסובכת רק בקצה.
- סוגי ערוצים רבים:
 - מאפינים שונים.
 - השירות האחד קשה ליישום.

השאלות שירותים בסיסיים

ATM (מעגל ורטואלי)

כאן אנחנו בונים נתיב מהמקור ליעד. השירות דומה לזה של רשת טלפונים המאפשרת לנו תקשורת real time מרגע שהוקם נתיב.

לחבילות אין את כתובת היעד, אלא מזהה של הנתיב, כי היעד ידוע לפי הנתיב, וכמו כן ידועים כל הנתיבים בדרך.

המשאבים של הנתיב (רוחב פס, חוצצים וכד') מוקצים בזמן הקמת הנתיב, ומספקים שירותים שונים כמו בקרה על אובדן חבילות, קצב שידור קבוע, העברת חבילות לפי הסדר, ורוחב פס קבוע.

עבור רשתות שתומכות במולטימדיה, כמו טלוויזיה בכבלים, תכונות כמו קצב שידור קבוע ושמירה על סדר החבילות הן תכונות חיוניות. עבור העברת מידע בין מחשבים הן פחות חשובות.

אינטרנט (חבילות מידע)

- השירות אותו אנחנו מקבלים ברשת האינטרנט, זהו שירות של best effort, כלומר הוא משתדל להעביר חבילות בלי טעות, אבל הוא לא מתחייב על כך. אין הקמת ערוץ שיחה, אין **connection**.
- לחבילות יש את כתובת IP של מחשב היעד, וחבילות שונות מאותו מקור יכולות להגיע לאותו היעד בנתיבים שונים.

מודלי שירותי שכת הרשת

Network Architecture	Service Model	Guarantees ?				Congestion feedback
		Bandwidth	Loss	Order	Timing	
Internet	best effort	none	no	no	no	no (inferred via loss)
ATM	CBR	constant rate	yes	yes	yes	no congestion
ATM	VBR	guaranteed rate	yes	yes	yes	no congestion
ATM	ABR	guaranteed minimum	no	yes	no	yes
ATM	UBR	none	no	yes	no	no

CBR - Constant Bit Rate

VBR - Variable Bit Rate

ABR - Available Bit Rate

UBR - Unspecified Bit Rate

ATM - כל שירות ברשת מגדירים מסלול תעבורה מקצה לקצה. כלומר, אם מידע מסוים ממתג A צריך להגיע למתג D, והוא עובר בדרכו את מתגים B ו-C, תוכנת הניהול של הרשת צריכה לקבוע הגדרות מתאימות בכל המתגים שהמידע עובר דרכם, כדי שיוכל להגיע ליעדו. חלק מההגדרות לכל מתג הן איכות השירות בכל כניסה במתג.

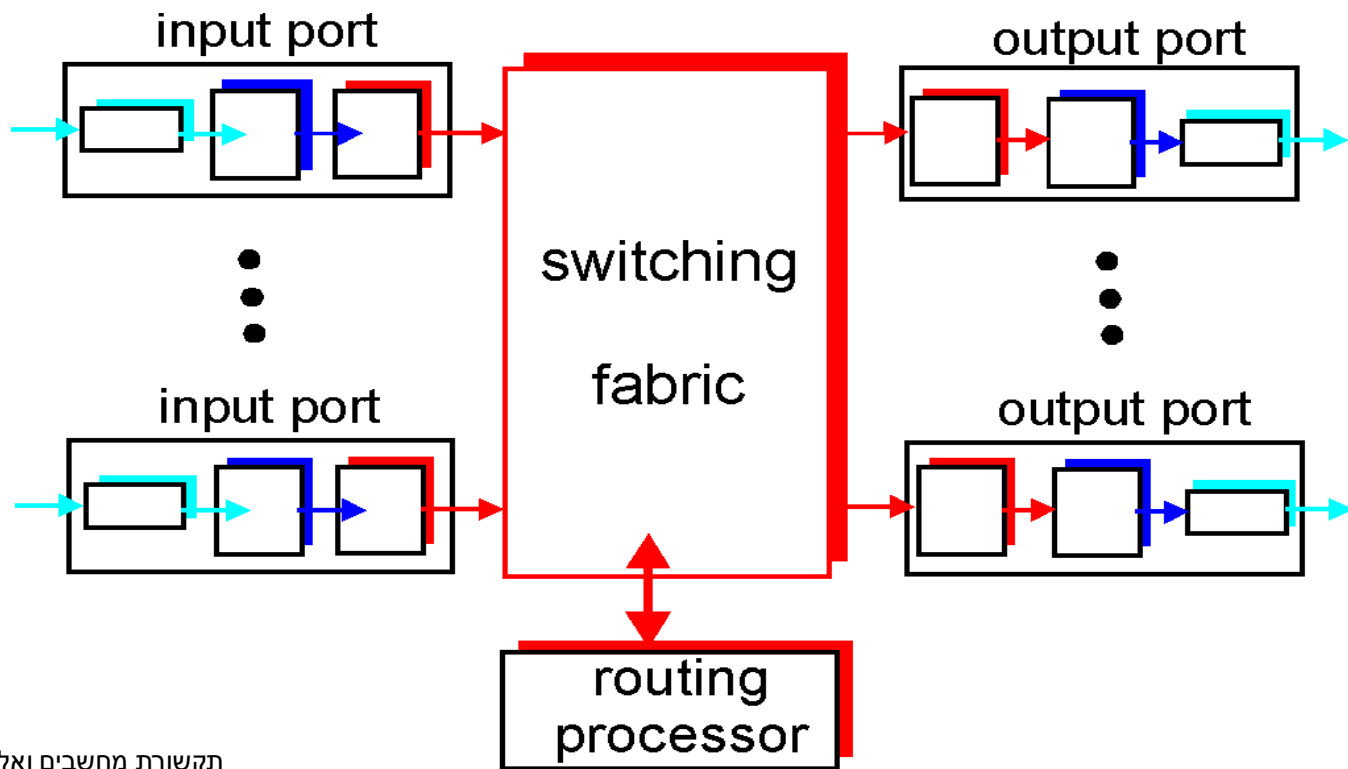
IP - מספק קישוריות חסרת חיבור (connectionless) ושולח נתונים במירב המאמצים (best effort).

שכבת הרשת

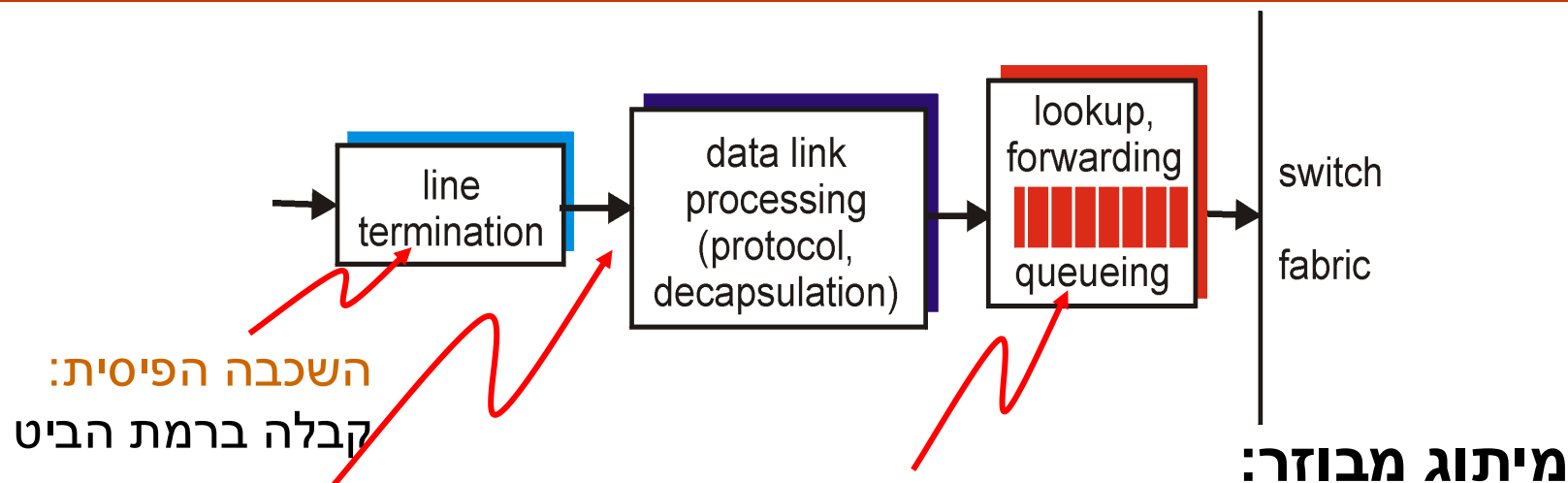
- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast

מהט על ארכיטקטורת הנתב

- לנתב שתי פונקציות מפתח:
- להריץ אלגוריתמי/פרוטוקולי ניתוב (RIP, OSPF, BGP).
- לשלוח חבילות מידע מערוצי קלט לערוצי הפלט.

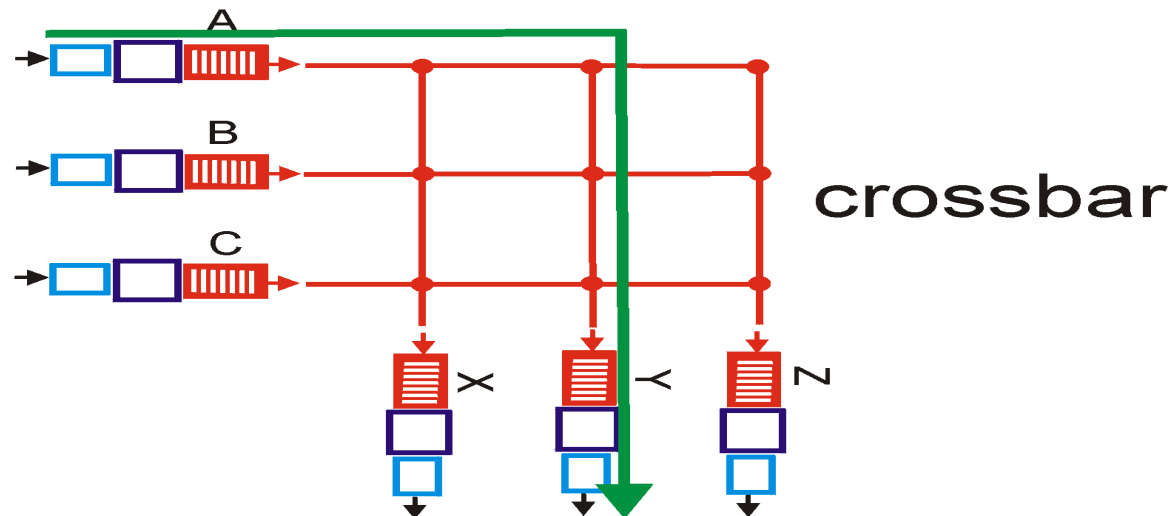
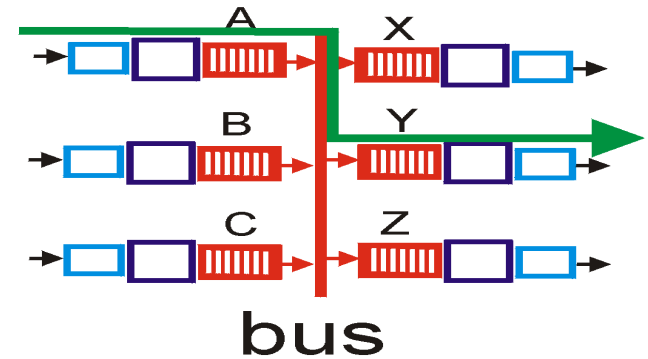
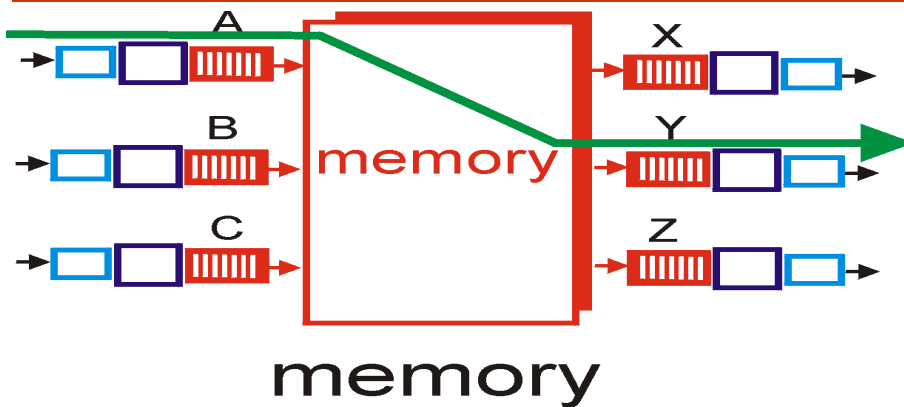


פונקציות ממשקי הקלט



- נתון יעד חבילת המידע, מתבצע חיפוש ממשק הפלט תוך שימוש בטבלת השליחות (forwarding table) הנמצאת בזכרון ממשק הקלט.
- המטרה: להשלים את תהליך ממשק הקלט במהירות הערוץ.
- עמידה בתור: אם חבילת המידע מגיע מהר יותר מאשר קצב השליחה של מבנה המיתוג (switch fabric).

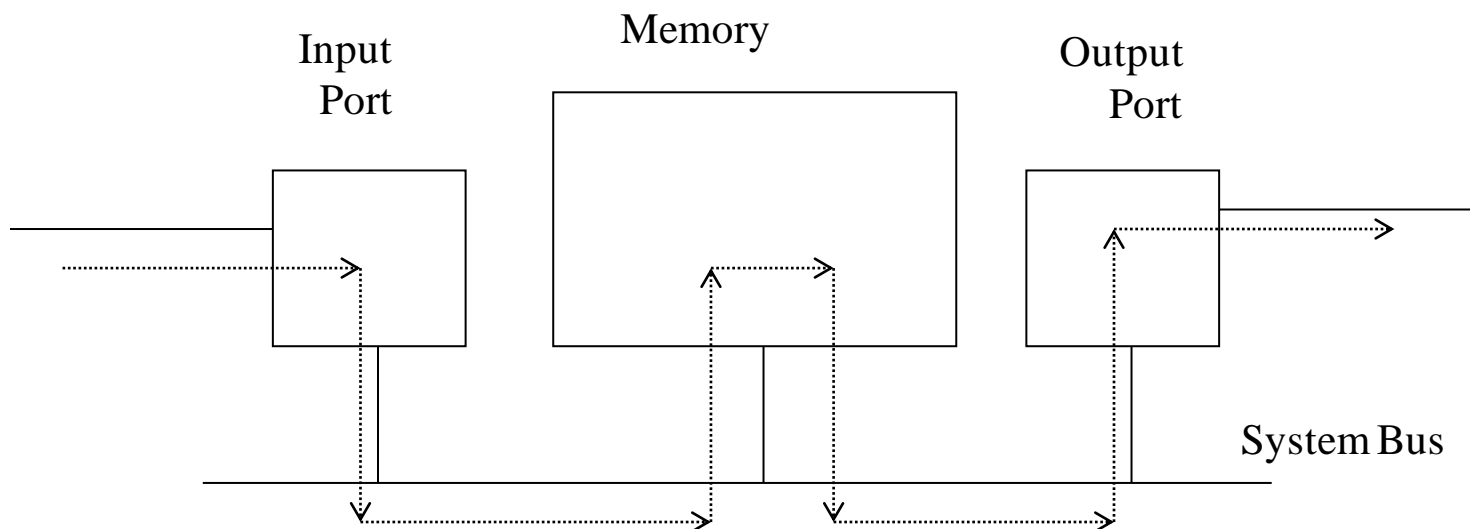
עלוש סולס el אנה הולול (switching fabrics) בנה



חיתול באמצעות הלכרון

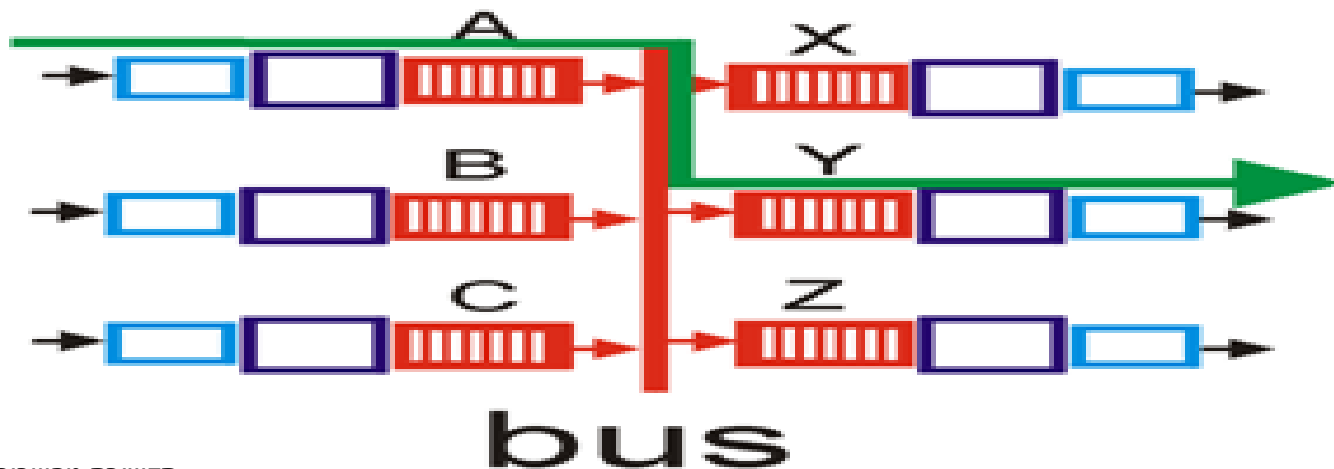
הדור הראשון של הנתבים:

- באופן מסורתי מחשבים עובדים עם מיתוג תחת בקרה ישירה של המעבד המרכזי (CPU).
- חבילות מועתקות למערכת הזכרון.
- המהירות מוגבלת ע"י רוחב הפס של הזכרון.



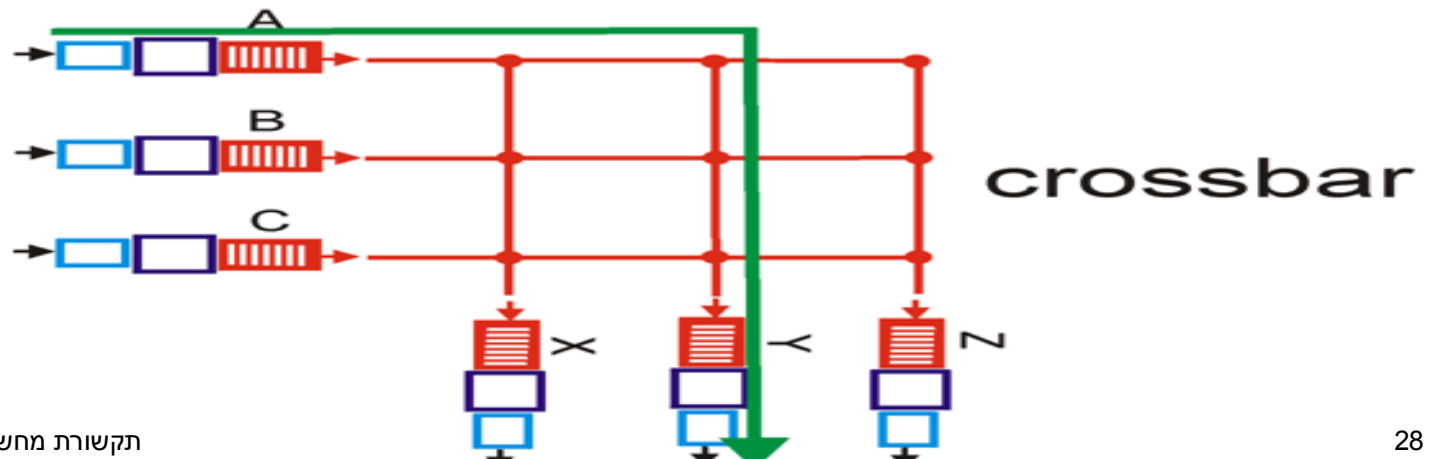
חיתוך באמצעות האפיק (Bus)

- חבילות מידע עוברות מזכרון ממשק הקלט לזכרון ממשק הפלט דרך אפיק משותף.
- קישוריות האפיק:** מהירות ההמיתוג מוגבלת ע"י רוחב הפס של האפיק.
- בנתב Cisco 5600, רוחב הפס של האפיק 32 Gbps : 10 לדוגמא מהירות המספיקה לביצוע פעולות של הנתב

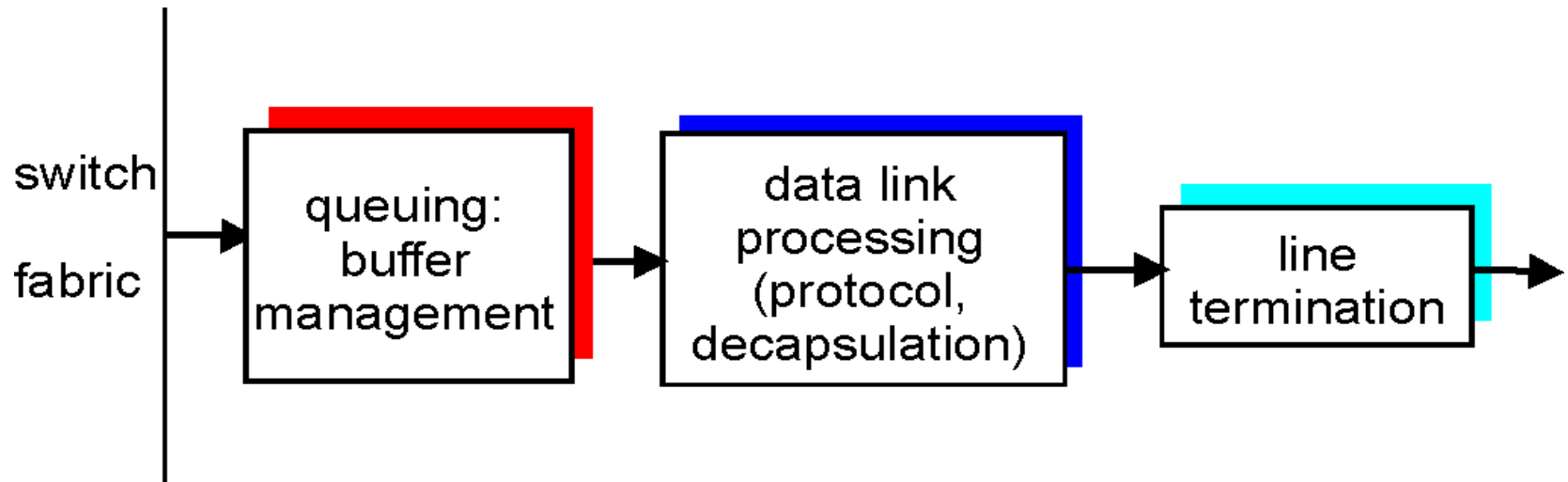


איתול דרך רשת fe קישור הדדי

- מתגבר על מגבלות רוחב הפס של האפיק.
- רשת Banyan, רשת פנימית של קישור הדדי הפותחה על מנת לקשר בין מעבדים במערכות ריבוי מעבדים.
- תכנון מתקדם: "שובר" את חבילות המידע לתוך תאים בעלי אורך קבוע, וממתג את התאים דרך המבנה.
- לדוגמא נתב Cisco 12000: ממתג 60 Gbps דרך רשת של קישור הדדי.



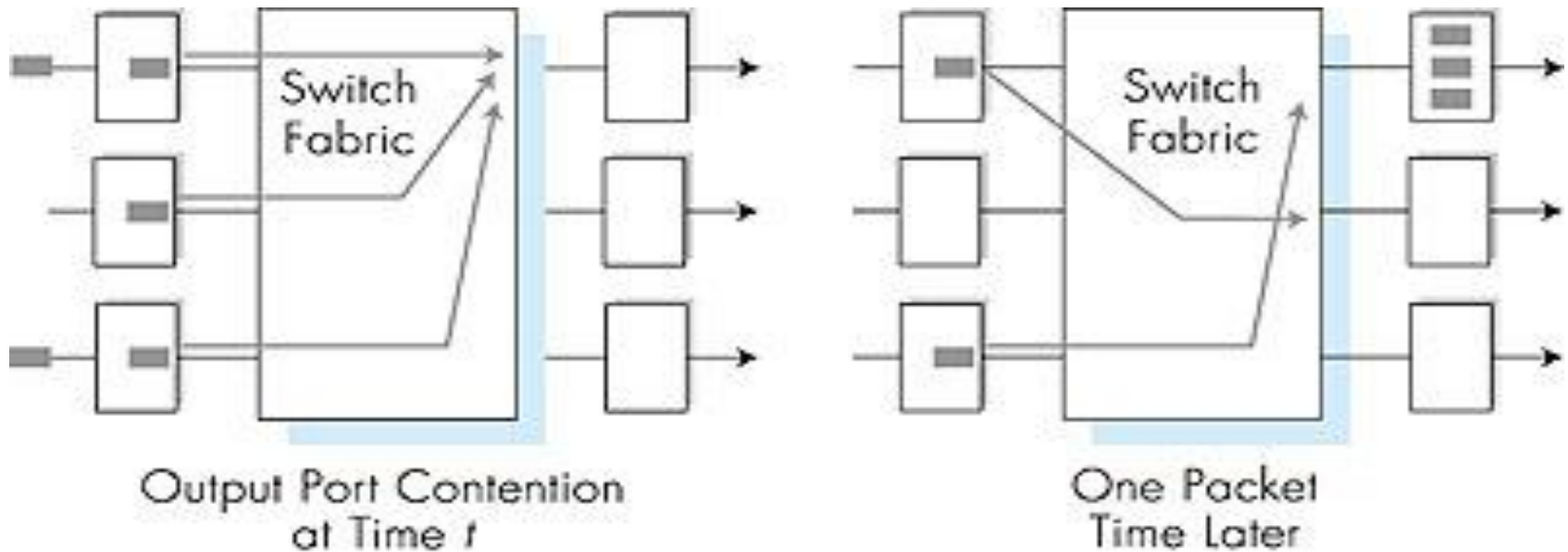
מעק הפלט



□ **חוצץ** נדרש כאשר חבילות מידע (datagrams) מגיעות מהמבנה המיתוג (switch fabric) יותר מהר מאשר קצב השידור.

□ **סינכרון זמנים** נדרש לבחור בין התורים של חבילות המידע עבור שידורם.

תורים בממשק הפלט



■ שימוש בחוצץ כאשר קצב ההגעה דרך מבנה המיתוג עולה על מהירות ערוץ הפלט.

■ עמידה בתור (עיכוב) ויתכן גם אובדן מידע הודות לגלישה של החוצץ (תקלה בה התכנית מנסה לאחסן נתונים כשאין מקום בזיכרון) של ממשק הפלט.

כמה "חציצה" נדרשת ?

□ כלל אצבע לפי RFC 3439 (Request For Comments)

□ בממוצע חוצץ שווה ל-

RTT (round trip time) טיפוסית (~250msec)
פעמים קיבולת הערוץ C.

■ לדוגמא: 2.5 Gbit buffer → link: 10 Gps = C.

□ ההמלצה האחרונה: עם N זרימות, החוצץ שווה ל-

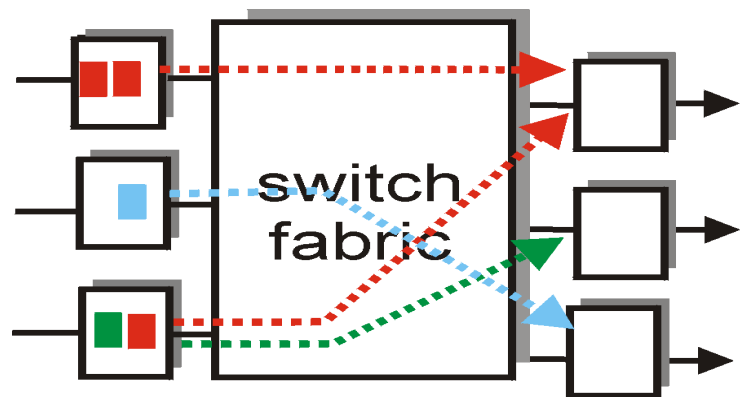
$$\frac{RTT \cdot C}{\sqrt{N}}$$

RTT (round trip time) - כמה זמן לוקח להודעה להגיע מקצה אחד של הרשת לקצה השני ובחזרה (ack).
31

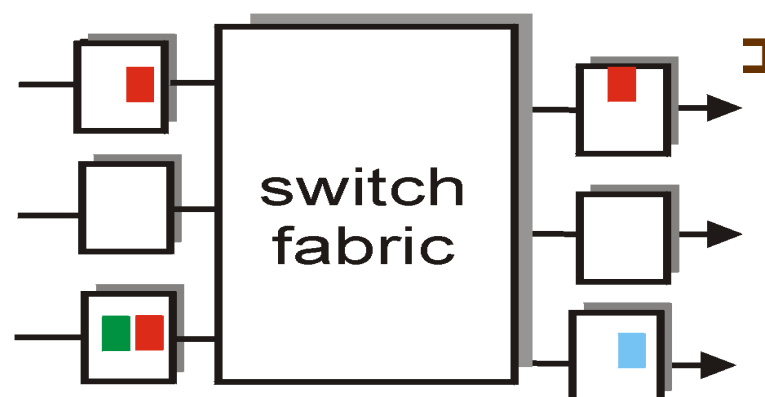
תורים בממשק הקלט

מבנה המיתוג (switch fabric) בנתב איטי יותר מאשר ממשקי הקלט המשולבים ← עמידה בתור יכולה לקרות כתוצאה מתורים בקלט.

חסימה בראש התור (Head-of-the-Line (HOL) blocking): יצירת תור של חבילות המידע בחזית של התור מונע מהאחרים בתור להתקדם.



output port contention
at time t - only one red
packet can be transferred



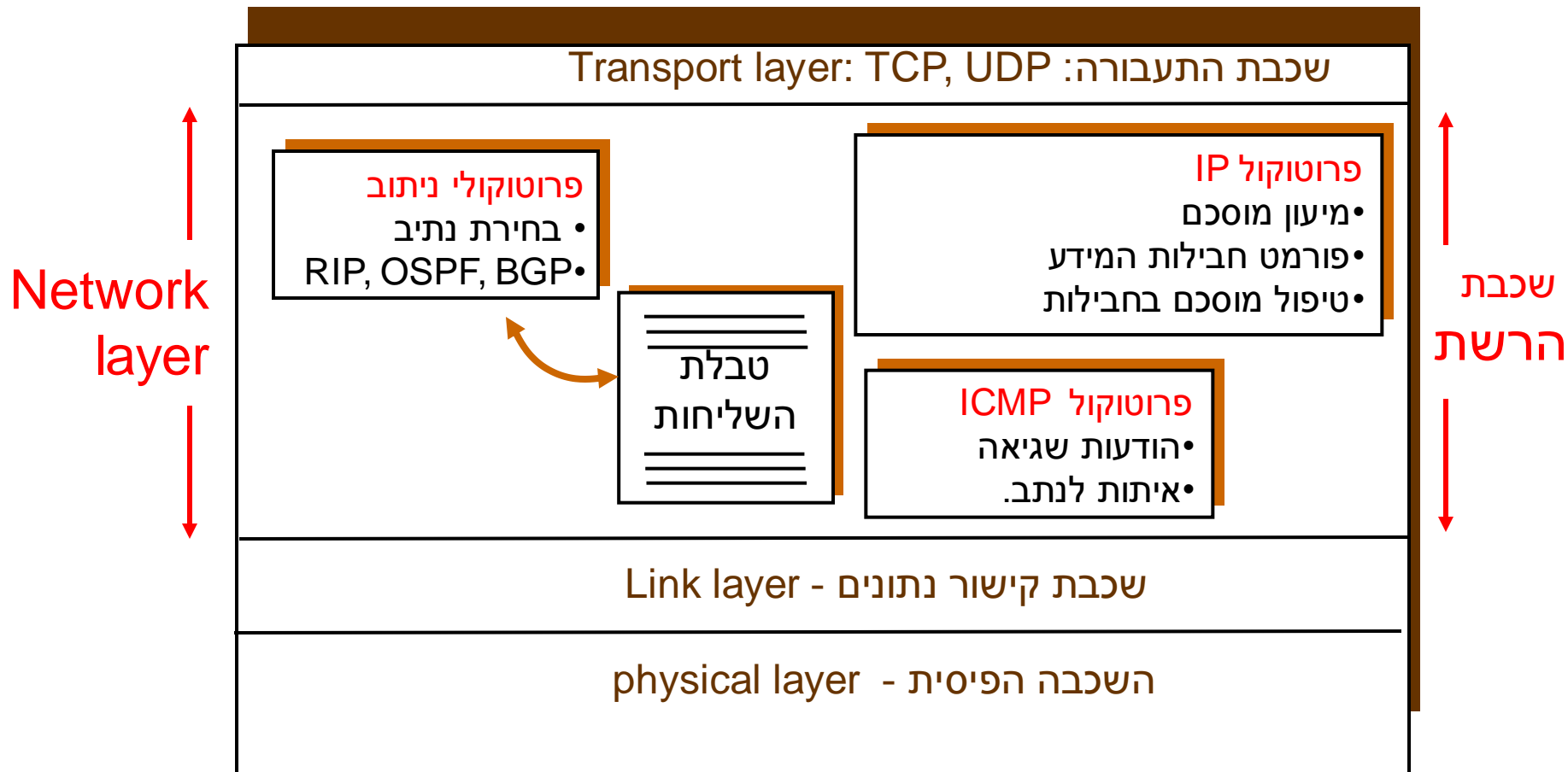
green packet
experiences HOL blocking

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

שכבת הרשת באינטרנט

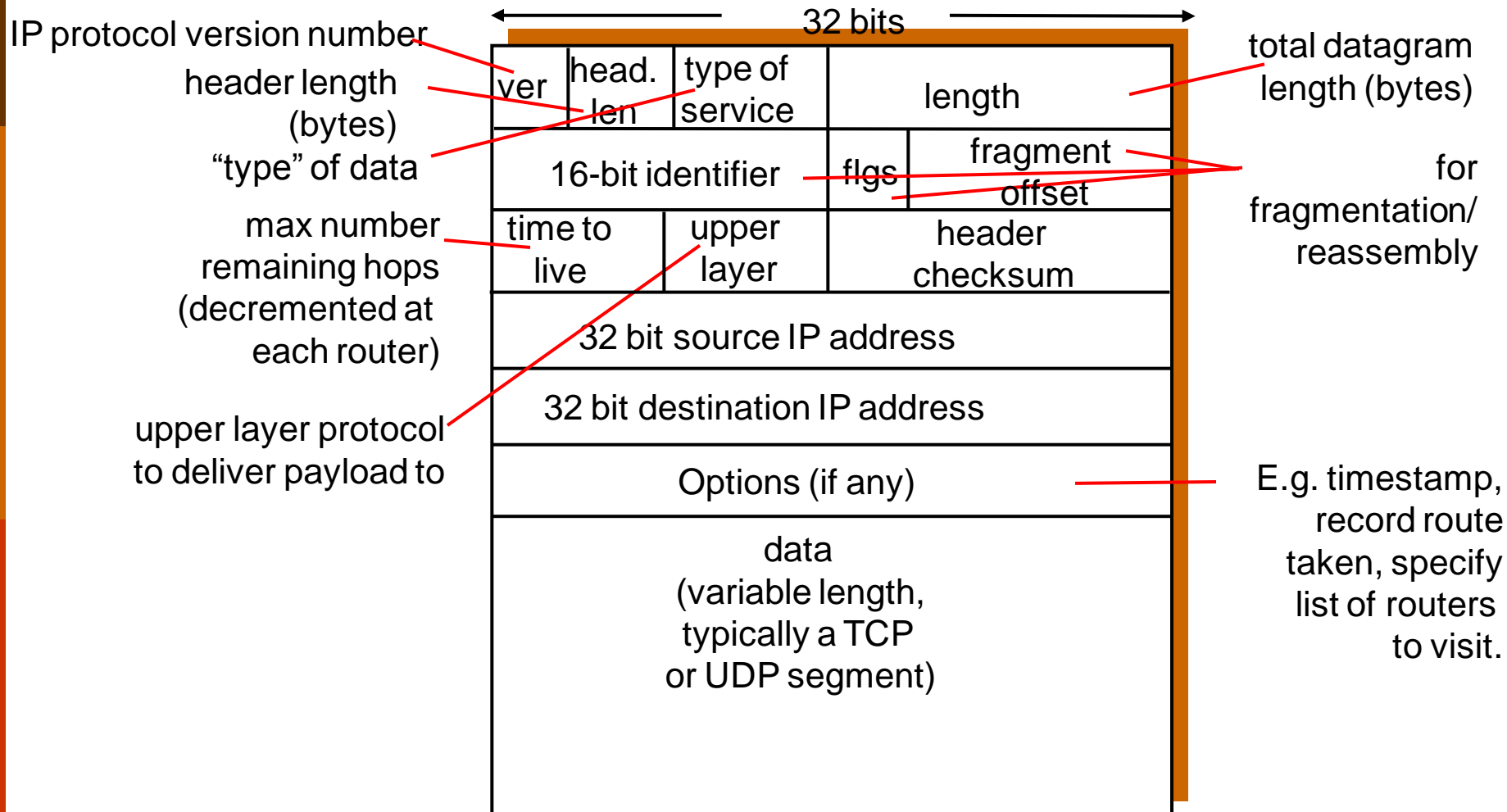
פונקציות שכבת הרשת



שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

פורמט חבילות IP



מבנה חבילת IP

4 • הסיביות הראשונות משמשות לסימון גרסת הפרוטוקול שבו נעשה שימוש. כיום נפוץ השימוש בגרסה 4 (IPv4) אך עקב מצוקת כתובות מתחיל לחדור השימוש בגרסה 6, IPv6.

4 • הסיביות הבאות מגדירות את אורך הפתיח, ביחידות של 32 סיביות. מכיוון שפתיח של חבילת IP יכולה להכיל נתונים שונים, בהתאם לאפשרויות השונות בהן נעשה שימוש, ולכן אורך הפתיח משתנה בין חבילה לחבילה. לדוגמה, פתיח בן 20 בתים יקבל את האורך 5 (20 בתים הם 160 סיביות). יש לשים לב שאורך הפתיח לא יכול לעלות על 60 בתים.

8 • הסיביות הבאות נועדו במקור כדי לאפיין את סוג השירות, (Type of Service) – ToS כלומר, האם על החבילה להעדיף מעבר מהיר על פני אמינות גבוהה.

16 • הסיביות הבאות מגדירות את גודלה הכולל של החבילה, ביחידות של בית אחד. הגודל המינימלי לחבילה מוגדר להיות 576 בתים, והגודל המקסימלי הוא 65535 בתים.

16 • הסיביות הבאות מייצגות את "מספר הזיהוי" של החבילה. שדה זה משמש בעיקר לצורך הרכבה מחדש של הודעות אשר חולקו. במקרה בו יש מספר רב של חלקים (Fragments), הצד המקבל צריך לדעת את מי מהחלקים להרכיב מחדש. בזמן חלוקה של הודעה, מספר זה חייב להישמר. מערכת ההפעלה בדרך כלל מעלה את המספר באופן סדרתי בכל הודעה שיוצאת מהמחשב.

3 • הסיביות הבאות משמשות במקרים שבהם המידע מחולק בין יותר מחבילה אחת. במקרה כזה הסיבית הראשונה תכיל 1, ואם זו היא החבילה האחרונה לאותו מידע מחולק גם הסיבית האחרונה תכיל 1. בכל מקרה אחר הן יכילו 0.

13 • הסיביות הבאות משמשות לזיהוי הסדר שבו צריך להרכיב את המידע, במקרה שהוא מחולק בין כמה חבילות. שדה זה מכיל את המיקום היחסי של המידע בחבילה הנוכחית, ביחס לחבילה המקורית שעברה חלוקה. המיקום נתון ביחידות של 8 בתים.

8 • הסיביות הבאות מייצגות את "זמן החיים" של החבילה – TTL (Time To Live) השדה מקבל מספר כלשהו בעת השליחה, ובכל תחנה שהחבילה עוברת בדרך המספר מוקטן באחד. אם "זמן החיים" של החבילה הגיע לאפס לפני שהיא הגיעה ליעדה, היא מושמטת ולא מועברת הלאה. מנגנון זה הומצא כדי למנוע מחבילות שלא יכולות להגיע ליעדן להסתובב באופן אינסופי ברשת.

8 • הסיביות הבאות מייצגות מספר, הקובע לאיזה פרוטוקול יש להעביר את המידע לאחר שהועבר. לדוגמה - 1 UDP, 17 TCP, 6 ICMP.

16 • הסיביות הבאות מכילות מספר שנועד לוודא את המעבר התקין של הפתיח. מפעילים על הפתיח פונקציה ידועה, שעל-פי נתונים שונים שהוא מכיל מחזירה מספר בן 16 סיביות. התחנה המקבלת מפעילה את אותה הפונקציה על הפתיח ומשווה את התוצאה עם תכולת השדה על מנת לוודא שהפתיח הגיע תקין. **IP לא מוודא את אמינות הנתונים, אלא רק את אמינות הפתיח.**

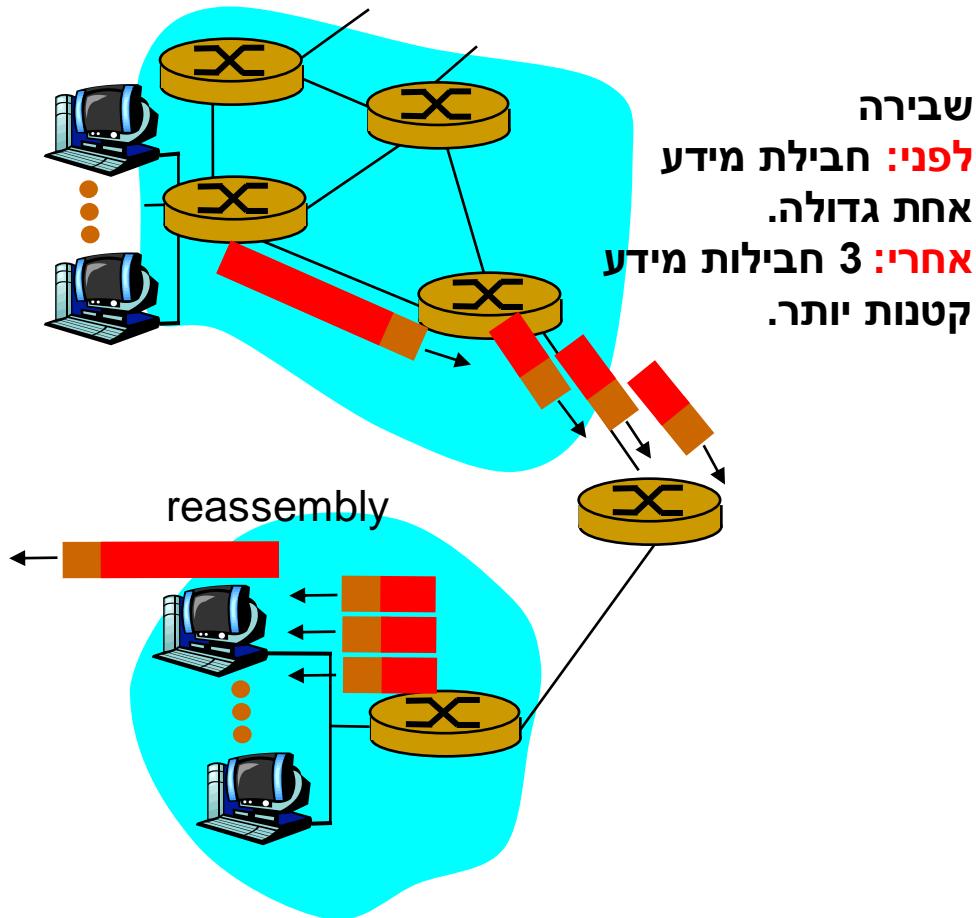
32 • הסיביות הבאות מכילות את כתובת ה-IP של השולח.

32 • הסיביות הבאות מכילות את כתובת ה-IP של הנמען.

• אופציות שונות למשל אפשרויות אבטחה.

• המידע DATA.

שבירה וכינוס מחדש IP



- לשכבת הרשת יש MTU (Maximum Transmission Unit) שזו חבילת הנתונים הגדולה ביותר הניתנת להעברה כיחידה אחת ברשת של ממסר חבילות נתונים.
- לכל סוג של ערוץ יש MTU שונה.
- חבילת IP גדולה מתחלקת ("נשברת") בתוך הרשת:
- חבילת מידע הופכת מספר חבילות מידע.
- כינוס מחדש רק ביעד הסופי.
- ביט הראש משמש לזיהוי לכל השברים.

שפירה וכינוס אחד IP

Example

4000 byte datagram

MTU = 1500 bytes

1480 bytes in
data field

offset =
1480/8

	length =4000	ID =x	fragflag =0	offset =0	
--	-----------------	----------	----------------	--------------	--

One large datagram becomes
several smaller datagrams

	length =1500	ID =x	fragflag =1	offset =0	
	length =1500	ID =x	fragflag =1	offset =185	
	length =1040	ID =x	fragflag =0	offset =370	

נעבור על השדות הרלוונטיים של חבילות IP.

שדה ID - בחבילה המקורית היה 12, והוא לא משתנה לאחר חלוקה לחבילות קטנות יותר, זה חיוני להרכבת החבילה המקורית ביעד.

שדה flag - בחבילה ראשונה ושנייה שווה לאחד כיוון שיש עוד חבילות אחריהן ובחבילה שלישית שווה לאפס כיוון שזאת חבילה אחרונה.

שדה length - מציין גודל החבילה כמובן שהוא כולל גם 20 בתים של IP header.

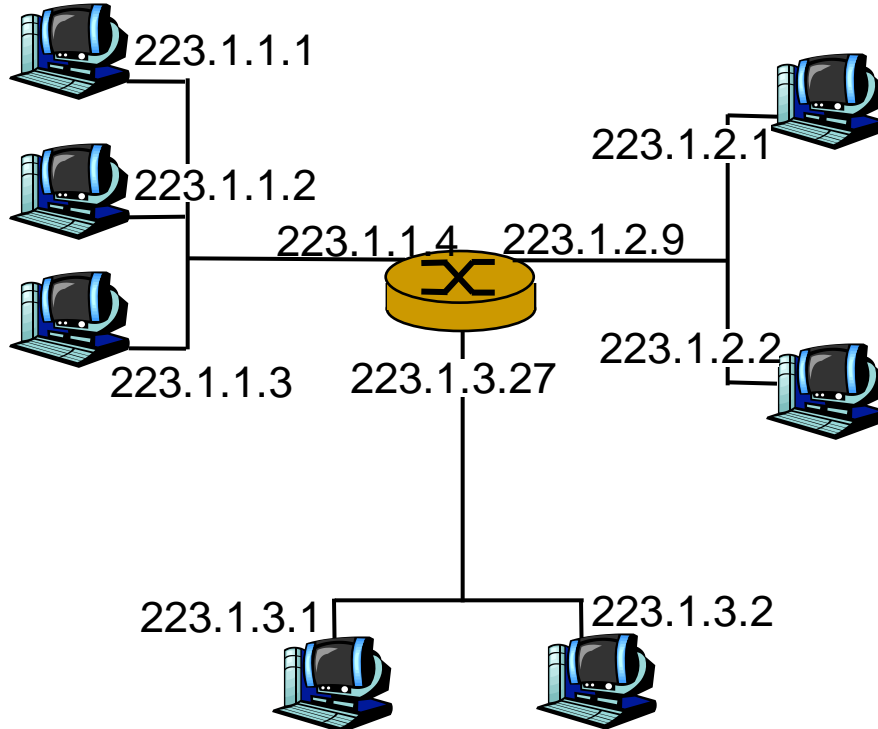
שדה offset - מציין את ההיסט של data, שמכילה חבילת IP קטנה מתחילת data של חבילת IP המקורית.

על פי שדה זה נוכל להרכיב ביעד את החבילה המקורית בסדר הנכון. בדוגמתנו בחבילה הראשונה שדה זה שווה לאפס בחבילה השנייה הוא 1480 בגלל 20 בתים של header שבחבילה ראשונה ובחבילה השלישית 2960. שוב זה לא 3000 בגלל 40 בתים של headers של שתי חבילות ראשונות. בדוגמא זאת קיימת תוספת של עוד 40 בתים של IP headers.

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

מבוא לאינטרנט IP



223.1.1.1 = 11011111 00000001 00000001 00000001

223

1

1

1

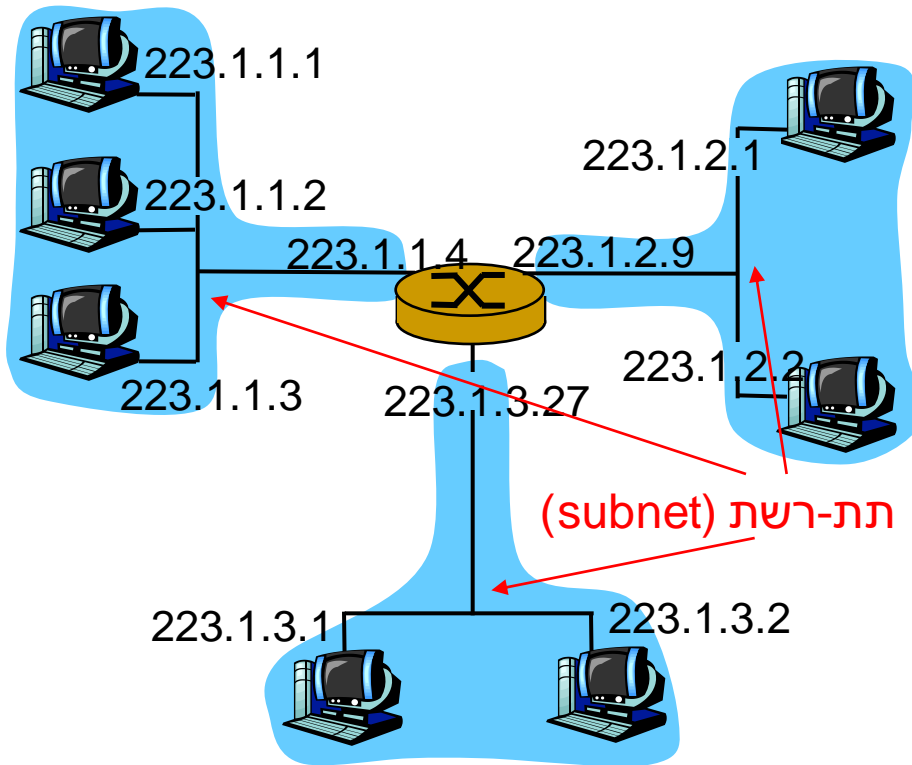
□ **כתובת IP:** 32 ביט לזיהוי המחשב (host), וממשק הנתב (router interface). מקובל לייצג ערך זה כארבעה מספרים עשרוניים, כל אחד בין 0 ל-255.

□ תאורטית, שיטה זו מאפשרת עד 2 בחזקת 32 (מעל 4 מיליארד) כתובות שונות, אולם, מכיוון שטווחים גדולים של כתובות שמורים למטרות מיוחדות, מספר הכתובות השמישות קטן יותר. למרות שהמספר נכבד, כמות זו ככל הנראה לא תספיק בעתיד.

□ **ממשק (interface):** מחבר בין נתב/מחשב והשכבה הפיסית:

- בד"כ לנתב יש מספר ממשקים
- למחשב בד"כ יש ממשק אחד.
- כתובת IP מקושרת עם ממשק.

Subnets - תת-רשתות



הרשת כוללת 3 תתי רשתות

■ כתובת IP:

- חלק התת-רשת (סדר ביטים גבוה).
- חלק המחשב (סדר ביטים נמוך)

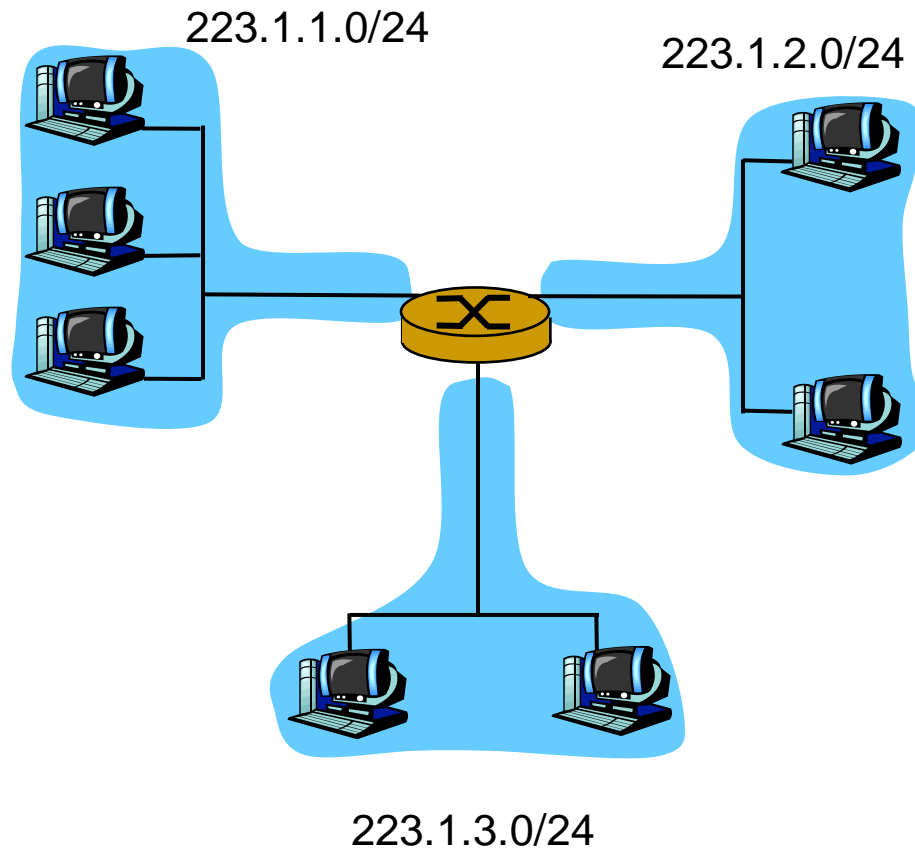
■ מהי תת-רשת ?

- רכיב חמרה שיכול לקבל או לשלוח נתונים עם אותו חלק של תת-רשת בכתובת IP.
- יכול פיסית להשיג כל התקן אחר בתת הרשת ללא התערבות של נתב.

תת-רשתות

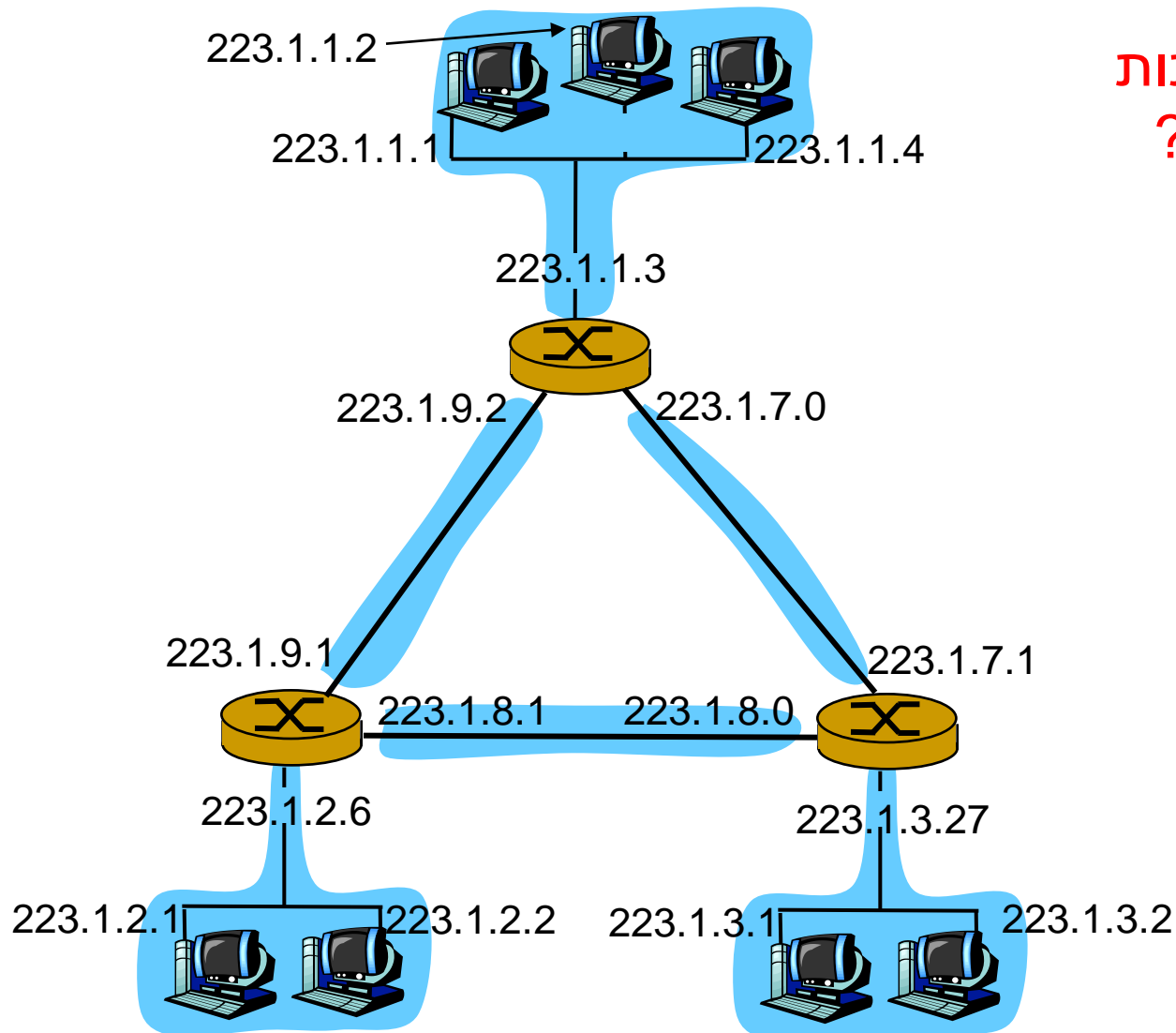
מתכון:

- לקביעת תת-רשת, ננתק כל ממשק מהמחשב או הנתב, ניצור כך ע"י בידוד רשת מבודדת.
- לכל רשת בודדת נקרא **תת-רשת**.



Subnet mask: /24

תת-רשתות



כמה תת-רשתות
ניתן ליצור ?
6

CIDR כתובות IP:

CIDR: Classless Inter-Domain Routing

- חלק תת-רשת של הכתובת של האורך השרירותי.
- פורמט הכתובת: $a.b.c.d/x$, כאשר x הוא # הביט בחלק תת הרשת של הכתובת.



200.23.16.0/23

מסכות רשת

- כדי שאפשר יהיה להבחין בין חלק הרשת וחלק המשתמש בכתובת ה-IP נהוג להצמיד לכל כתובת "מסכה" (Subnet Mask).
- המסכה מאפשרת, באמצעות פעולה לוגית פשוטה, למצוא את כתובת הרשת של כתובת IP מסוימת.
- המסכה, כמו כתובת ה-IP מורכבת מארבע קבוצות של שמונה סיביות. בייצוג בינארי, עבור כל סיבית בכתובת ה-IP שמייצגת את הרשת יופיע 1, ועבור כל סיבית בכתובת ה-IP שמייצגת את המחשב המסוים יופיע 0.
- למשל - בכתובת 192.168.0.17 החלק 192.168.0 מייצג את הרשת והמספר 17 מייצג את המחשב המסוים. מסכת הרשת של כתובת זו תהיה 255.255.255.0, או בייצוג בינארי:

11111111 11111111 11111111 00000000

- כדי למצוא את כתובת הרשת מתוך כתובת IP של מחשב ברשת, יש לבצע פעולת AND (וגם או "כפל") בינארית בין כתובת ה-IP למסכת הרשת. בדוגמה הקודמת, התוצאה תיראה כך -

ייצוג עשירי	ייצוג בינארי	
192.168.0.17	11000000 10101000 00000000 00010001	כתובת IP
255.255.255.0	11111111 11111111 11111111 00000000	מסכת רשת
192.168.0.0	11000000 10101000 00000000 00000000	כתובת הרשת

חלוקה ארחה הכתובות לאחלקות

- **מחלקה A:** כוללת את כל הכתובות שבהן המספר הראשון הוא בין 1 ל-126. הרשתות במחלקה זו מוגדרות כך שהרשת מיוצגת על ידי המספר הראשון (שמונה הסיביות הראשונות), והמחשב מיוצג על ידי שלושת המספרים הנותרים (24 הסיביות הנותרות). קיימות רק 127 רשתות שונות במחלקה A, אך כל אחת מהן יכולה להכיל 2 בחזקת 24 (מעל 16 מיליון) מחשבים. מכיוון שכך, רשתות ממחלקה A מוקצות למדינות או לספקיות אינטרנט גדולות במיוחד.
- **מחלקה B:** כוללת את כל הכתובות שבהן המספר הראשון הוא בין 128 ל-191. הרשתות בה מוגדרות כך שהרשת מיוצגת על ידי שני המספרים הראשונים (16 הסיביות הראשונות), והמחשב מיוצג על ידי 2 המספרים האחרונים (16 הסיביות האחרונות). קיימות 16,384 רשתות שונות במחלקה B, וכל אחת מהן יכולה להכיל עד 65,534 מחשבים. רשתות ממחלקה B מוקצות לחברות גדולות ובינוניות.
- **מחלקה C:** כוללת את כל הכתובות שבהן המספר הראשון הוא בין 192 ל-223. הרשתות בה מוגדרות כך שהרשת מיוצגת על ידי שלושת המספרים הראשונים (24 הסיביות הראשונות), והמחשב מיוצג על ידי המספר האחרון (8 הסיביות האחרונות). קיימות 2,097,152 רשתות שונות במחלקה C וכל אחת מהן יכולה להכיל עד 254 מחשבים. רשתות ממחלקה C מוקצות לחברות וארגונים קטנים בדרך כלל.
- **מחלקה D:** כוללת את כל הכתובות שבהן המספר הראשון הוא בין 224 ל-239. מחלקה D הוגדרת לצורכי מחקר וניסוי של שידור נתונים לקבוצות. כיום משמשת לתקשורת - Multicast שידור ממחשב בודד למספר מחשבים בו זמנית. השדר יוצא פעם אחת ממחשב המקור ומגיע לכל מחשבי היעד. בשיטה זו לא לכל מחשב יש כתובת, אלא לכל שירות התומך ב-Multicast, ולכל שירות יש כתובת זהה במחלקה D כלומר כל המחשבים מאזינים לכתובת המסוימת של השירות, בנוסף לכתובת ה-IP הרגילה שלהם. שימוש לדוגמה ניתן לראות בשרתי WINS של מיקרוסופט Windows 2000. משמש בעיקר לאפליקציות של שיחות ועידה בווידאו.
- **מחלקה E:** כוללת את כל הכתובות שבהן המספר הראשון הוא בין 240 ל-255. מחלקה E הוגדרה כ"שמורה לשימוש עתידי" ועד היום אין שימוש בכתובות שמוגדרות בה.

מחלקה	תחום כתובות	מסכה	מספר כתובות	מספר רשתות
A	0 עד 126	255.0.0.0	16,777,216	127
B	128 עד 191	255.255.0.0	65,536	16,384
C	192 עד 223	255.255.255.0	256	2,097,152
D	224 עד 239	לא מוגדרת	לא מוגדר	לא מוגדר
E	240 עד 255	לא מוגדרת	לא מוגדר	לא מוגדר

CIDR – Classes InterDomain Routing

• נסתכל שוב על המחלקות:

מחלקה	תחום כתובות	מסכה	מספר כתובות	מספר רשתות
A	0 עד 126	255.0.0.0	16,777,216	127
B	128 עד 191	255.255.0.0	65,536	16,384
C	192 עד 223	255.255.255.0	256	2,097,152
D	224 עד 239	לא מוגדרת	לא מוגדר	לא מוגדר
E	240 עד 255	לא מוגדרת	לא מוגדר	לא מוגדר

- נוצר מצב שטווח כתובות ממחלקה B גדול מדי וגורם לבזבוז של כתובות אבל טווח כתובות C קטן מדי ולכן נדרש פתרון טוב יותר.
- CIDR מאפשר הקצאת טווח משתנה של כתובות – כלומר ללא התחשבות במבנה המחלקות. למשל, כאשר ארגון מסוים זקוק ל- 2000 כתובות, ניתן יהיה להקצות להם בלוק של 2048 כתובות.
- כעת, לכל כתובת יצורף Mask בגודל 32 ביט וטבלאות הניתוב יכילו כניסות מהצורה הבאה – (IP address, Mask, Line).
- כאשר מנה מגיעה אל הנתב, הנתב יחפש את הכניסה המתאימה ע"י ביצוע AND עם כל אחת מה- Masks ובדיקה האם התוצאה שווה ל- IP address שבכניסה.
- אם יש יותר מהתאמה אחת, נבחרת ההתאמה עם ה- Mask הגדול ביותר מבחינת מספר ה- 1-ים ב- Mask.

CIDR – Classes InterDomain Routing

• למשל –

University	First address	Last address	How many	Written as
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

Address	Mask
C: 11000010 00011000 00000000 00000000	11111111 11111111 11111000 00000000
E: 11000010 00011000 00001000 00000000	11111111 11111111 11111100 00000000
O: 11000010 00011000 00010000 00000000	11111111 11111111 11110000 00000000

• כעת, אם מגיעה מנה עם כתובת יעד 194.24.17.4 , מבוצע AND עם ה-Mask של Cambridge ומקבלים : 11000010 00011000 00010000 00000000
כלומר – לא מתאים.

CIDR – Classes InterDomain Routing

• המשך –

• כעת מבצעים AND עם ה-Mask של Edinburgh ומקבלים:

11000010 00011000 00010000 00000000

כלומר – שוב לא מתאים.

• בשלב הבא, מבצעים AND עם ה-Mask של Oxford ומקבלים –

11000010 00011000 00010000 00000000

כלומר – מתאים!

• אם אין יותר התאמות בטבלה, שולחים את המנה על הקו שמצוין בכניסה זו.

אק "נקנס" כתובת IP

שאלה: אך מחשב משיג כתובת IP ?

hard-coded by system admin in a file ■

Wintel: control-panel->network-
>configuration->tcp/ip->properties ■

UNIX: /etc/rc.config ■

DHCP: Dynamic Host Configuration Protocol: ■

dynamically get address from as server

"plug-and-play" ■

DHCP: Dynamic Host Configuration Protocol

מטרה:

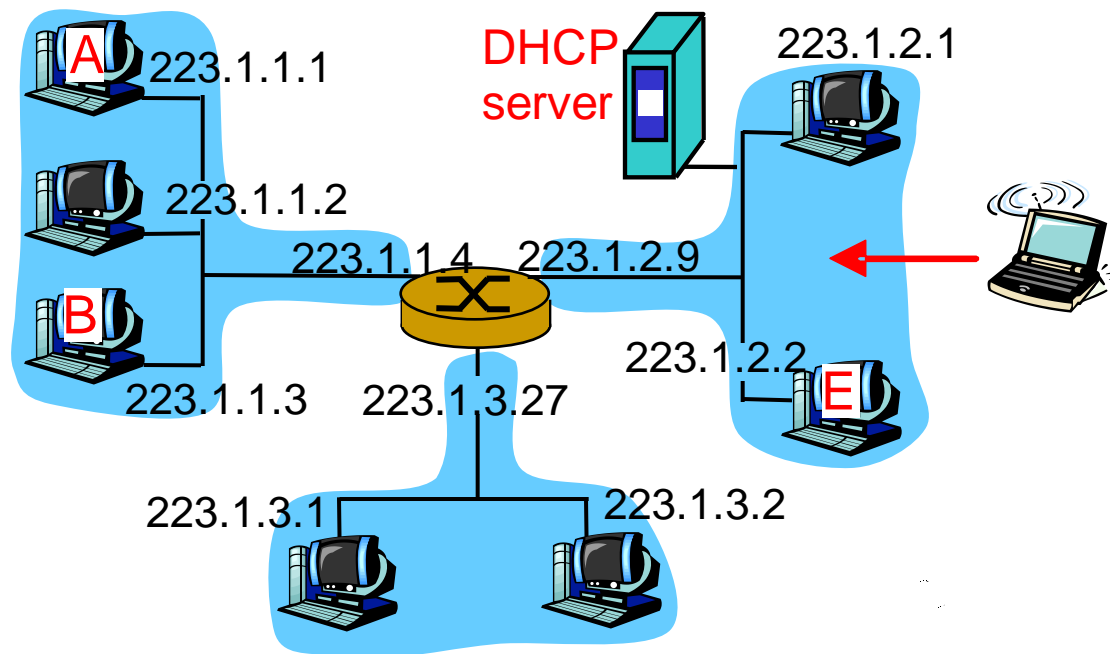
1. להרשות למחשב באופן דינמי להשיג כתובת IP משרת הרשת כאשר הוא מצטרף לרשת.
2. לאפשר לחדש את החכירה של הכתובת תוך כדי שימוש.
3. לאפשר שימוש מחדש בכתובת.
4. לתמוך במשתמשים ניידים שמצטרפים לרשת לזמן קצר.

DHCP overview:

- host broadcasts "DHCP discover" msg
- DHCP server responds with "DHCP offer" msg
- host requests IP address: "DHCP request" msg
- DHCP server sends address: "DHCP ack" msg

DHCP client-server scenario

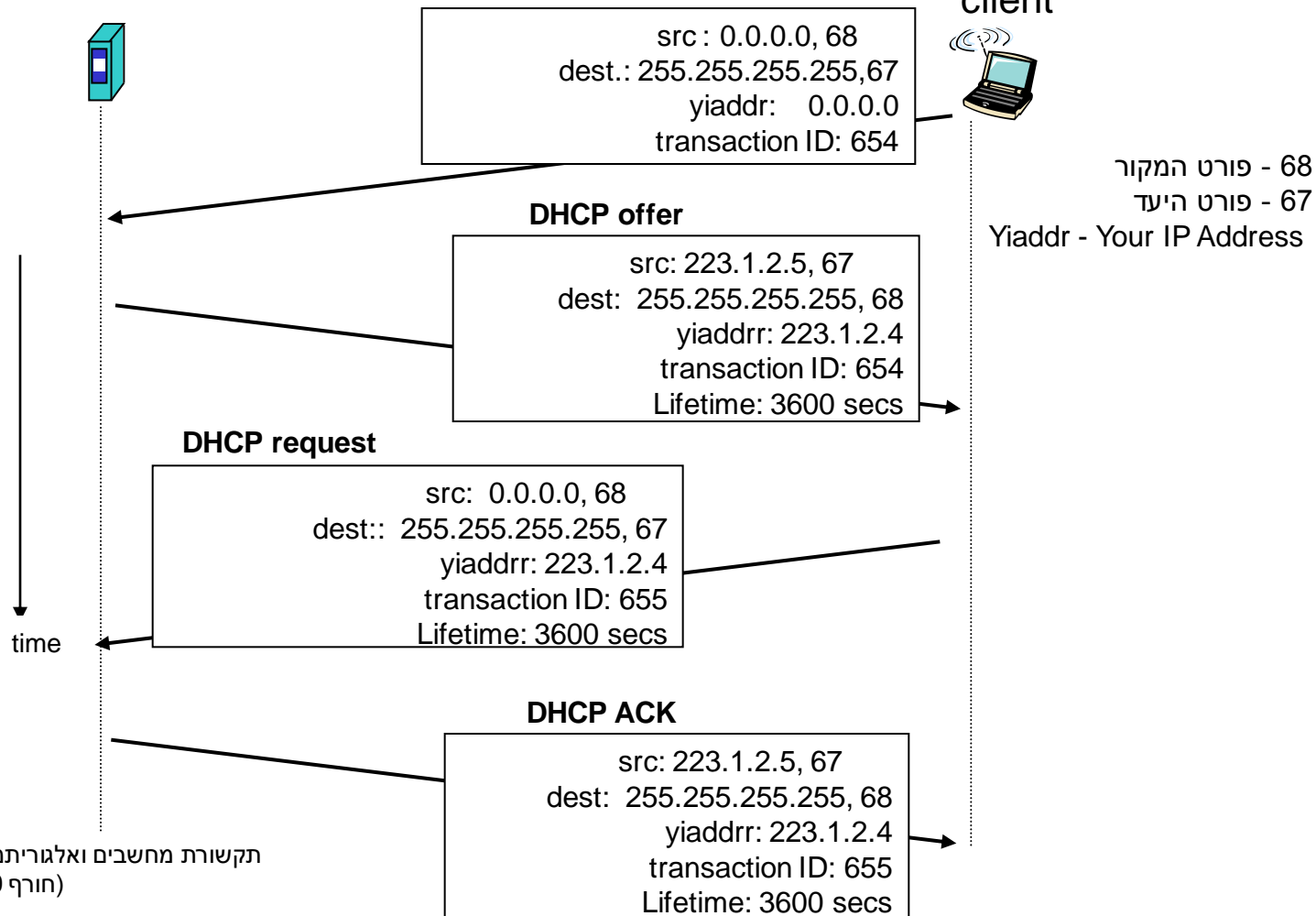
מגיע לקוח DHCP
שצריך כתובת IP ברשת



DHCP client-server scenario

DHCP server: 223.1.2.5

arriving
client



שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

פרוטוקול ICMP

(Internet Control Message Protocol)

- ICMP הינו סוג של פרוטוקול מסוג IP שפועל בשכבת הרשת ואשר מספק הודעות שגיאה ומידע בקרה אחר בחזרה למקור.
- סוגי תקלות: נפילת רשת, נפילת נתב במסלול, ניתוק מחשב היעד מהרשת, עומס באחד הנתבים, בעיות ניתוב.
- אם חבילה הולכת לאיבוד או נזרקת מסיבות שונות – יש צורך במנגנון שיאפשר דיווח על התקלה.
- הודעת ICMP מאפשרת לנתב לדווח על תקלה ב-Datagram חזרה לתחנת המקור שממנה נשלחה החבילה.
- הודעות ICMP מועברות בחלק המידע של ה-IP Datagrams בדומה לכל חבילת מידע בפורמט IP.
- חבילות ICMP יכולות ללכת לאיבוד ברשת – אין מדווחים על שגיאות שנגרמו ע"י חבילות IP הנושאות חבילות ICMP (למעט שאילות).
- ICMP הינו חלק אינטגרלי מפרוטוקול ה-IP ממומש בכל מודל IP.
- יעדן הסופי של הודעות ICMP היא תוכנת ה-IP עצמה בתחנת המקור.
- הפרוטוקול אינו מוגבל רק לשימושם של נתבי IP אלא מאפשר לכל מכונה לשלוח הודעות לכל מכונה אחרת ברשת.
- ICMP הוא מנגנון דיווח של שגיאות. הפרוטוקול יפנה בעיות שאינם יכול לפתור לפרוטוקול העליון או לאפליקציה שיזמה את החבילה, במיוחד אם התקלה נגרמת לאורך המסלול שבו עוברת החבילה.

ICMP: Internet Control Message Protocol

Type	Code	description
------	------	-------------

0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

בשימוש ע"י מחשבים &

נתבים לקבלת/העברת מידע ברמת הרשת.

■ דיווח שגיאות: מחשבים, רשתות, ממשקים ופרוטוקולים שאינם ברי-השגה.

■ בקשה או החזרה של echo (שימוש בפרוטוקול ping).

שכבת הרשת "מעל" ה-IP:

■ הודעת ICMP נישאת בחבילת ה-IP.

■ הודעת ICMP: סוג, קוד פלוס 8 הבייטים הראשונים של חבילת ה-IP שגרמה לשגיאה.

Traceroute and ICMP

Traceroute - תכנית שירות באינטרנט היוצרת תרשים של הנתיב והזמן הדרוש להגעת חבילות נתונים ממחשב אישי למחשב מארח רחוק

- Source sends series (סדרה) of UDP segments to destination.
 - First has TTL (Time To Live) = 1
 - Second has TTL=2, etc.
 - Unlikely (לא סביר) port number
 - When nth datagram arrives to nth router:
 - Router discards (משליך) datagram
 - And sends to source an ICMP message (type 11, code 0)
 - Message includes name of router & IP address
 - When ICMP message arrives, source calculates RTT (round trip time)
 - Traceroute does this 3 times
- Stopping criterion
- UDP segment eventually arrives at destination host
 - Destination returns ICMP "host unreachable" packet (type 3, code 3)
 - When source gets this ICMP, stops.

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

IPv6

□ **מוטיבציה ראשונית:** מרחב הכתובות של 32-bit יגמרו בקרוב ולכן נדרשת הרחבה.

□ **מוטיבציה נוספת:**

■ פורמט הראש (header) עוזר למהירות התהליך / השליחה.

■ פורמט הראש (header) מקדם את איכות השירות (QoS).

□ **פורמט חבילות מידע IPv6:**

■ אורך קבוע של הראש (header) - 40 byte

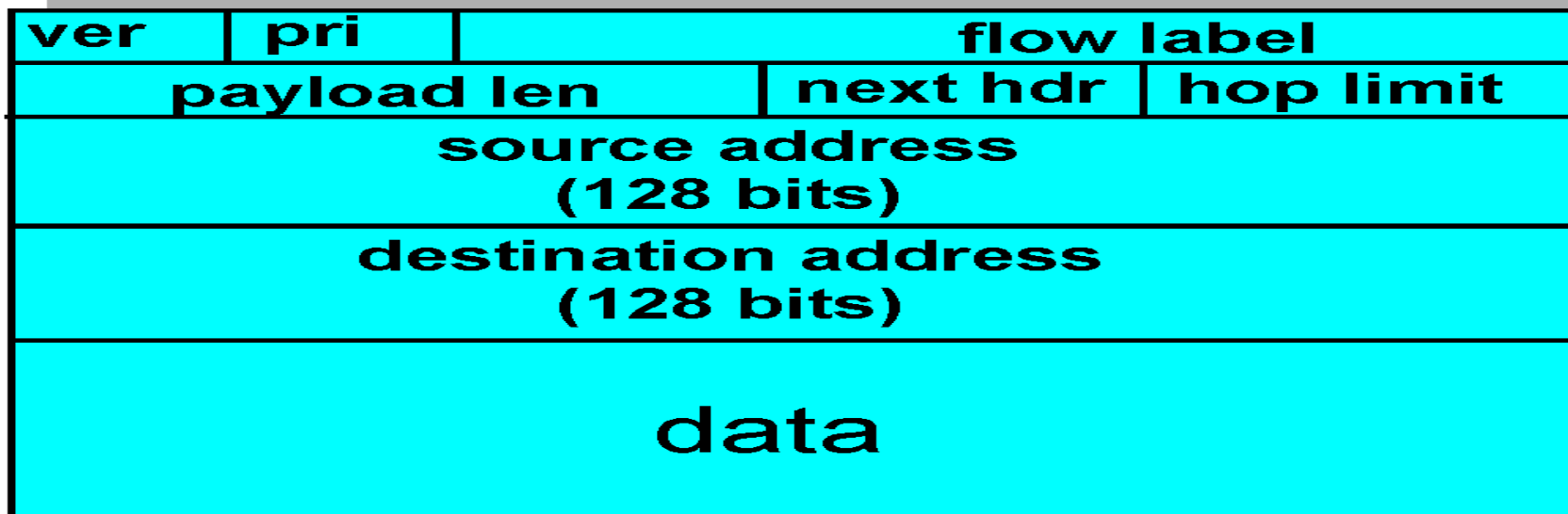
■ שבירת חבילות אינה מורשת.

□ **ב- IPv6 נעשה שימוש ב-128 סיביות, המאפשרות עד 3.4×10^{38} כתובות שונות (2^{128}).**

כתובות IPv6

- בניגוד לכתובת IPv4, שנכתבת בצורה של 4 מספרים עשרוניים המופרדים בנקודות, כתובת IPv6 מיוצגת ע"י מספרים הקסאדצימליים (בבסיס 16 ולא בבסיס 10), והמספרים מופרדים זה מזה ע"י נקודותיים (:). דוגמא לכתובת כזו היא **fe80::2a0:d2ff:fea5:e9f5** כאשר כל קבוצה של 16 ביטים מיוצגת ע"י 4 ספרות הקסאדצימליות. כאשר יש קבוצה של 16 ביטים שמכילה אפסים בלבד, ניתן לייצג אותה ע"י "::". כך, הכתובת שראינו קודם יכולה להיות מיוצגת גם באופן מלא, כך: **fe80::2a0:d2ff:fea5:e9f5** (HEX(16) : 0...9ABCDEF, A=10, B=11,..., F=15).
- IPv6 משתמש בשיטה הנוחה הזו לציון מסיכת הרשת: **a01:2::/64:2001:638** מסכה זו מציינת שאורך מזהה הרשת, בכתובת ה- IP הזו, הוא 64 ביטים.
- בדרך כלל, כתובות IPv6 מכילות 48 ביטים של מזהה הרשת כולה, 16 ביטים של מזהה תת-רשת (ובסה"כ 64 ביטים של מזהה רשת), דבר המשאיר 64 ביטים לזיהוי המחשב ברשת. זה יכול להיראות כבזבז - קשה לחשוב על רשת שתצטרך להשתמש בכל מרחב מזהי המחשב הזה (כמה מיליארדי מיליארדים!). הסיבה לכך היא, שב- IPv6 מומלץ לבנות את מזהה המחשב לפי כתובת ה-MAC של המכשיר (הכתובת הצרובה בכרטיס הרשת שלו). תבנית כתובות זו מכונה "כתובות EUI64".
- לדוגמא, במקרה שבו כרטיס הרשת בו משתמש המכשיר הוא אתרנט, וכתובת ה-MAC שלו היא **ab:01:23:45:67:89** מזהה המחשב בכתובת ה- IP שלו יהיה **0123:45ff:fe67:89ab** כלומר, כתובת ה-MAC, שבנויה מ- 48 ביטים, ועוד 16 ביטים באמצע. השימוש בכתובות ה-MAC כבסיס לכתובות האינטרנט, מאפשר השמה אוטומטית של מזהה המחשב בכתובות IP לכל מכשיר, ויכול להקל מאוד על מנהלי רשתות.

IPv6 Header (Cont)



← 32 bits →

Priority – עדיפות: מטרתו לזהות עדיפות של חבילות בתוך זרם הנתונים, שדה זה מאפשר להבחין בין תעבורה הרגישה במיוחד לעיכובים, כמו וידאו וקול, לבין תעבורה בעדיפות נמוכה יותר, כמו דואל ותעבורת web.

Flow Label – תוית זרימה: המחשב השולח יכול להשתמש בשדה זה כדי לסמן חבילות השייכות לאותו "זרם", כלומר לאותו קישור בין מחשבי קצה. כך ניתן להגדיר טיפול מיוחד לזרמים שונים, ולספק דרישות של איכות השירות (QoS).

Payload len – אורך החבילה: שדה זה מציין את אורך הנתונים של חבילת ה-IP. בניגוד לשדה המקביל ב-IPv4, שדה זה אינו כולל את אורך הפתיח.

Next header – הראש הבא: זיהוי פרוטוקול השכבה הגבוה יותר עבור נתונים.

IPv6 Header (Cont)

■ **non-hop limit הקפיצות:** שדה זה מגדיר את המספר המקסימלי של נתבים שונים (כלומר מספר הקפיצות) שחבילה יכולה לעבור בדרך ליעדה, לפני שהיא מושמדת. שדה זה מקביל לשדה ה-TTL ב-IPv4. שדה ה-TTL הוגדר בתחילה כמספר השניות שנותרו לחבילה לחיות, ולא כמספר הקפיצות שמותר לו עוד לעבור, אך בפועל השימוש בשדה ה-TTL התבצע לפי קפיצות ולא לפי זמן.

■ **סיווי נוספי IPv4-**

■ **Checksum:** הסרת תכונה זו לחלוטין כדי שנוכל להוריד את זמן התהליך בכל hop ("ניתור" נקודת חיבור בין מחשבים ברשת).

■ **Options:** לאפשר, אבל מחוץ ל-header להעיד על שדה "Next Header".

■ **ICMPv6:** גירסה חדשה של ICMP.

■ הודעה נוספת מסוג "Packet Too Big"

■ multicast group management functions

מעבר מ-IPv4 ל-IPv6

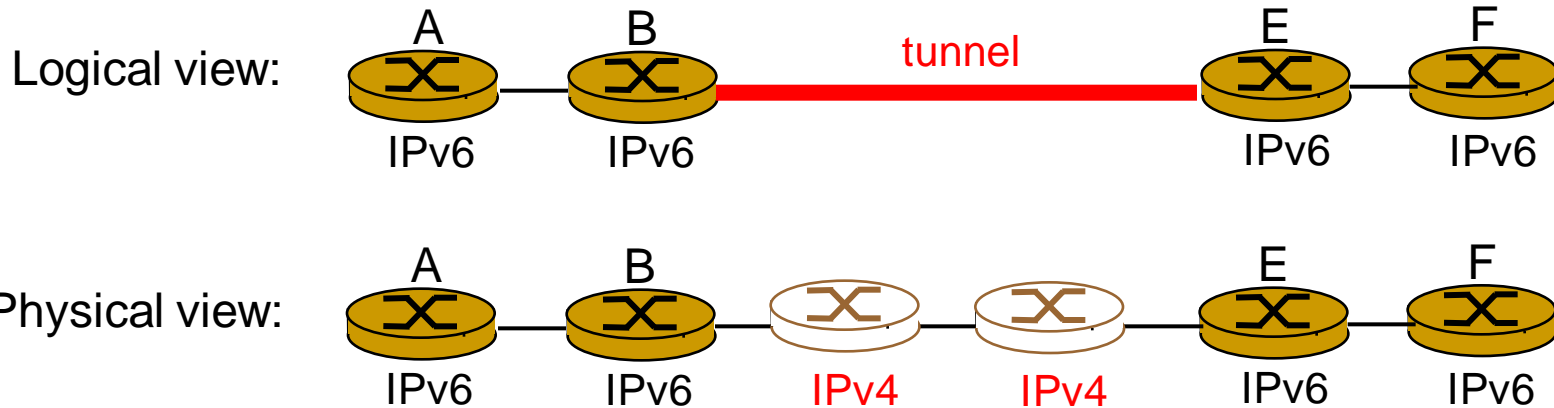
□ לא כל הנתבים יכולים להתעדכן בזמן.

■ אין "יום הדגל"

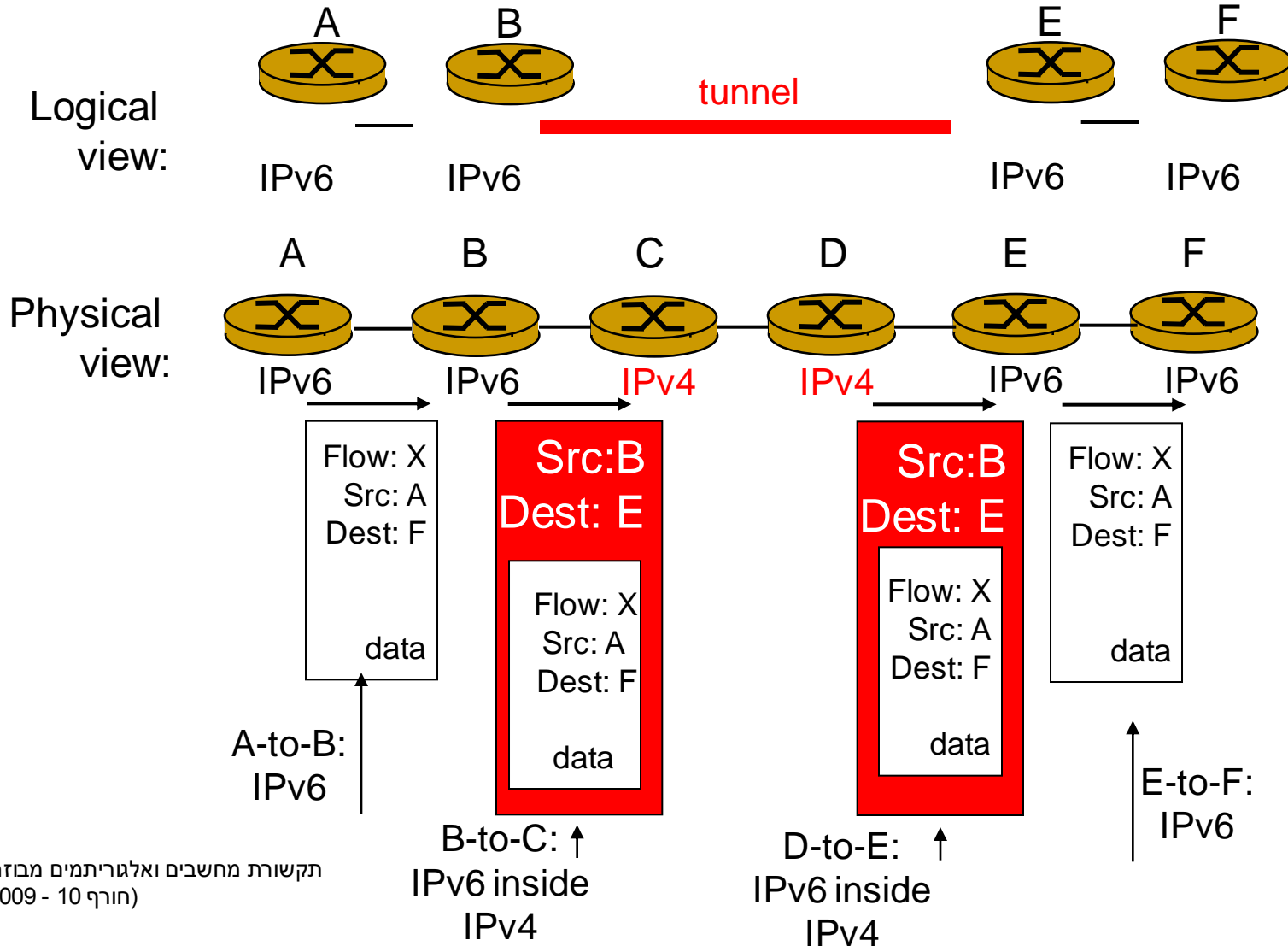
■ לכן נשאלת השאלה אך הרשת תפעל עם עירבוב של נתבי IPv4 ו-IPv6 ?

□ **מינהור:** IPv6 ינשא (מלשון לשאת) כמטען בחבילות המידע של IPv4 בין נתבי IPv4.

Tunneling - ט'נהול



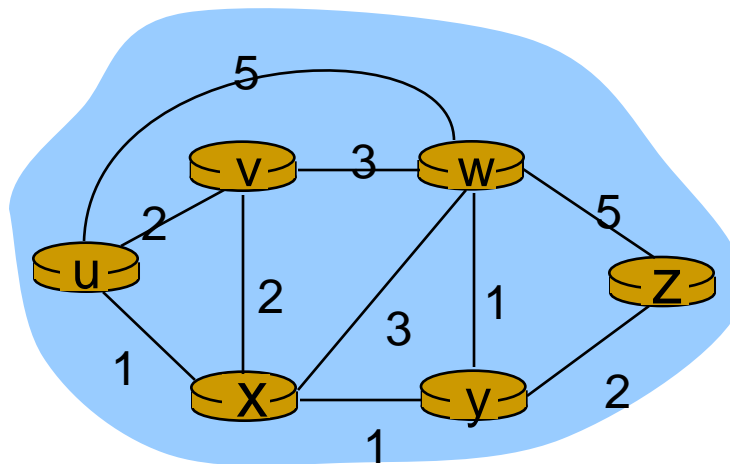
Tunneling טונל



שכבת הרשת

- **מבוא**
- **מעגלים וירטואלים וחבילות המידע ברשת**
- **מה יש בתוך הנתב (router)**
- **פרוטוקלי אינטרנט**
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6
- **אלגוריתמי ניתוב:**
 - Link state
 - Distance Vector
 - Hierarchical routing
- **ניתוב באינטרנט:**
 - RIP
 - OSPF
 - BGP
- **ניתוב Broadcast ו-multicast**

הפשטה של תורת הגרפים



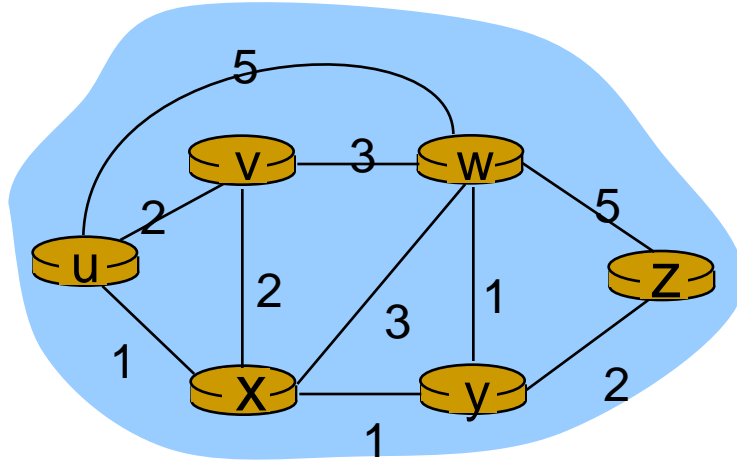
Graph: $G = (N, E)$

N = set of routers = $\{ u, v, w, x, y, z \}$

E = set of links = $\{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

תזכורת: הפשטה של תורת גרפים שימושית בהקשר של רשת תקשורת

הכנסה על תורת הארפים: מחיר



Cost of (path (x, x')) = $c(x, x')$ מחיר הנתב)

• לדוגמא: $5 = c(w, z)$

• מחיר רצוי שיהיה תמיד 1, או ביחס הפוך לרוחב הפס או לעומס.

מחיר הנתב $(x_1, x_2, x_3, \dots, x_p)$ = $c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$

שאלה: מהו המחיר הנמוך ביותר בין u ל-z ?

אלגוריתם ניתוב: אלגוריתם שמוצא את הנתב במחיר הנמוך ביותר.

סיווג אלגוריתמי ניתוח

Routing Algorithm classification

מידע כללי (גלובלי) או ביזורי
(נקודתי)?

גלובלי:

□ לכל הנתבים יש את הטופולוגיה שלמה,
ומידע על עלות הערוץ.

□ אלגוריתם "link state".

ביזור:

□ הנתב יודע את השכנים המחוברים
אליו פיסית, ואת מחיר הערוץ לכל שכן.
□ תהליך איטרטיבי של חישוב, והחלפת
מידע עם השכנים.

□ אלגוריתם "distance vector".

סטטי או דינמי ?

סטטי:

□ הניתוב משתנה באיטיות לאורך
הזמן.

דינמי:

□ הניתוב משתנה יותר מהר:

- עידכון תקופתי.
- בתגובה לשינויים במחיר
הערוץ.

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast

אלגוריתם ניתוח "מצב הערוץ" (Link-State)

האלגוריתם של Dijkstra

- טופולוגית הרשת, מחיר הערוצים ידועים לכל הקודקודים.
 - לכל הקודקודים אותו מידע.
- מחשבים את מחיר הנתיב הנמוך ביותר מכל קודקוד לכל קודקודים האחרים.
- נתונה טבלת השליחות (forwarding table) של הקודקודים.
- איטראטיבי: אחרי k חזרות, ידועים מחירי הנתיבים ל- k יעדים.

האלגוריתם Dijkstra של חיפוש הנתיב הקצר ביותר

1 Initialization:

2 $N = \{A\}$

3 for all nodes v

4 if v adjacent to A

5 then $D(v) = c(A,v)$

6 else $D(v) = \text{infinity}$

7 end

8 Loop

9 find w not in N such that $D(w)$ is a minimum

10 add w to N

11 update $D(v)$ for all v adjacent to w and not in N :

12 $D(v) = \min(D(v), D(w) + c(w,v))$

13 /* new cost to v is either old cost to v or known

14 shortest path cost to w plus cost from w to v */

15 **until all nodes in N**

❖ Each router calculates lowest cost path to all its neighbors, starting from itself;

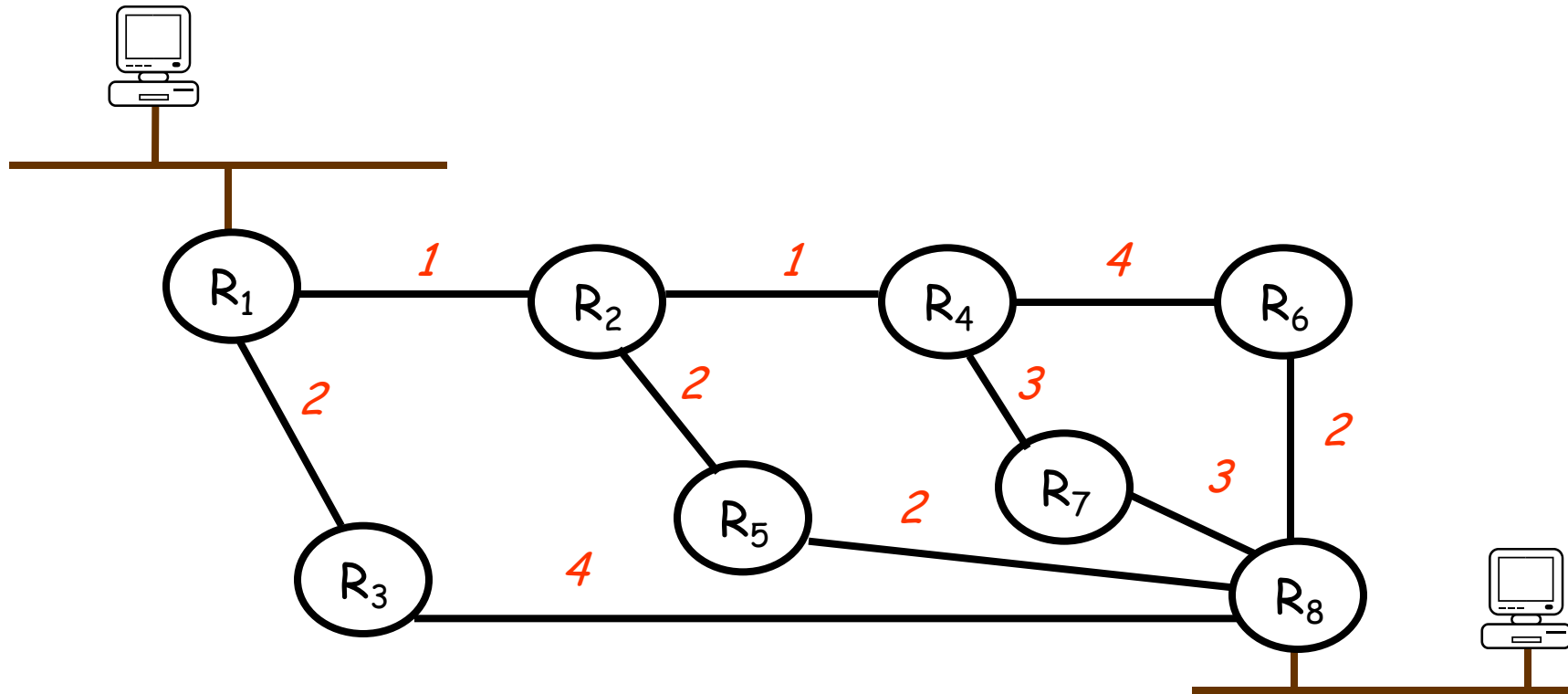
❖ At each step of the algorithm, a router

- adds a node to N to which it can reach with its current minimum cost
- For every node outside N , updates the minimum path cost to it

הסבר על פאלאטת האלגוריתם Dijkstra

- בשלב הראשון אנחנו מגדירים את קבוצת הקודקודים שהמרחק אליהם ידוע מקודקוד ההתחלה N להיות רק קודקוד A .
- נותנים מחירי מעבר לכל הקודקודים שמקושרים ל- A לפי מחיר הקשת הישירה מ- A אליהם ולכל שאר הקודקודים מחיר אינסופי. כאשר אנחנו יודעים שהמחיר של הקשת הישירה מ- A לאותם קודקודים הוא המחיר המינימאלי כדי להגיע אליהם לפי אי שוויון המשולש (המחיר של קשת ישירה שמחברת 2 קודקודים זול יותר מסכום מחירי הקשתות שיוצרות כל מסלול עקיף בניהן).
- בתוך הלולאה אנחנו מוסיפים בכל איטרציה את הקודקוד שמחיר הקשת שמגיעה אליו מתך קבוצת הקודקודים הידועים N הוא הנמוך ביותר.
- מכיוון שההתייחסות שלנו לקבוצה N היא כמו לקודקוד אחד, אנחנו יכולים לנסות למזער מחירי קשתות בכל פעם שאנחנו מוסיפים קודקוד חדש, כי יכול להיות שקיבלנו עכשיו קשת חדשה וזולה לקודקוד מסוים. הלולאה תמשיך לרוץ עד שכל הקודקודים יכנסו ל- N , כלומר, עד שכל הגרף ימופה.

דואל אפאפאוריתם Dijkstra פפצ'אט הנתיה הקצר ביותר



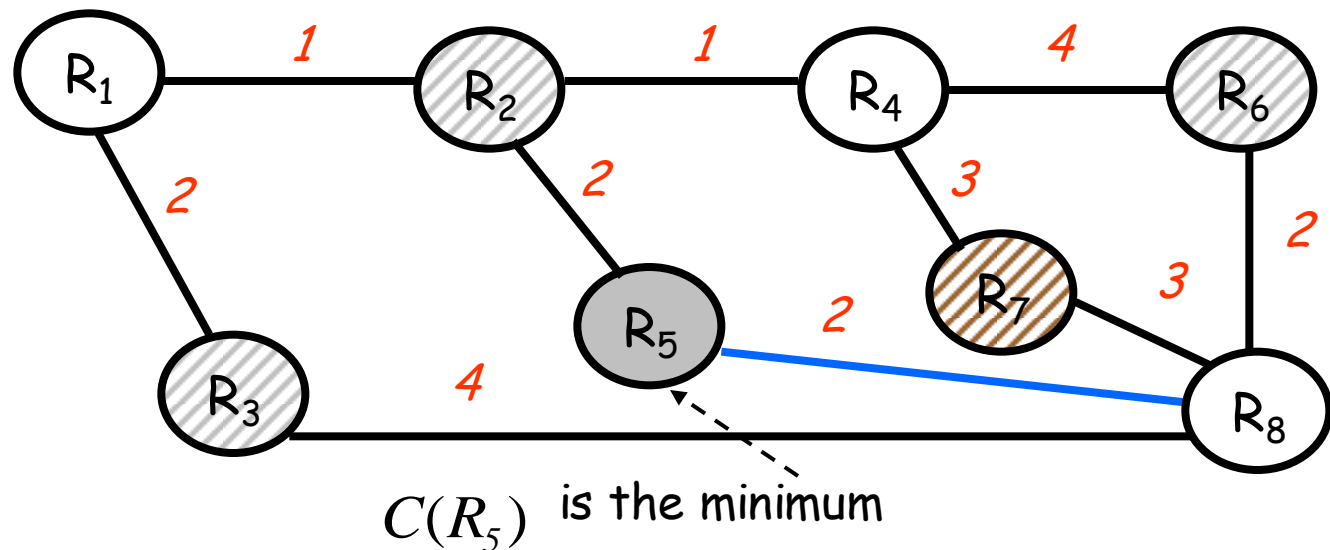
דונא אפאדאוריתם Dijkstra פאצ'אט הנתיב הקצר ביותר

Step 1. Shortest path set : $M = \{R_8\}$

Candidate set (finite cost) : $C = \{R_3, R_5, R_7, R_6\}$

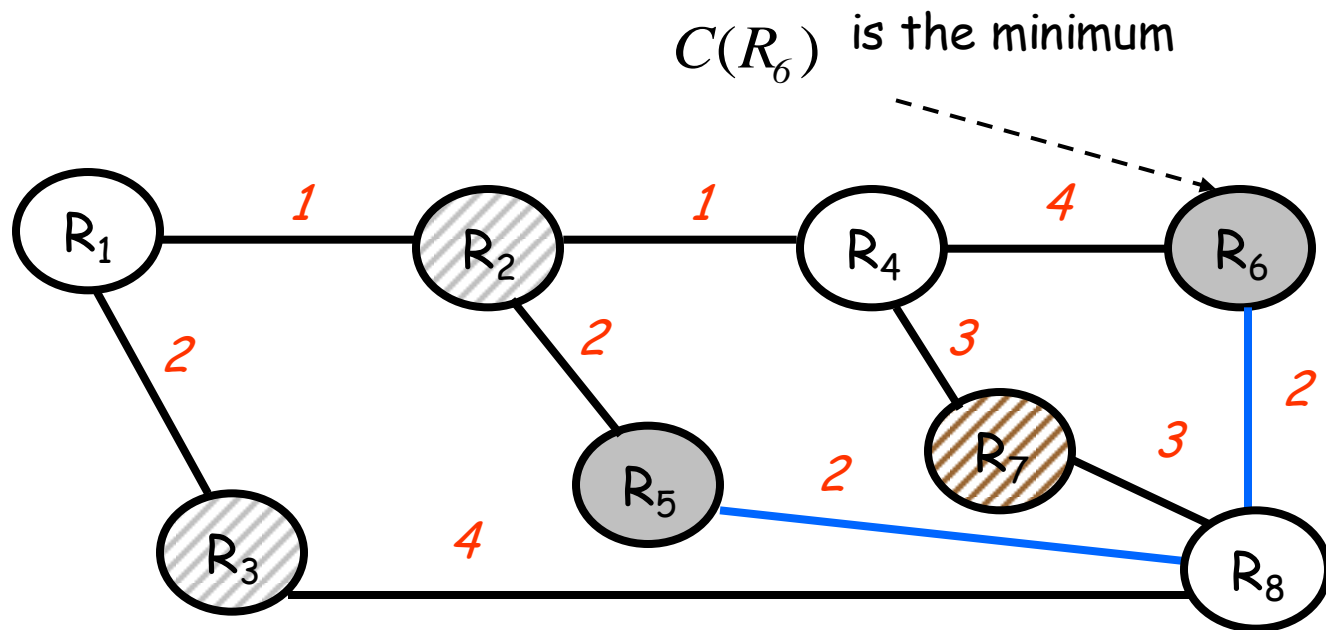
Step 2 : $M = \{R_8, R_5\}$,

$C = \{R_3, R_7, R_6, R_2\}$.



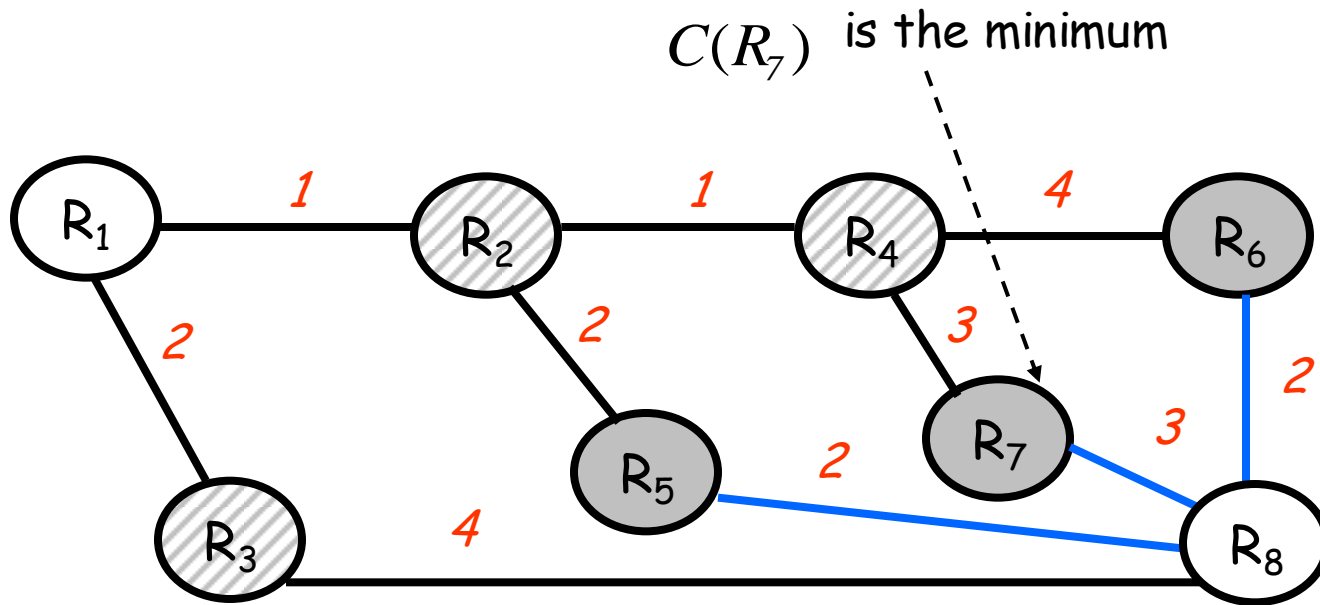
דונאטא פאפאדאריטס Dijkstra פֿע מציאת
הנתיב הקצר ביותר

Step 3: $S = \{R_8, R_5, R_6\}$,

$$C = \{R_3, R_7, R_2, R_4\}.$$


דואלס אלכאולריתס Dijkstra אלצ'אט הנתיה הקצר ביותר

Step 4 : $S = \{R_8, R_5, R_6, R_7\}$,
 $C = \{R_3, R_2, R_4\}$.

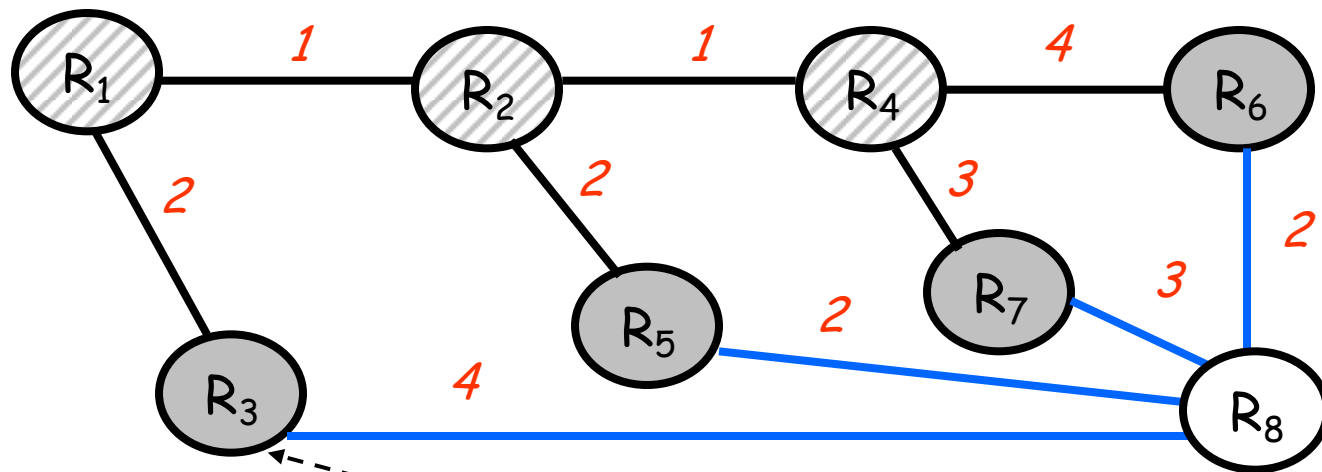


דואל אלגוריתם Dijkstra של זמן

הנתיב הקצר ביותר

Step 5 : $S = \{R_8, R_5, R_6, R_7, R_3\}$,

$C = \{R_2, R_4, R_1\}$.

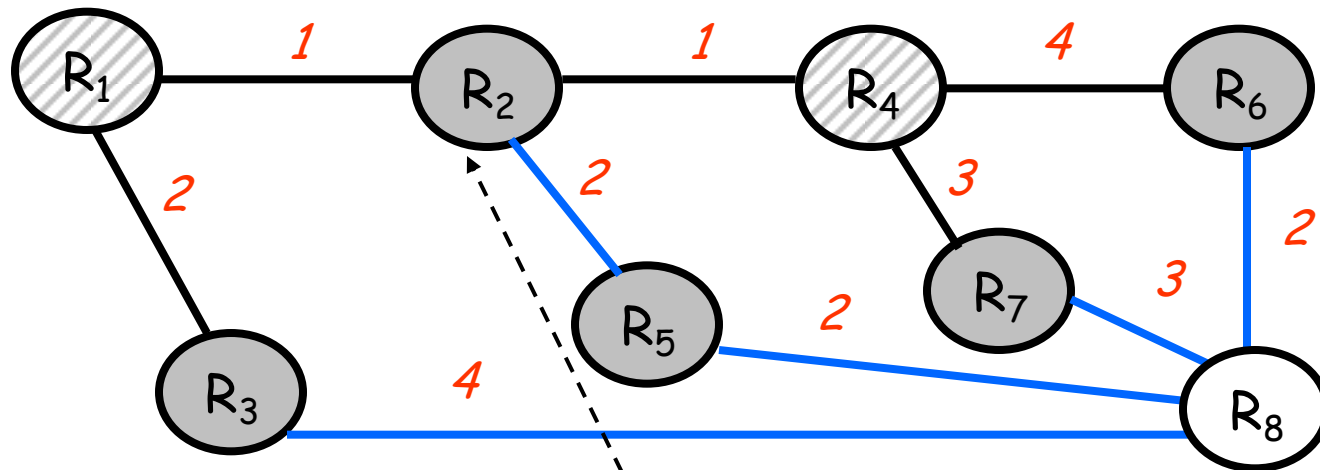


$C(R_3)$ is the minimum

דואלס אלגוריתם Dijkstra מציג הנתיב הקצר ביותר

Step 6: $S = \{R_8, R_5, R_6, R_7, R_3, R_2\}$,

$C = \{R_4, R_1\}$.

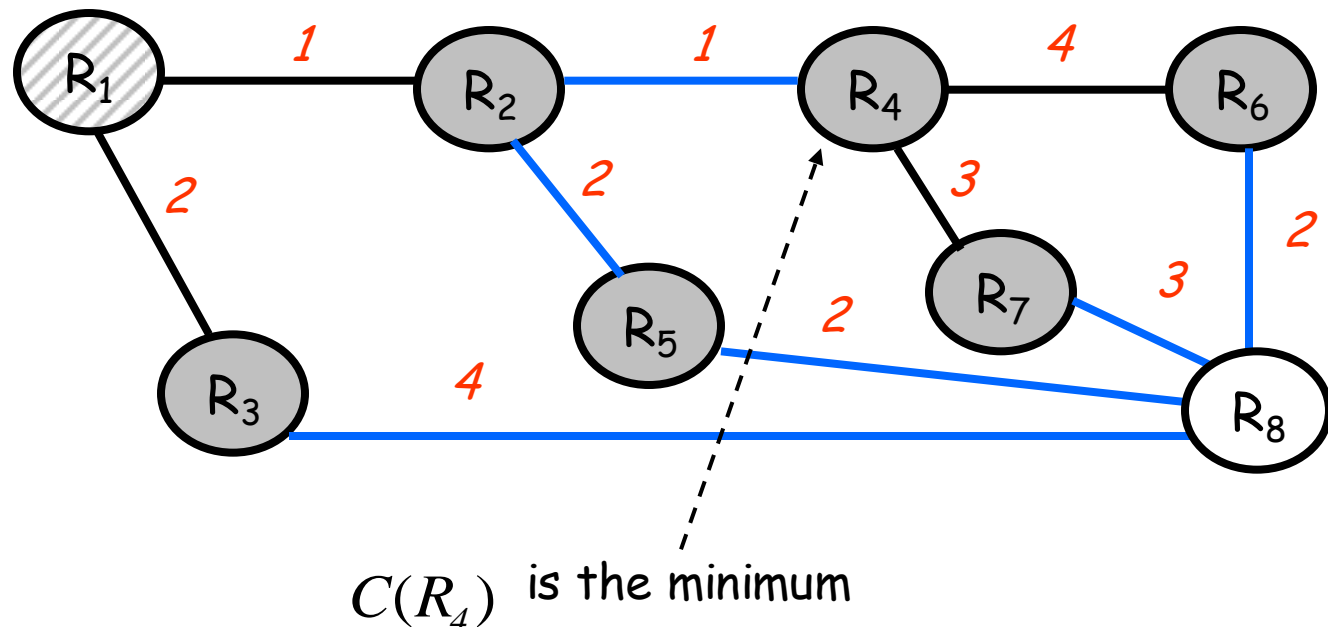


$C(R_2)$ is the minimum

דיון אלגוריתם Dijkstra הנתיב הקצר ביותר

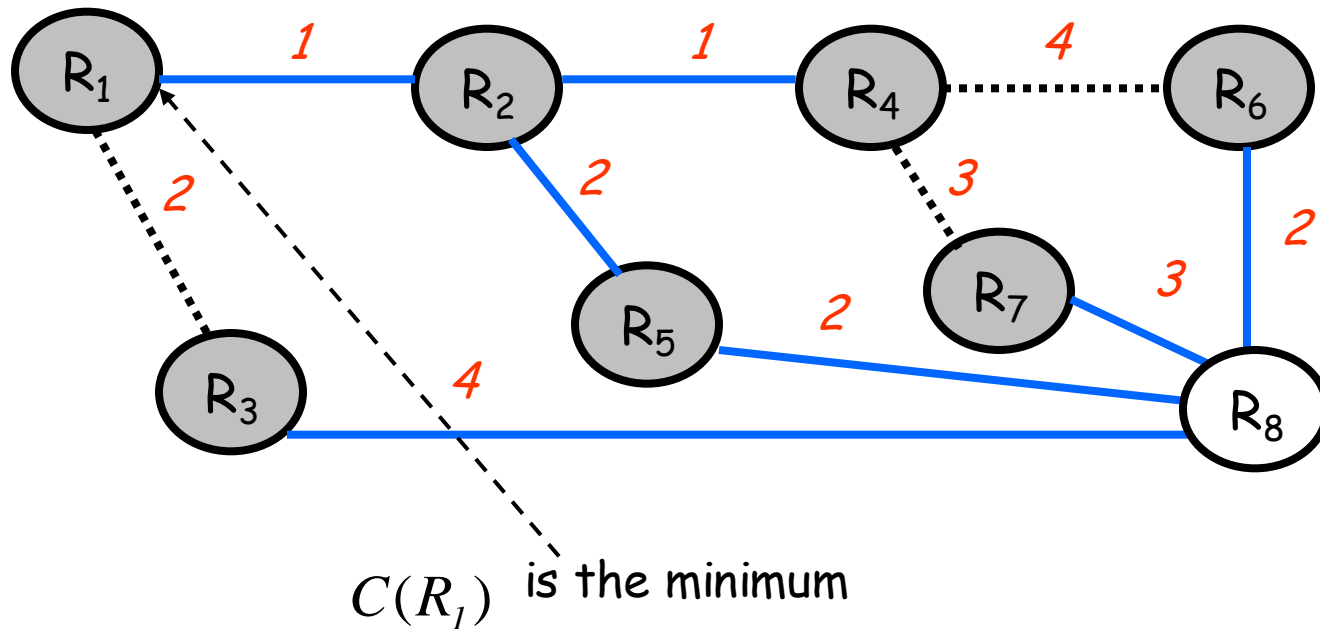
Step 7 : $S = \{R_8, R_5, R_6, R_7, R_3, R_2, R_4\}$,

$C = \{R_1\}$.



דואלס אלגוריתם Dijkstra מציאת הנתיב הקצר ביותר

Step 7 : $S = \{R_8, R_5, R_6, R_7, R_3, R_2, R_4, R_1\}$



אלגוריתם Dijkstra – דיון

סיבוכיות האלגוריתם: n קודקודים

□ בכל איטרציה: נדרש לבדוק את כל הקודקודים, C , שאינן ב- S .

□ לכן זמן ריצה של האלגוריתם (**ללא הוכחה**): $O(n \log n)$

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

Distance Vector *וקטור המרחק*

- מחזורי, שני דרכים להחליף מידע בין שכנים.
- חוזר על עצמו (איטרטיבי), לא בו-זמני (לא סינכרוני) ומבוזר.
- בזמן ההחלפה, ה-switch שולח:
 - רשימה של זוגות.
 - כל זוג כולל יעד ומרחק ליעד
- הקולט (Receiver):
 - משווה כל פריט ברשימה לנתיב המקומי.
 - משנה את הנתיב אם נתיב טוב יותר קיים.

אינטואיציה וקטור במרחקים

Distance Vector Intuition

□ יהי:

- N מספר השכנים אשר שולחים הודעות ניתוב
- V היעד בזוג (זוג כולל יעד ומרחק ליעד).
- D המרחק ליעד בזוג (זוג כולל יעד ומרחק ליעד)
- $C = D + W_N$ כאשר W_N הוא המחיר הנדרש להשיג את השולח.

□ אם אין נתיב מקומי ליעד V או לנתיב המקומי יש מחיר הגדול מ- C

- נקבע נתיב עם נקודת חיבור (hop) הבאה N , במחיר C .
- אחרת התעלם מהזוג.

אלגוריתמי וקטור מרחקים (distance vector):

Initialization

```
2  for all adjacent nodes v:
3      D (*,v) = infinity      /* the * operator means "for all rows" */
4      D (v,v) = c(X,v)
5  for all destinations, y
6      send minw Dx (y,w) to each neighbor /* w over all X's neighbors */
```

loop

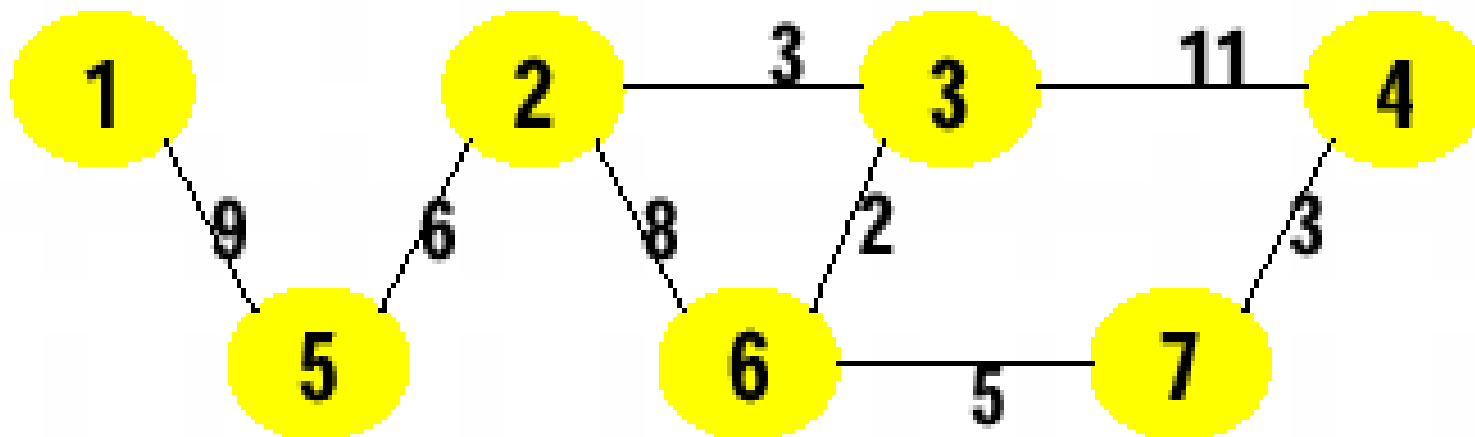
```
7  wait (until I see a link cost change to neighbor V
8      or until I receive update from neighbor V)
9  if (c(X,V) changes by d)
10     for all destinations y: D (y,V) = D (y,V) + d
11 else if (update received from V wrt destination Y)
12     for the single destination y: D (Y,V) = c(X,V) + newval
13 if we have a new min D(Y,w) for any destination Y
14     send new value of minD(Y,w) to all neighbors
```

forever

הסבר על פצולת האלגוריתם

- אלגוריתם זה ירוץ על כל צומת בגרף עם כניסתו לגרף. תחילה כל המחירים יאתחלו לאינסוף (שורה 3), ולאחר מכן מחירי הקשתות לשכנים יאתחלו כמחירי המינימום לצמתים אלה (שורה 4), ואז המידע הזה יופץ לכל שכניו של אותו קודקוד (שורות 5,6).
- קטע הקוד הבא (שורות 7,14) ירוץ על כל קודקוד בגרף כל הזמן, תפקידו לנהל עדכונים. בכל פעם שיש אירוע של שינוי בקשת לשכן V (שינוי בקשת שמחוברת ישירות לקודקוד או שינוי ששכן הודיע עליו), נסכם את השינוי יחד עם המחיר המקורי שהיה לנו (אם השינוי היה שלילי אז המחיר ירד), ואם המחיר האופטימלי לקודקוד V אכן השתנה נשלח את העדכון – המחיר החדש לשכנים (שיבצעו את אותה שגרה).

דואנא fe ניתוב וקטור מרחקים



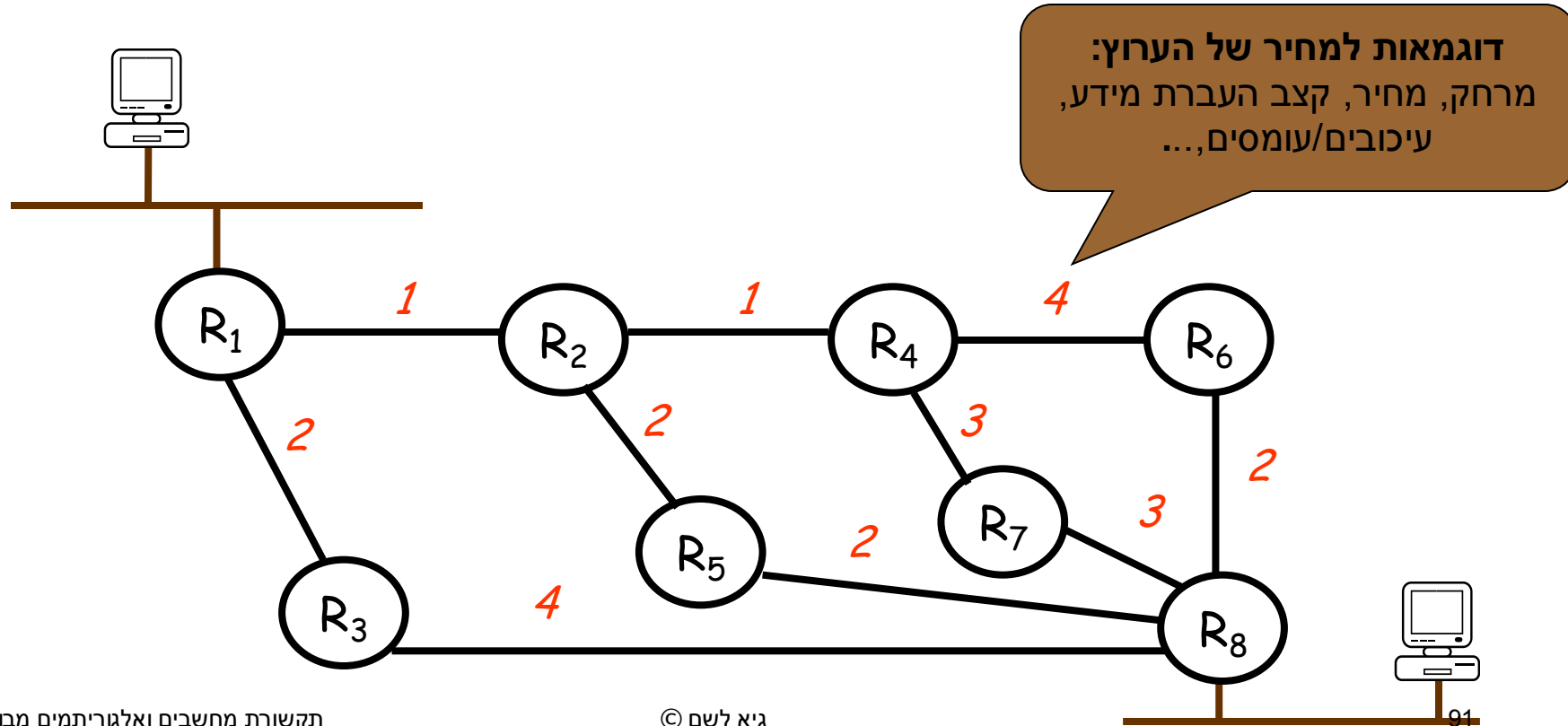
□ נסתכל על העברה של הודעת ניתוב וקטור מרחקים (DV) אחת:

1. צומת מספר 2 שולח לצמתים 3, 5, ו-6.
2. צומת 6 קובע מחיר 8 בניתוב לצומת 2.
3. יותר מאוחר צומת 3 שולח עידכון לצומת 6.
4. צומת 6 משנה ניתוב כדי לעשות את צומת 3 נקודת החיבור הבאה עבור היעד 2.

אלגוריתם Bellman-Ford

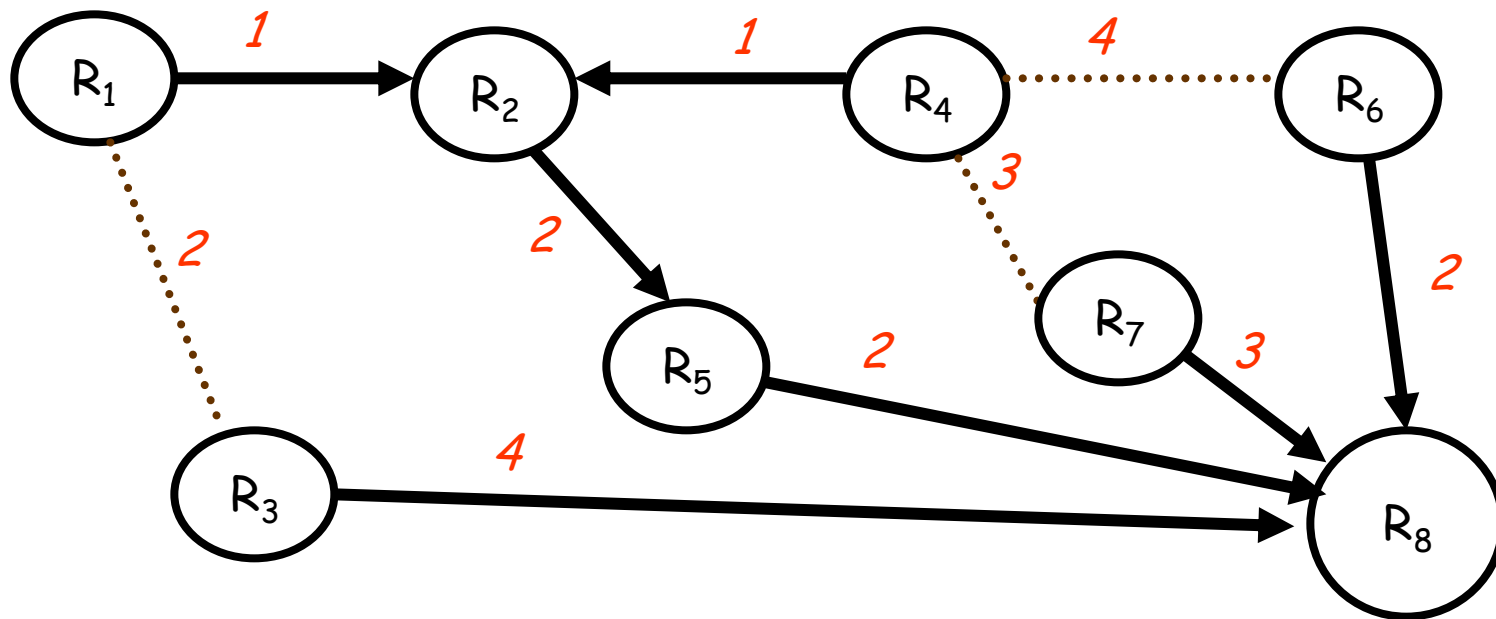
מטרה: למצוא נתיב עם מחיר מינימאלי מכל צומת לכל צומת אחרת בגרף.

לדוגמא: לקבוע נתיבים מ- (R_1, \dots, R_7) ל- R_8 אשר ממזערים את המחיר



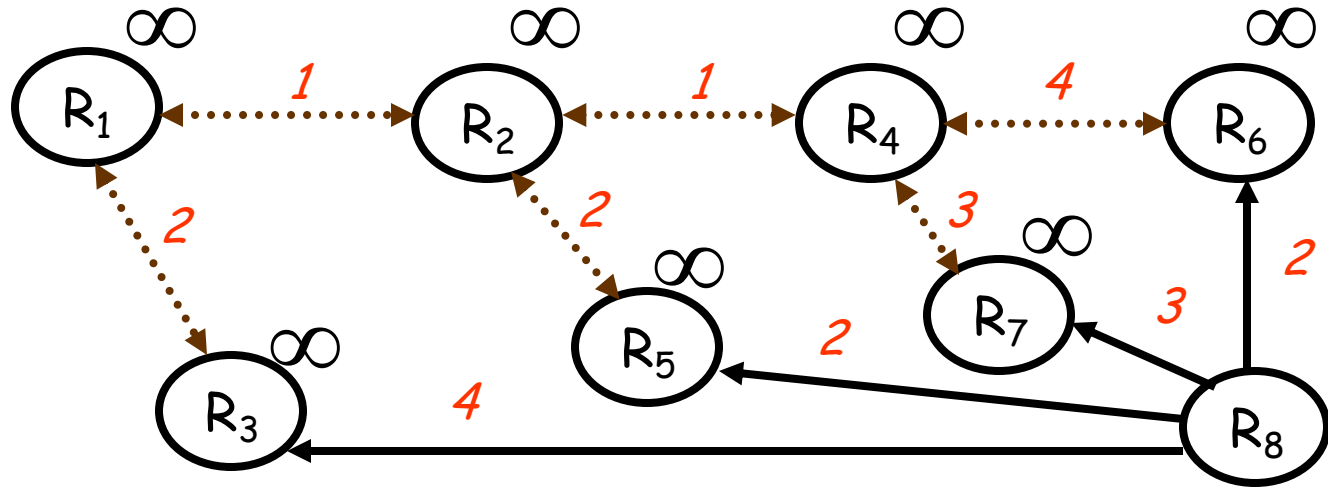
אלגוריתם B-F הוא אלגוריתם וקטור המרחקים

B-F is a Distance Vector Algorithm

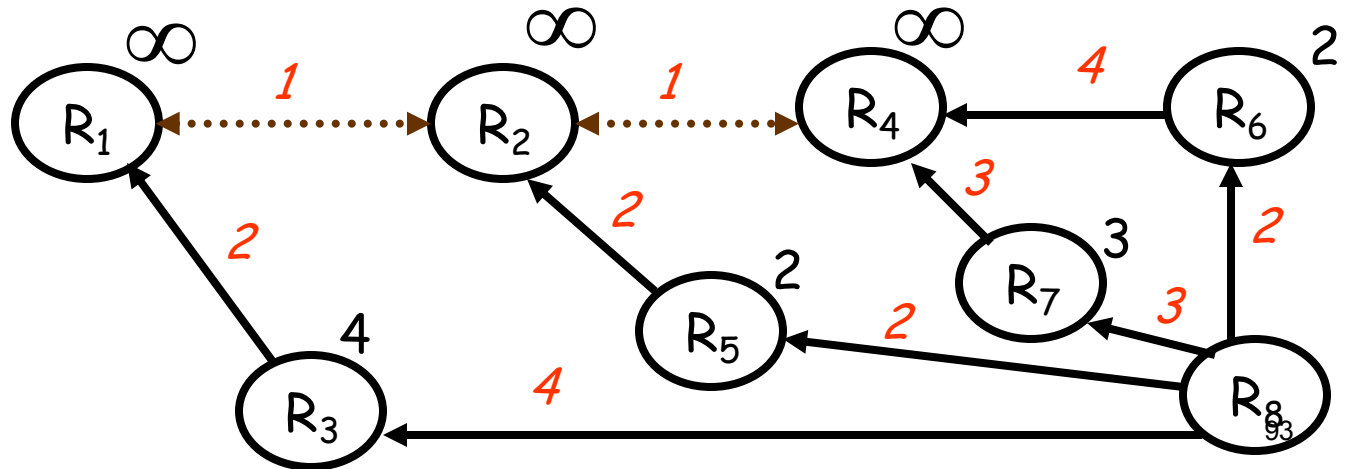


- ❖ הפתרון הוא עץ פורס (*spanning tree*) עם R_8 כשורש של העץ,
- כאשר מחירי הקשתות (הערוצים) הם המחירים של הגישה ל- R_8 .
- ❖ אלגוריתם Bellman-Ford מוצא את כל העצים הפורסים בגרף.

B-F אפיון אפיון

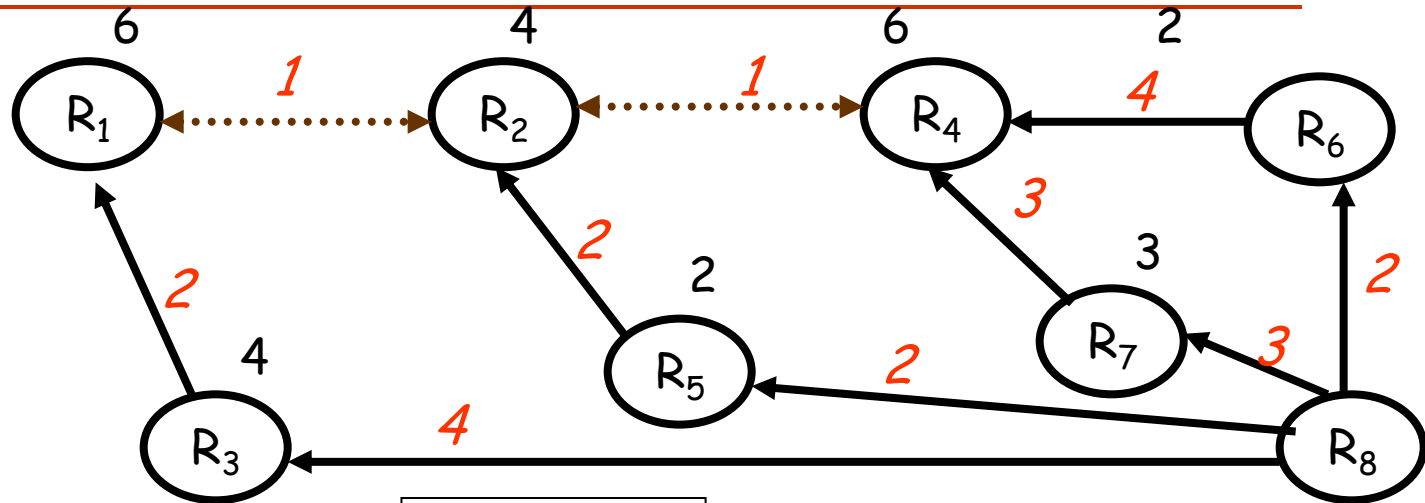


R ₁	Inf
R ₂	Inf
R ₃	4, R ₈
R ₄	Inf
R ₅	2, R ₈
R ₆	2, R ₈
R ₇	3, R ₈



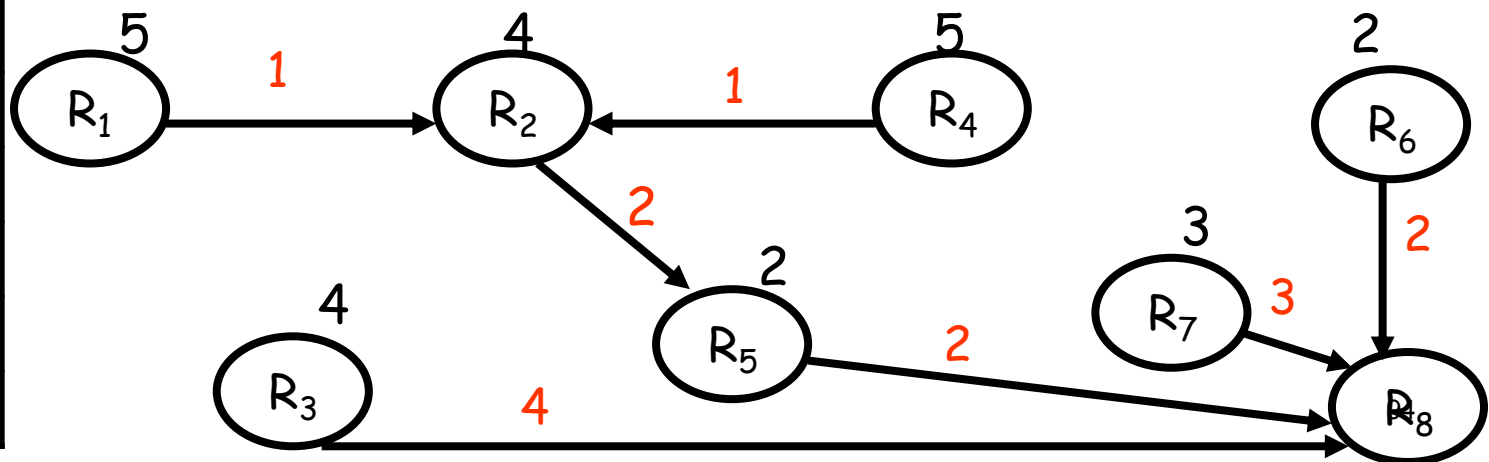
דואנא פאפולא אפאריטא B-F

R ₁	6, R ₃
R ₂	4, R ₅
R ₃	4, R ₈
R ₄	6, R ₇
R ₅	2, R ₈
R ₆	2, R ₈
R ₇	3, R ₈



פתרון

R _{1Z}	5, R ₂
R ₂	4, R ₅
R ₃	4, R ₈
R ₄	5, R ₂
R ₅	2, R ₈
R ₆	2, R ₈
R ₇	3, R ₈



שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

ניתוק הירארכי Hierarchical Routing

לימוד הניתוב שלנו עד עכשיו:

- אידיאלי (חסר פגמים).
- כל הנתבים זהים.
- רשת שטוחה.
- ... כל זה לא נכון באופן מעשי כי:

קנה מידה: עם 200 מיליון יעדים אוטונומיה אדמיניסטרטיבית

- אין אפשרות לאחסן את כל היעדים
- אינטרנט = רשת של רשתות.
- טבלת הניתוב.
- מנהל כל רשת רוצה לפקח על הניתוב של הרשת שלו.
- טבלת הניתוב מוצפת שינויים של הערוצים.

ניתוח היררכי Hierarchical Routing

□ צירוף הנתב לתוך איזור "מערכת אוטונומית" (autonomous systems (AS))

□ נתב עם אותה מערכת אוטונומית מריץ את אותם פרוטוקולי ניתוב:
■ פרוטוקול ניתוב פנימי "intra-AS" המתמקדים בעיקר בביצועים.
■ נתבים עם מערכת אוטונומית שונה יכולים להריץ פרוטוקולי ניתוב "inter-AS" שונים, והם מתמקדים לא רק בביצועים אלא גם באפשרות לקבוע מדיניות.

□ נתב Gateway

■ מחבר ישירות לנתב עם מערכת אוטונומית שונה.

פתרון אפשרי – מודל היררכי.

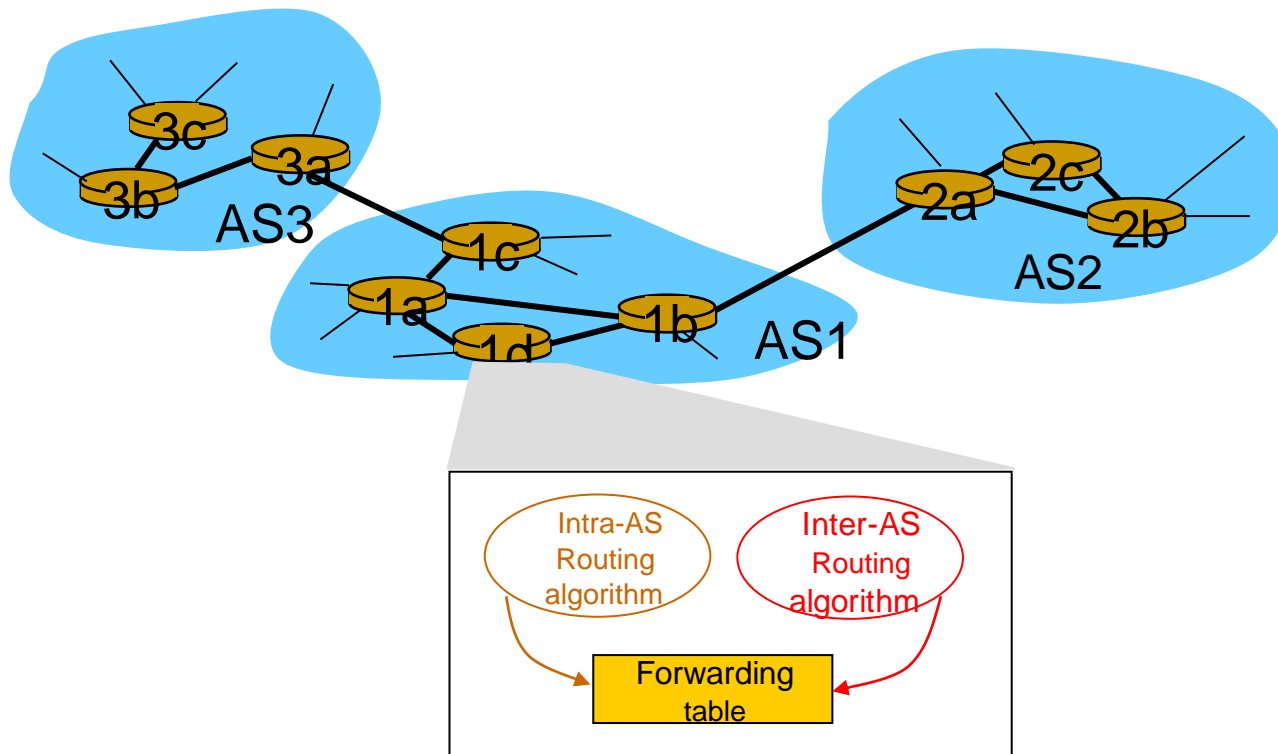
□ הרשת מחולקת לאוטונומיות (Autonomous Systems), כאשר כל אוטונומיה יכולה להכיל מספר רשתות, כך שהיא מעין "אינטרנט קטנה" (למשל קמפוס בן-גוריון יכול להיות אוטונומיה). אם נרצה להעביר הודעות בתוך אוטונומיה אז נשתמש בנתבים מסוימים של אותה אוטונומיה, ואם נרצה להעביר הודעה לאוטונומיה אחרת, נשתמש בנתב שיודע להעביר הודעות בין אוטונומיות.

□ פרקנו את הבעיה לשתי רמות של אלגוריתמי ניתוב – פנים אוטונומי `intra autonomous system routing protocols (intra-AS)` ובין אוטונומי. `Inter autonomous system routing protocols (inter-AS)`

□ אלגוריתם הניתוב הבין אוטונומי (inter-AS) צריך להיות אחיד לכל הרשת, אבל בתוך כל AS אפשר להריץ אלגוריתם ניתוב אחר. המחשבים בכל AS צריכים להכיר רק את `intra AS routing protocol` של אותה אוטונומיה.

□ בכל AS חייב להיות לפחות נתב אחד שמכיר את הפרוטוקול הבין אוטונומי, אחרת ה-AS הזו תהיה לא תוכל לתקשר עם שאר הרשת.

מקושרים הדדית – מספר ASes



טבלת שליחות "מקונפגת" ע"י אלגוריתמי הניתוב של intra-AS ו-inter-AS :

■ intra-AS קובעת כניסות עבור יעדים פנימיים.

■ inter-AS קובעים כניסות עבור יעדים חיצוניים.

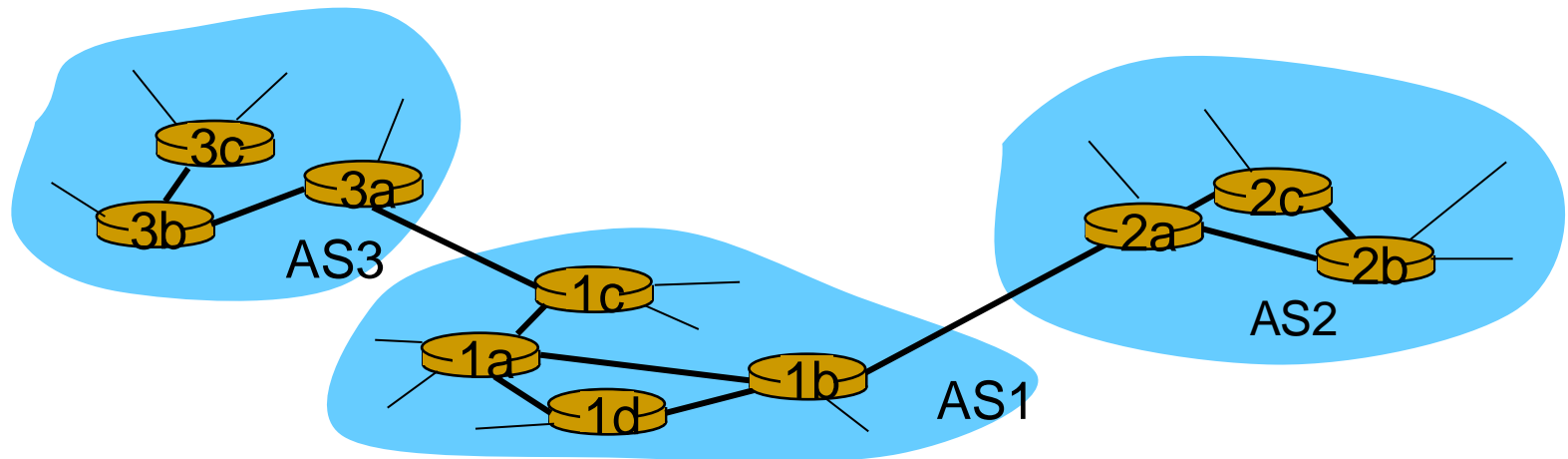
Inter-AS משימות

AS1 חייב:

1. ללמוד איזה יעד אפשרי דרך AS2, ואיזה דרך AS3.
2. להפיץ את המידע על הנגישות הזו לכל הנתבים ב- AS1.

□ נניח שהנתב ב-AS1 מקבל חבילת מידע עם יעד מחוץ ל-AS1.

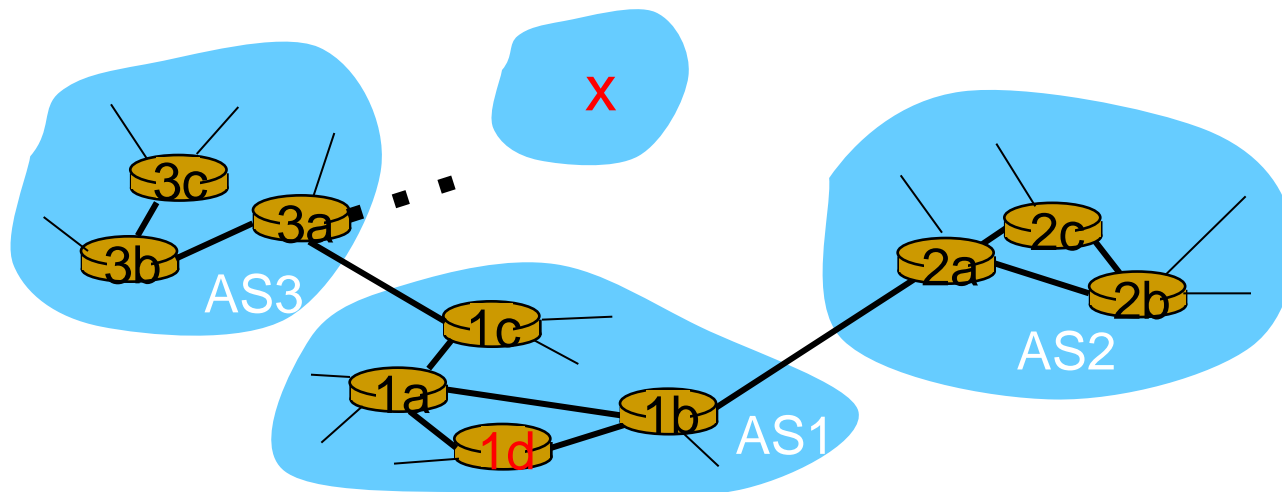
□ הנתב חייב לשלוח את החבילה לנתב gateway, אבל לאיזה מהם? גם AS2 וגם AS3 באים בחשבון!



דואל: קביעת טבלת שליחות

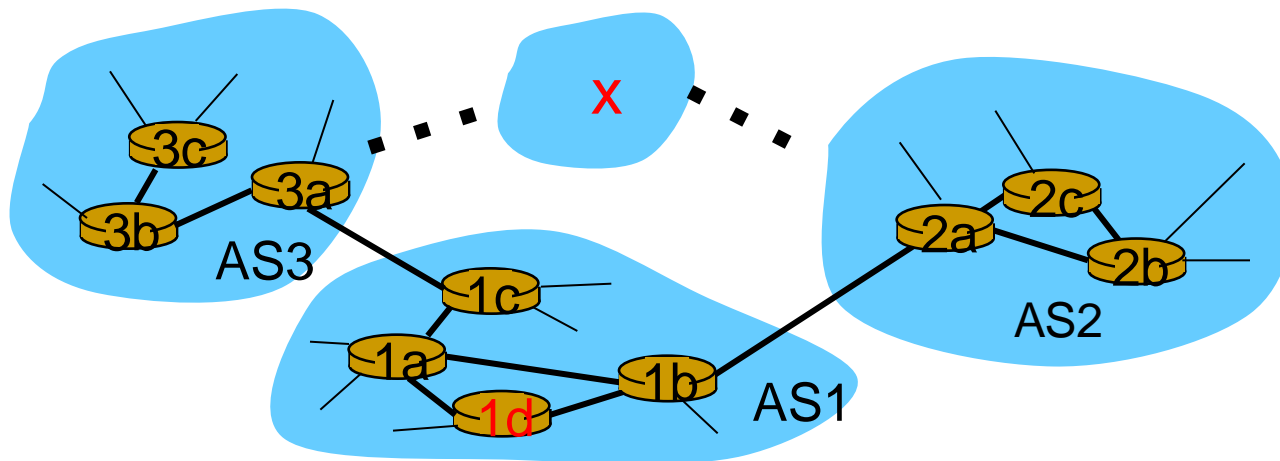
בנתב אחד – "1d"

- ניהל ש-AS1 לומד (דרך פרוטוקול inter-AS) שנת רשת x יכולה להיות גישה דרך AS3 (gateway 1c), אבל לא דרך AS2.
- פרוטוקול inter-AS מפיץ מידע על גישות זו לכל הנתבים הפנימיים.
- הנתב 1d קובע בעזרת מידע על ניתוב מ-intra-AS שהממשק שלו הוא I והוא הנתביב הנמוך ביותר ל-1c.
- נקבע בטבלת השליחות (forwarding table) רישום/כניסה ל- (x, I) .



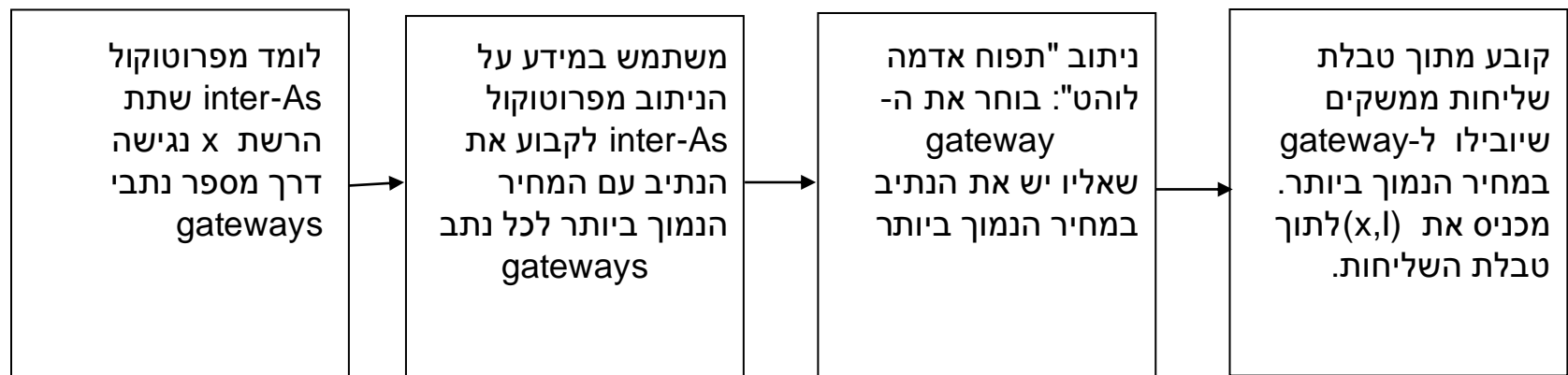
דוגמא: בחירה בין מספר ASes

- נניח עכשיו ש- AS1 לומד מפרוטוקול inter-AS שתת הרשת **X** נגישה מ- AS3 ומ- AS2.
- ל"קינפוג" טבלת השליחות (forwarding table) הנתב **1d** חייב לקבוע לקראת איזה נתב gateway הוא חייב לשלוח חבילות עבור היעד **X**.
- זה גם תפקיד של פרוטוקול ניתוב inter-AS.

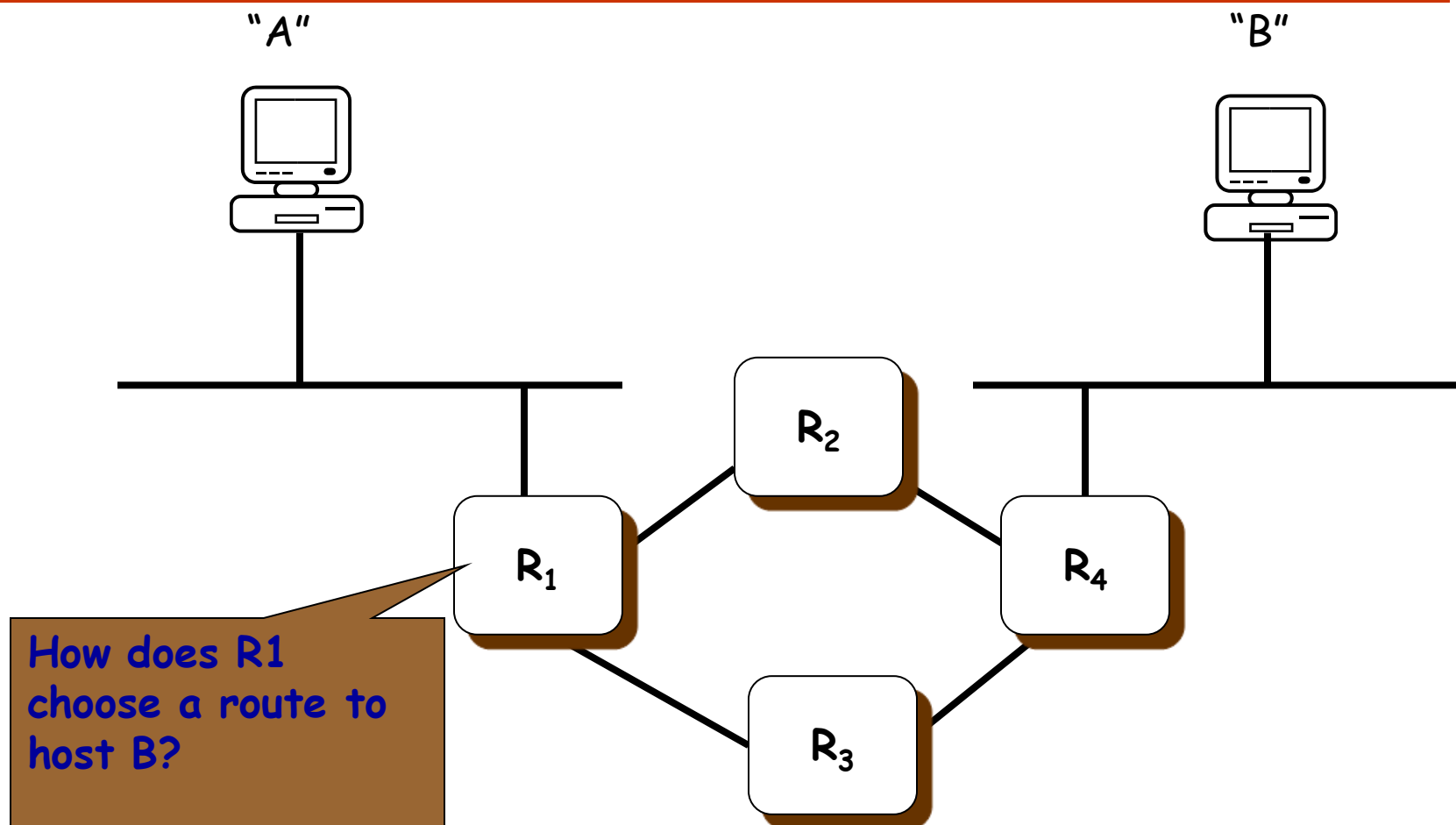


דוגמא: בחירה בין מספר ASes

- עכשיו נניח ש-AS1 לומד מפרוטוקול inter-AS שרת x היא נגישה גם מ-AS3 וגם מ-AS2.
- ל"קינפוג" טבלת השליחות (forwarding table) הנתב $1d$ חייב לקבוע לקראת איזה נתב gateway הוא חייב לשלוח חבילות עבור היעד x .
- זה גם תפקיד של פרוטוקול ניתוב inter-AS.
- **ניתוב "תפוח אדמה לוהט"**: לשלוח חבילה לקראת "הכי קרוב" של שני הנתבים.



The Routing Problem

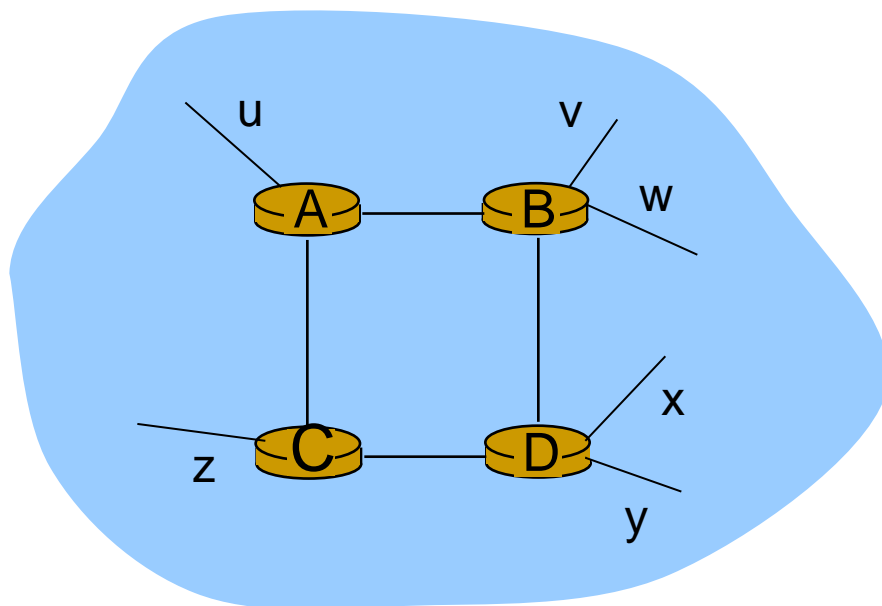


שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

RIP (Routing Information Protocol)

- מפעיל אלגוריתם וקטור מרחקים (distance vector) כאשר מרחק מוגדר ככמות הנתבים בדרך אל היעד.
- בשימוש כחלק מתוכנת BSD-UNIX משנת 1982.
- מטריצת מרחקים של hops (נקודת חיבור בין מחשבים ברשת, דרכה עובר המידע ברשת מנקודה לנקודה) ← מקסימום 15 hops.



From router A to subsets:

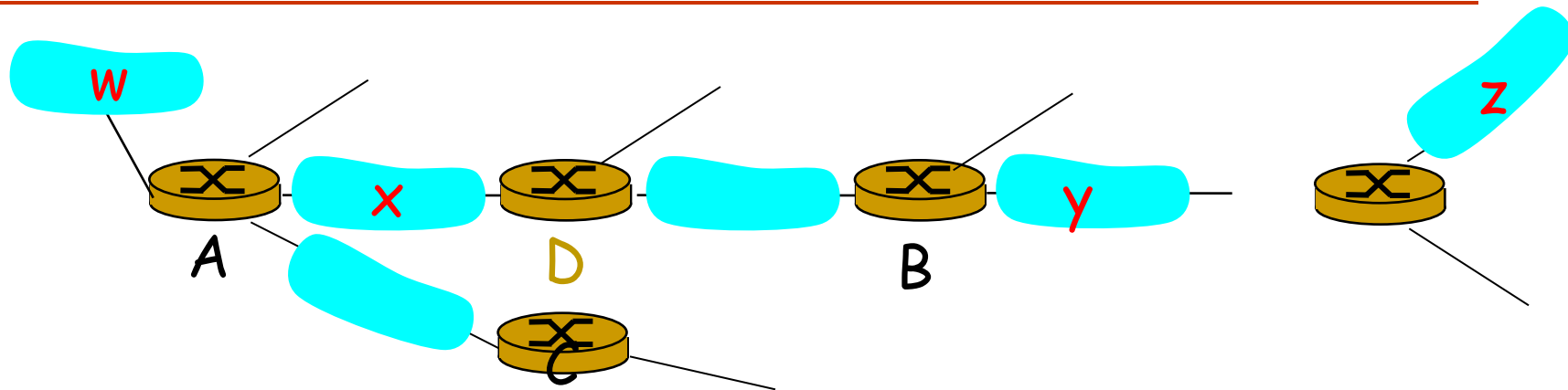
<u>destination</u>	<u>hops</u>
u	1
v	2
w	2
x	3
y	3
z	2

RIP פרסום "fe"

□ וקטורי מרחקים (distance vectors): מוחליפים בין השכנים כל 30 sec דרך הודעות תגובה (Response Message), הנקראות גם כ"פרסום" (advertisement).

□ כל "פרסום" (advertisement): רשימה של למעלה מ-25 יעדים ברשת הכוללים גם AS.

RIP-*f* *kNc1?*



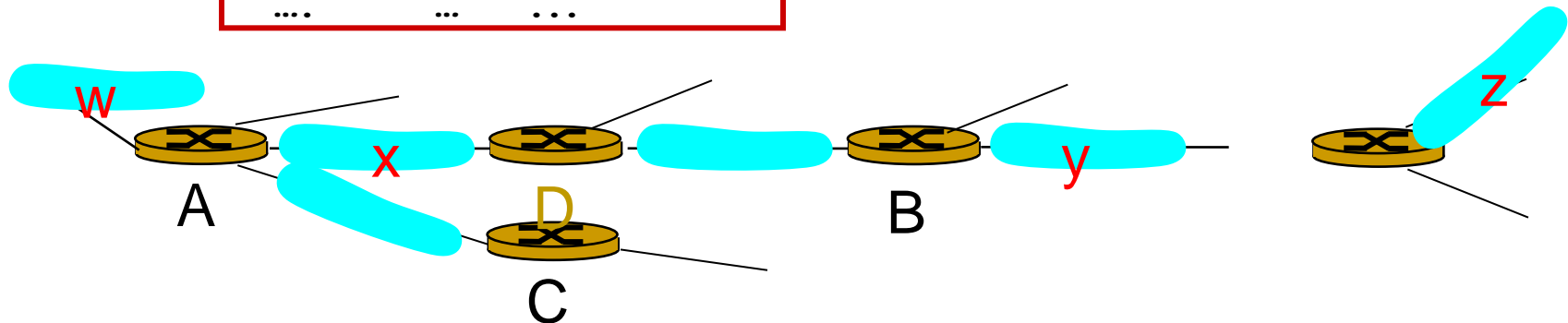
Destination Network	Next Router	Num. of hops to dest.
w	A	2
y	B	2
z	B	7
x	--	1
...

Routing table in D

RIP-*ל* *כ* *נ* *ד* *ל* *?*

Dest	Next	hops
<i>w</i>	-	1
<i>x</i>	-	1
<i>z</i>	<i>C</i>	4
...

Advertisement
from A to D



Destination Network	Next Router	Num. of hops to dest.
<i>w</i>	<i>A</i>	2
<i>y</i>	<i>B</i>	2
<i>z</i>	<i>B A</i>	7 5
<i>x</i>	--	1
...

Routing table in *D*

RIP: נפילת ערוצים והתאוששות

אם לא נשמע "פרסום" (advertisement) אחרי 180 שניות ← שכנים / ערוצים מוכרזים "מתים".

■ נשלל תוקפו של ניתוב דרך שכנים.

■ "פירסום" (advertisements) חדש נשלח לשכנים.

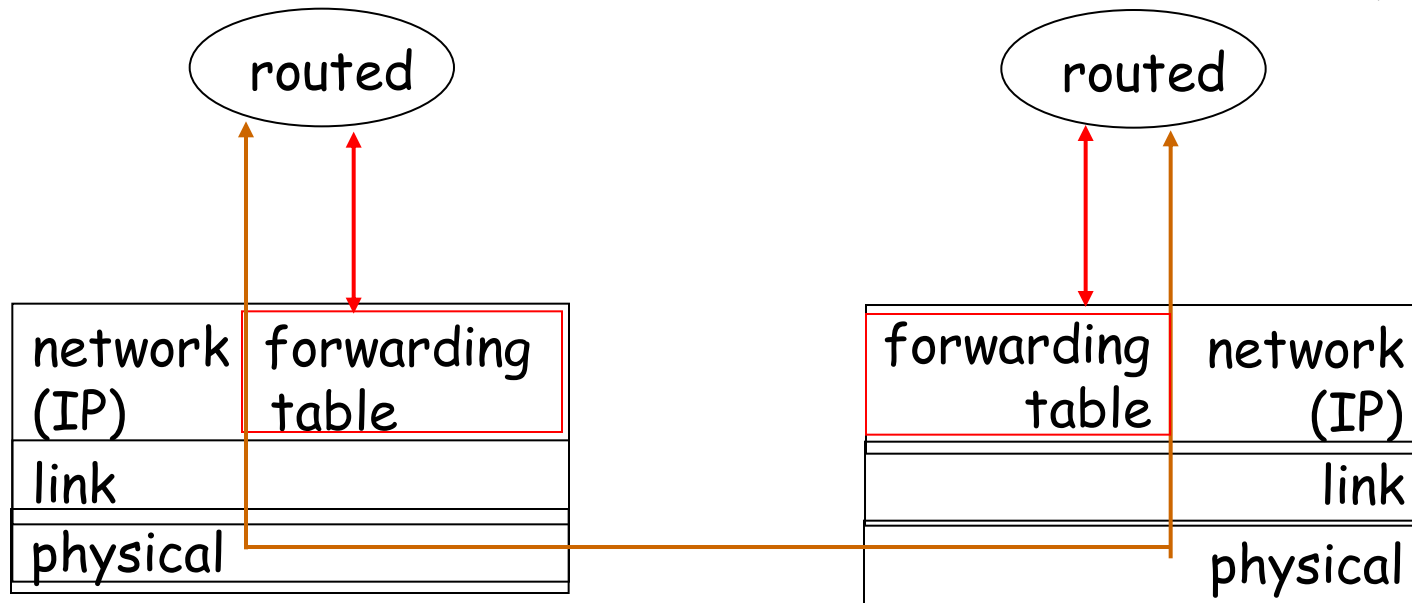
■ שכנים לפי תור שולחים "פירסום" (advertisements) חדש (אם הטבלה השתנתה).

■ המידע על נפילת ערוצים מהיר ומופץ לכל הרשת.

■ משתמשים ב-"רעל נגדי" (*poison reverse*) למנוע לולאות פינג-פונג (מרחק אינסופי = 16 hops).

RIP: תהליך כניית הטבלת הניתוב

- ניהול טבלת הניתוב של RIP מתבצע ע"י התהליך נקרא שד-הניתוב (route-d (daemon)).
- הפירסום (advertisements) נשלח בחבילות UDP, במחזוריות החוזרת על עצמה.



שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

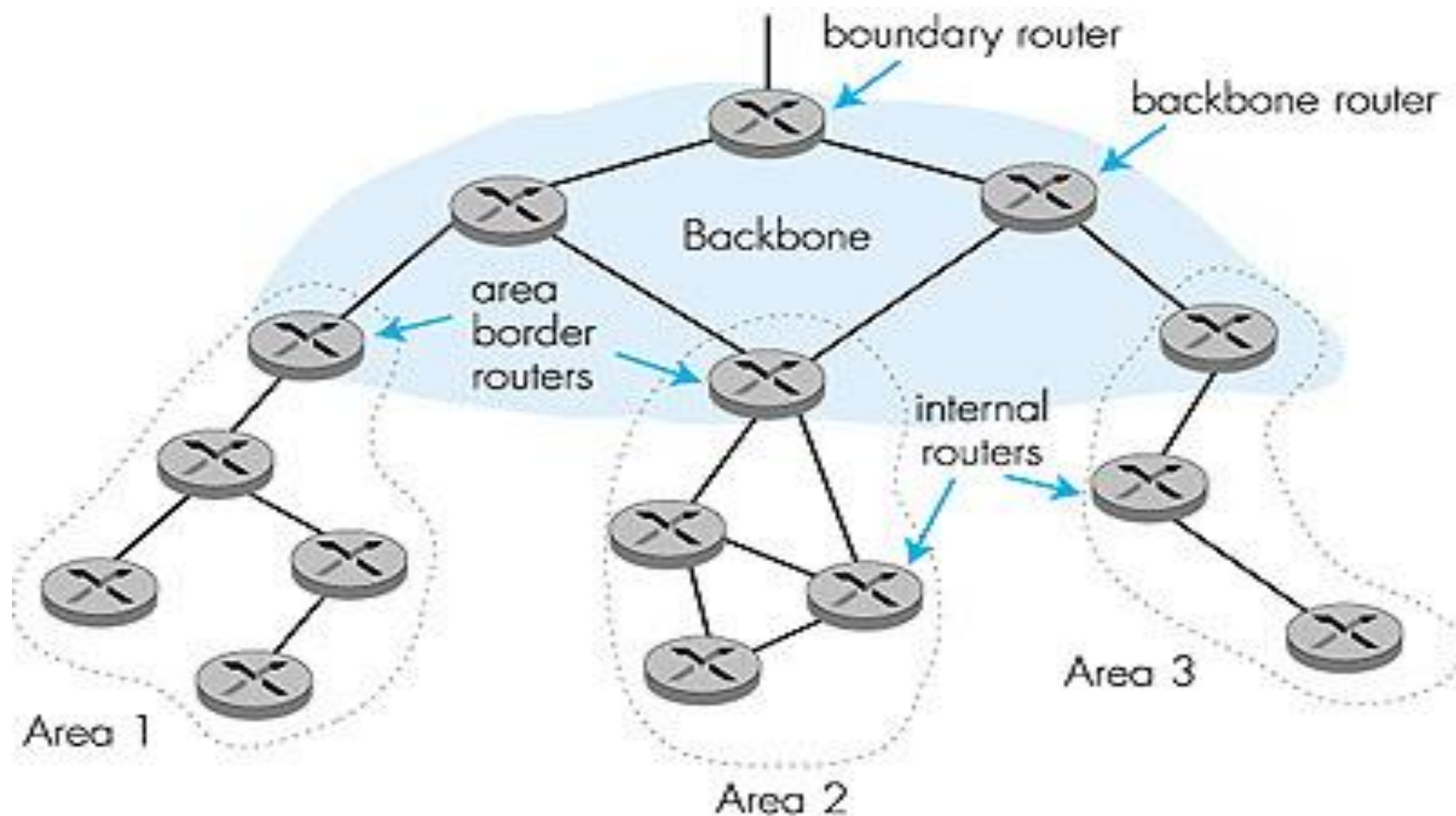
פרוטוקול OSPF (Open Shortest Path First)

- כל צומת מכיר את כל גרף הרשת.
- כל צומת מחשב את המסלולים הקצרים ביותר ממנו לכל שאר הצמתים ע"י אלגוריתם של Dijkstra.
- כל צומת בונה טבלת ניתוב על-סמך תוצאות האלגוריתם. הטבלה מציינת את ה-Next hop לכל יעד.
- ניתוב אופטימלי מחייב שלכל הצמתים יהיה גרף רשת זהה.
- כל צומת מפיץ מידע לגבי נתבים/רשתות המחוברים אליו. מידע זה מוצף לכל הרשת.
- פרוטוקול פתוח: פומביות אפשרית.
- משתמש באלגוריתם מצב הערוץ (Link State).
- חבילות Link State מופצות.
- מפה טופולוגית של הגרף נמצאת אצל כל קודקוד.
- חישוב הניתוב מתבצע ע"י האלגוריתם של Dijkstra.
- OSPF מפרסם (advertisement) כניסה אחת לכל נתב שכן.
- הפירסום מופץ ל-AS.
- נשיאה בהודעת OSPF ישירות ל-IP (ולא דרך TCP או UDP).

פרוטוקול OSPF

- כל נתב אחראי לדווח לכולם על הקשתות (רשתות) היוצאים ממנו. הפצת המידע בהצפה. ישנם מספר סוגי נתבים:
 - נתבים רגילים - Designated Router.
 - נתבים על גבול האזור - Area Border Router.
 - נתבים על גבול ה-AS - External Border Router
- OSPF תומך בחלוקת ה-AS לאזורים, תוך דרישה על קיום Backbone Area (אזור אשר מחובר לכל שאר האזורים):
 - ההצפה של מידע על קשתות באזור נעשה רק באיזור עצמו.
 - Area Border Routers מציפים מידע לגבי יעדים מהאזורים האחרים.
 - External Border Routers מציפים מידע לגבי יעדים מחוץ ל-AS.
- סוגי "קשתות" (דיווחים / רשומות במבנה נתונים של הנתבים):
 - Router Link – דווח על קשתות היוצאות מנתב.
 - Network Link – דווח על הקשתות היוצאות מ-Network.
 - External Link – דווח על יעדים מחוץ ל-AS.

ה'רארכ'ית OSPF



ה'רארכ'ית OSPF

- **ה'רארכ'יה בשתי רמות:** רמת השטח, השלד.
 - פירסום (advertisements) מצב הערוץ (Link-state) רק ברמת השטח (area).
 - לכל קודקוד יש מידע על טופולוגית השטח, ויודע רק כיוון (הנתיב הקצר ביותר) לרשת בשטחים האחרים.
- **נתבי גבול השטח (area border routers):** "סיכום" מרחקים לרשת בשטח "האישי", מודיע בפומבי לנתבי גבול השטח האחרים.
- **נתבי השלד (backbone routers):** מריצים ניתוב OSPF המוגבל לשלד (backbone).
- **נתבי התחום (boundary routers):** מחוברים ל-AS's אחרים.

אלגוריתם ההצפה

□ לכל דווח יש את הנתונים הבאים:

- מזהה הקשת - LS-ID, וסוג.
- מספר סידורי - LS Sequence Number.
- גיל.

□ התיישנות מבנה הנתונים:

- הגיל מקודם ב 1- כל מעבר דרך נתב וכל שנייה.
- כאשר מגיעים ל-MaxTime יש לזרוק את הרשומה.
- זריקת הרשומה גוררת הצפת הודעה על כך (עדכון שבו הזמן הינו MaxTime).
- כל נתב מפרסם את הקשתות שבאחריותו כל פרק זמן (בד"כ חצי של MaxTime). שינוי במצב קשת גורר שידור מיידי. כל פירסום כזה הינו עם גיל 0 ומספר סידורי עולה.

□ פרוטוקול ההצפה - בקבלת הודעת עדכון משכן:

- אם רשומה אינה קיימת, הכנס למבנה נתונים והפץ לשכנים אחרים (והגדל גיל).
- אם רשומה קיימת, ובעלת מספר סידורי גדול יותר, שלח רשומה ממבנה נתונים לשכן.
- אם רשומה קיימת, ובעלת מספר סידורי קטן יותר, עדכן מבנה נתונים, והפץ לשכנים אחרים.
- אם רשומה קיימת, ובעלת אותו מספר סידורי, אז:
 - אם גיל ההודעה MaxAge, קבל הודעה (ז"א מחק ממבנה נתונים, והפץ הודעה לשכנים אחרים).
 - אם יש הבדל גיל עד 15 דקות, התעלם.
 - אחרת השתמש בהודעה המאוחרת יותר (גיל נמוך יותר).

תכונות מתקדמות של OSPF

- **אבטחה:** כל הודעות OSPF מאושרות (למנוע חדירה של הודעות זדוניות).
- **הכפלה** של נתיבים באותו מחיר מורשים (לעומת נתיב אחד ב-RIP).
- עבור כל ערוץ, הכפלה של מטריצת מחיר עבור TOS שונה (לדוגמא, ערוץ לוויין יהיה בעל מחיר נמוך עבור דרישה נמוכה שדורשת מאמץ קטן, וגבוה בזמן אמת).
- מאחד תמיכה בשידור מרובב (**multicast**) ויחיד (**uni-**):
- **Multicast OSPF (MOSPF)** משתמש באותה טופולוגית נתונים בסיסית כמו OSPF.

שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

ניתוח inter-AS כא'נטרנט: BGP

□ **BGP (Border Gateway Protocol)**: באופן מעשי הסטנדרט האינטרנט.

□ BGP מספק לכל מערכת אוטונומית (AS) אמצעי ל:

1. לקבל תת רשת נגישה ומידע משכנים בעלי מערכת אוטונומית (AS).

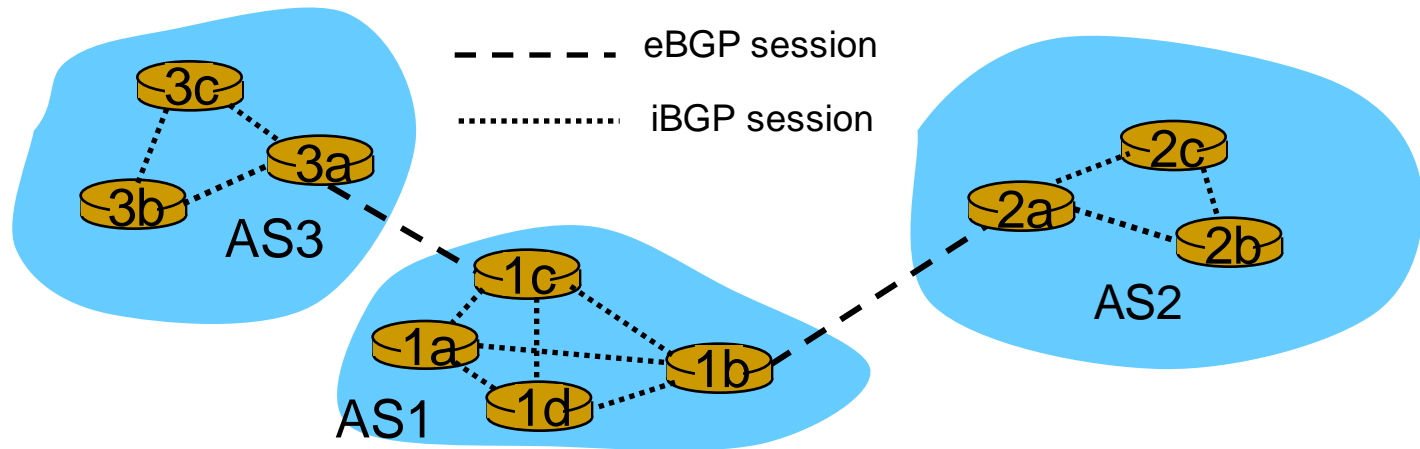
2. להפיץ מידע על נגישות לכל הנתבים הפנימיים בכל מערכת אוטונומית (AS).

3. לקבוע נתיב "טוב" לתת רשת על בסיס מידע על נגישות ומדיניות.

□ מאפשר לתת רשת להפיץ את ניסיונה לכל האינטרנט בסגנון: *"I am here"*.

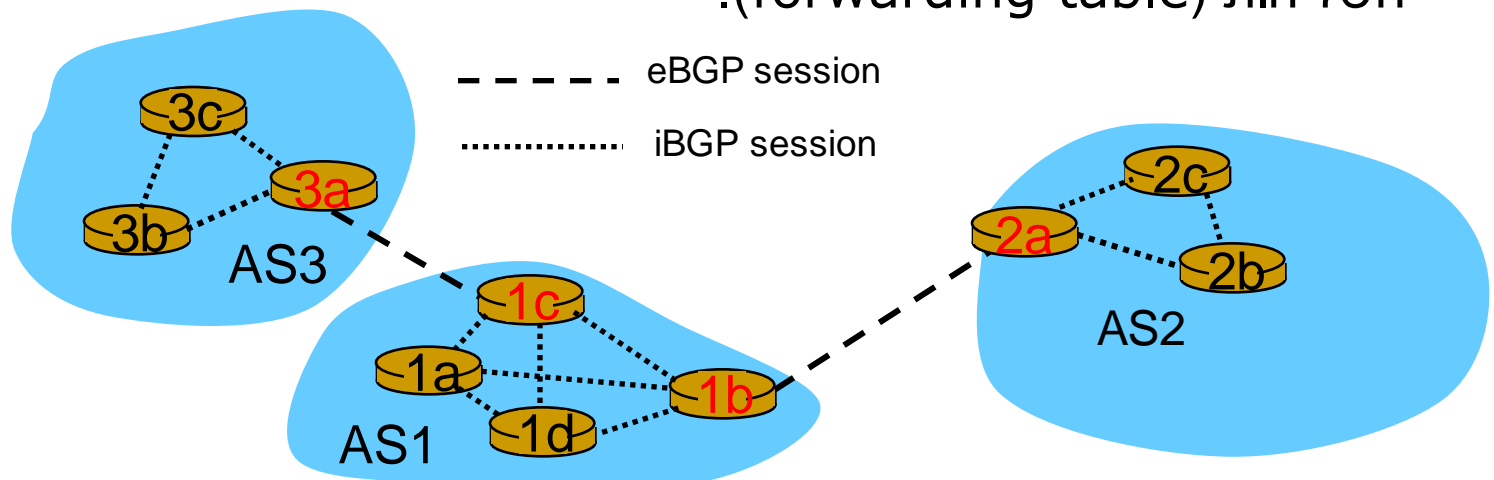
BGP גסיס

- זוג של נתבים (עמית BGP) מחליפים מידע על ניתוב על סמך חיבור TCP קבוע למחצה: **שיחת BGP (BGP sessions)**.
- שיחות BGP (BGP sessions) הנדרשת לא מתאימה לשכבה הפיסית.
- כאשר AS2 מפיצה "קידומת (prefix)" ל-AS1:
- AS2 מבטיחה שהיא תשלח כל כתובת חבילת המידע קדימה לקידומת זו.
- AS2 יכולה להבטיח קידומת בפירסום שלה.



הפצת מידע על נאישות

- שימוש בשיחת eBGP בין 3a ל-1c, AS3 שולח קידומת של מידע על נגישות ל-AS1:
- 1c אצי יכול להשתמש ב-iBGP להפיץ מידע על קידומת חדשה לכל הנתבים ב-AS1.
- 1b אצי יכול להפיץ שוב מידע על נגישות חדשה ל-AS2 באמצעות שיחת eBGP מ-1b ל-2a.
- כאשר הנתב לומד על קידומת חדשה, תותר כניסה חדשה לקידומת בטבלת השליחות (forwarding table).



מאפייני נתיב & ניתוב BGP

□ קידומת (prefix) שמתפרסמת כולל מאפייני BGP.

■ $\text{prefix} + \text{attributes} = \text{"route"}$

□ שני מאפיינים חשובים:

■ **AS-PATH**: מכיל מערכות אוטונומיות (ASs) שדרכם הקידומת

מפרסמת את הדרך: לדוגמא, AS 67, AS 17.

■ **NEXT-HOP**: מצביע על נתב AS פנימי ספציפי ל-AS שבו

התרחש ה-hop הבא (יכול להיות על מספר ערוצים מה-AS הנוכחי ל-AS שבו התרחש ה-hop הבא).

□ כאשר נתב gateway מקבל פירסום על נתיב, הוא

משתמש ב-"**יבוא מדיניות (import policy)**" על מנת קבל או לדחות.

בחירת נתיב BGP

□ הנתב יכול ללמוד על יותר מאשר נתיב אחד לכמה קידומות. הנתב חייב לבחור נתיב.

□ הסרת כללים:

1. מאפיני ערך של ביצועים מקומיים: מדיניות החלטות.

2. הנתב הקצר ביותר של AS-PATH.

3. הנתב הקרוב ביותר עבור NEXT-HOP.

4. קריטריונים נוספים.

הנדסת BGP

□ החלפת הודעות BGP ע"י TCP.

□ הודעות BGP:

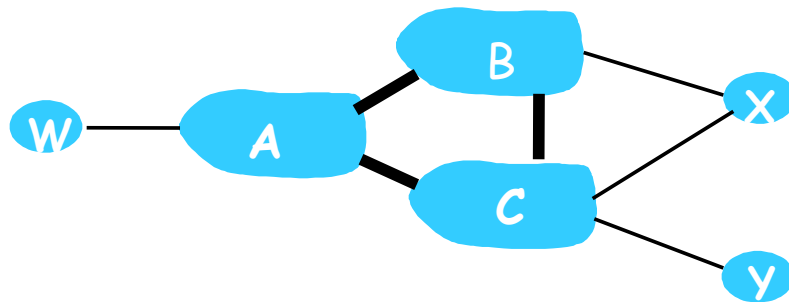
■ **פתוח:** חיבור TCP פתוח לעמיתים ומאמתים כך את שולח.



■ **עדכון:** פירסום נתיבים חדשים (או להוציא ישנים).

■ **לשמור בחיים:** שומר קישורים/חיבורים חיים בהעדר עדכון, ובנוסף אישור (ACKs) פותח בקשה.

■ **הודעות:** דיווח על תקלות בהודעות הקודמות, בנוסף משמש לסגור חיבורים.

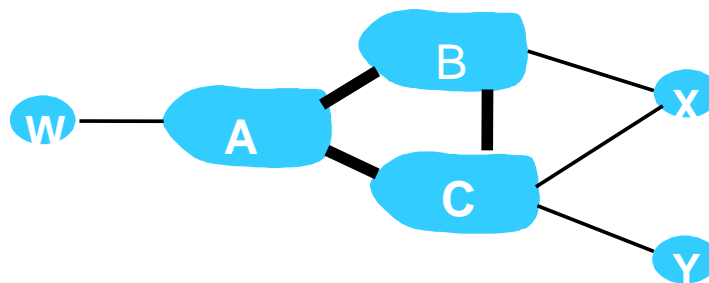
BGP (1) מדיניות ניתוק



legend:  provider network
 customer network:

- A,B,C ספקים של רשת.
- X,W,Y הם הלקוחות של ספקי הרשת.
- X שייך לשתי רשתות, כלומר מסופח לשתי רשתות.
- X אינו רוצה נתיב מ-B דרך X ל-C.
- לכן X לא יפרסם ל-B נתיב ל-C.

BGP (2) מדיניות ניתוק



legend:  provider network
 customer network:

A מפרסם נתיב AW ל-B. ☐

B מפרסם נתיב BAW ל-X. ☐

האם B מפרסם נתיב BAW ל-C? ☐

○ אין סיכוי! B מקבל שאין "הכנסה" עבור ניתוב CBAW מאחר שגם לא W

ואף לא C הם לקוחות של B.

○ B רוצה לכפות על C לנתב ל-W דרך A.

○ B רוצה לנתב רק ל-M- הלקוחות שלו.

למה קיים שוני בניתוח Inter-AS / Intra-AS

מדיניות:

Inter-AS : המנהל רוצה לבקר על ניתובי התנועה שלו, ואיזה נתיב עובר באיזה רשת.

Intra-AS: מנהל יחיד, לכן אין מדיניות החלטות נדרשת.

קנה מידה:

ניתוב הירארכי שומר על גודל טבלה, מוריד עדכוני תנועה.

ביצועים:

Intra-AS: יכול להתפקס על הביצועים.

Inter-AS: דרך המדיניות הוא יכול לשלוט על הביצועים.

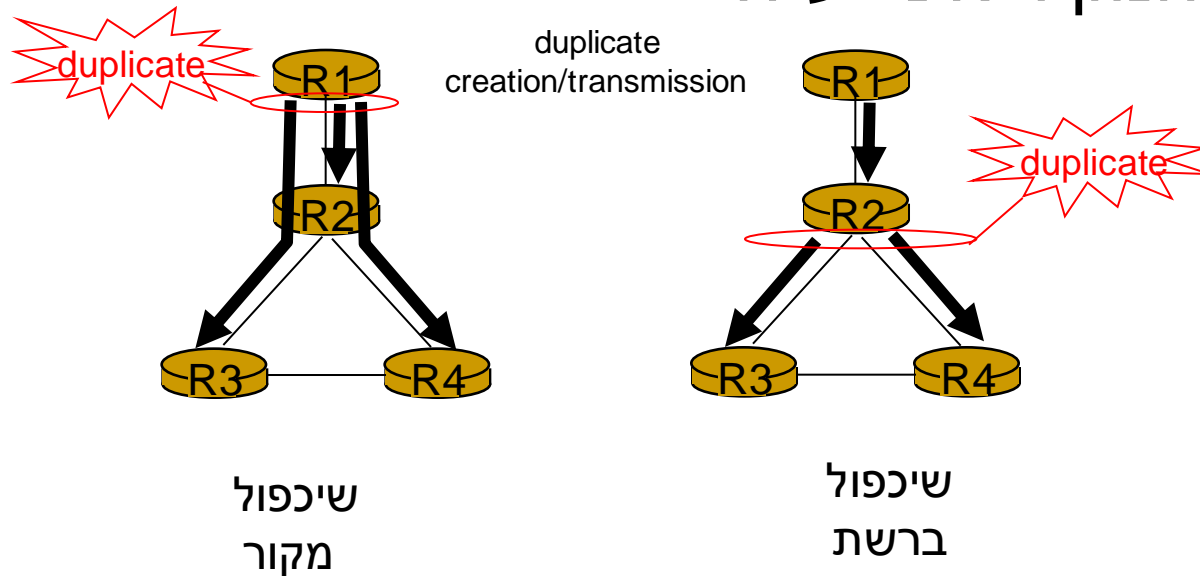
שכבת הרשת

- מבוא
- מעגלים וירטואלים וחבילות המידע ברשת
- מה יש בתוך הנתב (router)
- פרוטוקלי אינטרנט
- אלגוריתמי ניתוב:
 - Link state
 - Distance Vector
 - Hierarchical routing
- ניתוב באינטרנט:
 - RIP
 - OSPF
 - BGP
- ניתוב Broadcast ו-multicast
 - Datagram format
 - IPv4 addressing
 - ICMP
 - IPv6

Broadcast ניתוח

□ העברת חבילות מהמקור לכל הצמתים האחרים.

□ שיכפול המקור אינו יעיל:



□ שיכפול המקור: אך המקור קובע את הכתובות שיקבל ?

שיכנוף ברשת

□ הצפה: כאשר צומת מקבלת חבילת Broadcast הוא שולח עותק לכל השכנים.

■ בעיה: מעגל & "סערת" broadcast.

□ הצפה מבוקרת: צומת שולח חבילת Broadcast רק אם לא קיבל את אותה חבילה קודם.

■ צומת שומר את המסלול של חבילות שכבר נשלחו ב-broadcast.

■ או, רק שולח חבילות קדימה אם הם מגיעים בנתיב הקצר ביותר בין הצומת והמקור.

□ עץ פורש

■ אין כל חבילות עודפות המתקבלות בצומת כלשהיא.

יִתְרוֹנוֹת: □

- אמינות גבוהה מאד (אין תלות בפרוטוקול ניתוב כלשהו).
- זמן ההגעה ליעדים הוא המינימלי האפשרי.

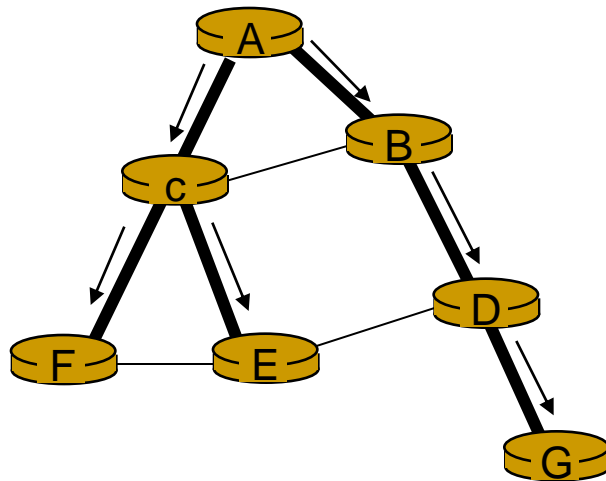
חסֵרוֹנוֹת: □

- בזבוז משאבי רשת (חבילות נשלחות על-פני כל מסלול אפשרי, במקום מסלול אחד נבחר).
- בזבוז משאבי זכרון (נתבים נדרשים לזכור ולזהות חבילות שהם קיבלו בתקופה האחרונה).
- אינה מתאימה לזרמים מהירים, אלא לעדכונים תקופתיים.

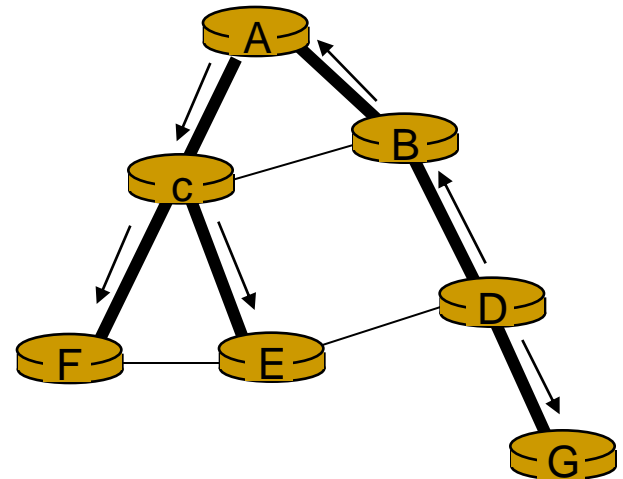
Spanning Tree עץ כולר

□ תחילה נבנה עץ פורס

□ צמתים שולחים עותקים לאורך העץ הפורס.



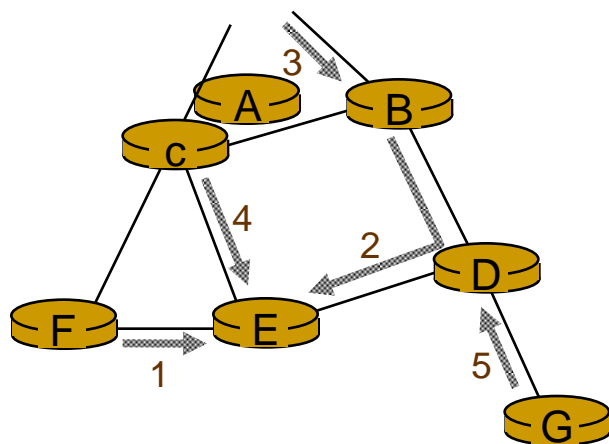
Broadcast מתחיל ב-A



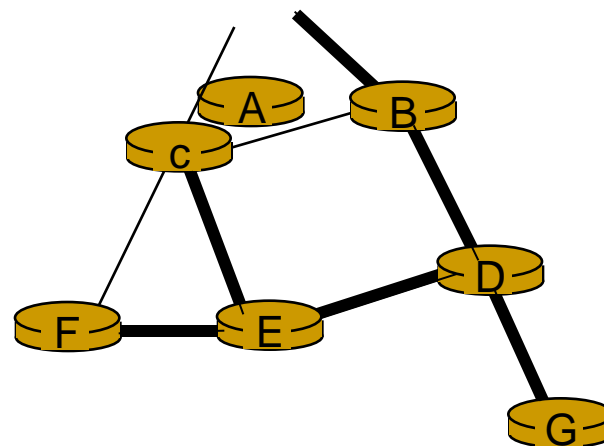
Broadcast מתחיל ב-D

יצירת עץ פורס

- צומת מרכזית
- כל צומת שולח unicast (שידור על גבי רשת, המיועד להתקבל על-ידי מחשב אחד ויחיד המחובר לרשת זו) ומצרף הודעה לצומת המרכזית.
- הודעה מועברת עד אשר היא מגיע לצומת שהיא כבר שיכת לעץ הפורס.



בניה בשלבים של עץ פורס



בניה של עץ פורס

יִתְרוֹנוֹת □

- חסכוני בזכרון (רק ביט אחד לכל ממשק לסמן האם הוא שייך לעץ הפורש).

חסרונות □

- משתמש במסלולים שאינם בהכרח יעילים.

Multicast נ"מ/ק

□ הגדרה: שידור multicast יוצא מתחנה אחת ומגיע לקבוצה של תחנות.

□ דוגמאות לישומי multicast:

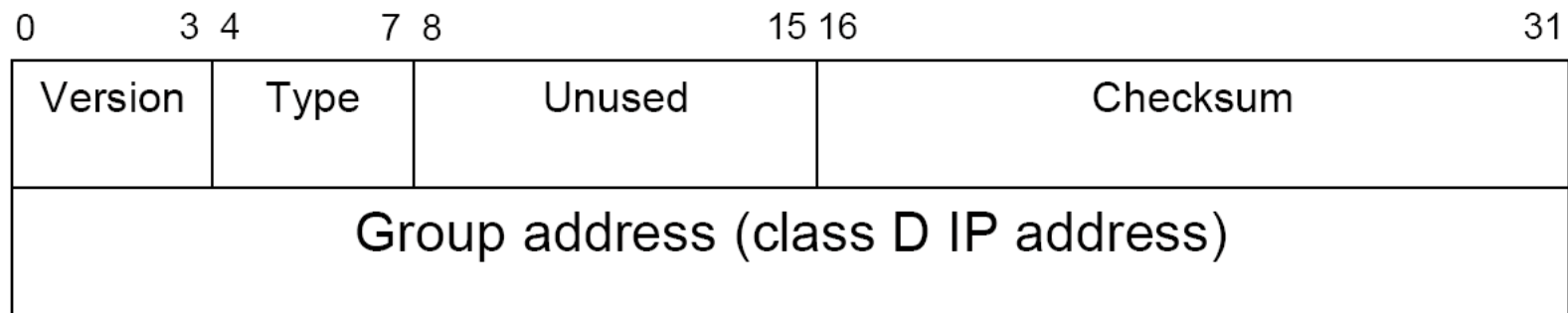
- שידור רדיו
- לימוד מרחוק
- שיחת ועידה
- הפצת מידע למנויים.

כתובות multicast ק-IP

- קבוצת הכתובות ב-Class D שמורה לתקשורת multicast.
- כל קבוצת multicast מאופיינת ברגע נתון ע"י כתובת יחידה. כתובות אלה הן דינמיות; אין שיוך קשיח בין תחנות לבין כתובות multicast, אלא כל תחנה יכולה להצטרף או להתנתק מקבוצת multicast בכל עת.
- תחנה המעוניינת לשדר לקבוצת multicast תשים את כתובת הקבוצה לשדה Destination IP. אין צורך להודיע מראש על השתייכות לקבוצה.
- תחנה המעוניינת לקלוט חבילות המיועדות לקבוצת multicast כלשהי צריכה להודיע על כך באמצעות פרוטוקול IGMP (Internet Group Management Protocol).
- מספר קטן של כתובות Class D שמורות לקבוצות בעלות משמעות מיוחדת, למשל:
 - 224.0.0.1 - כל התחנות באינטרנט (התומכות ב-multicast).
 - 224.0.0.5 - כל נתבי OSPF במערכת האוטונומית.
- מיפוי כתובות Class D לכתובות פיזיות - אופן המיפוי מכתובת IP לכתובת פיזית תלוי בטכנולוגיה של הרשת הפיזית:
 - ברשתות התומכות ב-broadcast, אפשר לשלוח חבילות המיועדות לכתובת ב-Class D באמצעות broadcast.
 - רשת Ethernet תומכת ב-multicast ברמה הפיזית.
 - ברשתות שאינן תומכות בשידור למספר תחנות בו-זמנית, אין מנוס מלשכפל כל חבילת multicast לכל אחד מהיעדים בנפרד.

פרוטוקול IGMP

- IGMP - נחשב פרוטוקול בשכבת IP.
- חבילות IGMP משודרות בתוך חבילות IP.
- חבילות IGMP הן בעלות אורך קבוע.
- מבנה חבילת IGMP:



- השדה Version מכיל תמיד 1.
- השדה Type הוא 1 עבור שאלות, 2 עבור דיווחים.

כאלי שידור חבילות

- כאשר תחנה רוצה להצטרף לקבוצת multicast, היא משדרת חבילת דיווח (type=2).
- תחנה אינה מדווחת כאשר היא מפסיקה להאזין לקבוצה, אלא מתעלמת מחבילות המגיעות אליה.
- מדי פעם, הנתב משדר ב-broadcast לרשת המקומית "חבילת שאלה" (type=1) כדי לברר מהן קבוצות ה-multicast אליהן יש תחנות מאזינות (אם בכלל).
- בתשובה לשאלה, כל תחנה המאזינה לקבוצת multicast מסוימת תשלח חבילת דיווח (type=2) עם כתובת הקבוצה.
- תחנה יכולה לשלוח יותר מחבילת דיווח אחת (אם היא מאזינה למספר קבוצות multicast).
- לפני שידור תשובה, כל תחנה ממתינה זמן קצר אקראי. אם במהלך זמן זה משודרת חבילת דיווח ע"י תחנה אחרת לאותה קבוצת multicast, התחנה מבטלת את שידור התשובה שלה.

ניתוק Multicast

□ פרוטוקול IGMP מטפל בניהול קבוצות multicast רמת הרשת המקומית - מתחנת הקצה עד הנתב הקרוב אליה.

□ Multicast routing עוסק בהפצת חבילות Multicast בין נתבים.

□ שיטות בסיסיות:

■ הצפה

■ עץ פורש

■ עץ Steiner

□ שיפורים:

■ Reverse Path Forwarding + prunes

■ Core-Based Tree

■ תקן PIM