



# Transport Layer

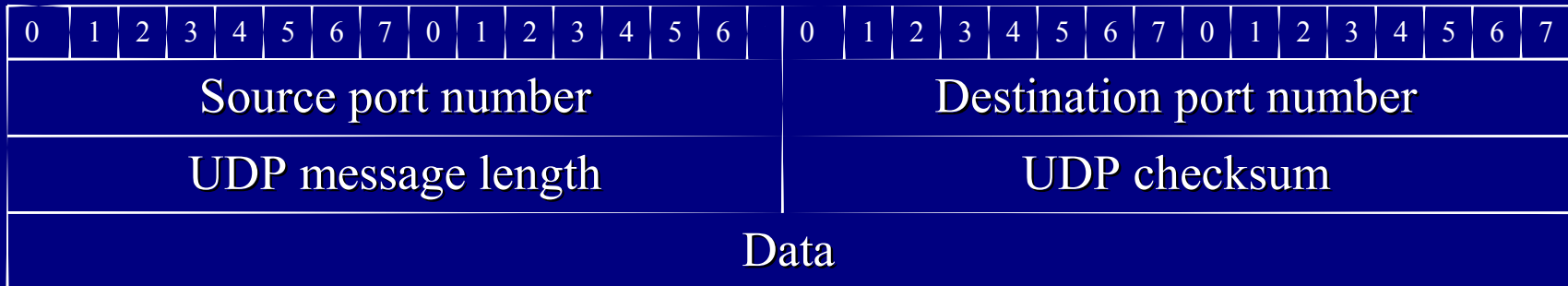
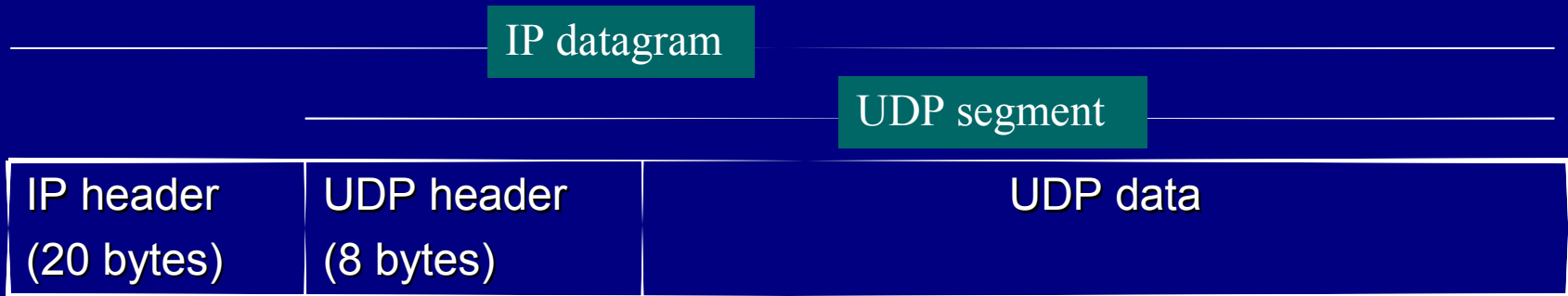
- שיכבה זו דואגת שכל המידע שנשלח ע"י תחנת המקור יגיע בצורה תקינה אל תחנת היעד.
- שיכבה זו אחראית לתקשורת בין תהליכים (Peer Entities).
- בשכבה זו 2 פרוטוקולים עיקריים: TCP, UDP.

# UDP – User Datagram Protocol

- תקשורת connectionless בין התהליכים.
- UDP מוסיף מעט פונקציונליות לזו שנותן IP.
  - ❖ מאפשר לגלות שגיאות אך לא לתקן אותן או לבצע שידורים חוזרים.
  - ❖ מאפשר לכמה אפליקציות להתבצע בו זמנית באמצעות port-number.
- כאשר מגיעה חבילה למחשב היעד, שכבת ה-IP מזהה בשדה ה-Protocol שהמידע בחבילה מיועד ל-UDP.
- UDP בודק את נכונות ה-segment שהתקבל, ומחליט האם לטפל בו או לזרוק אותו.
- UDP בודק את ה-destination port number ומחליט לאיזו אפליקציה לשלוח את ה-segment.

# UDP - המשך

מבנה UDP segment:



# TCP – Transmission Control Protocol

יתרונות הפרוטוקול:

- מאפשר demultiplexing של מספר אפליקציות מעל אותו IP (כמו UDP).
- אמיץ (end-to-end reliability).
- תקשורת connection-oriented בין תהליכים.

חסרונות הפרוטוקול:

- מאוד מסובך לעומת UDP.
- קיימות אפליקציות רבות (שידור וידאו או טלפוניה מעל IP) אשר אינן צריכות שידורים חוזרים. אפליקציות אלה יודעות להתמודד עם אובדן חבילות.

# TCP - המשך

מבנה TCP segment:

IP datagram

TCP segment

IP header (20 bytes)	TCP header (20 bytes)	TCP data
-------------------------	--------------------------	----------

0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Source port number														Destination port number									
Sequence number																							
Acknowledgment number																							
Header length				reserved				Flags				Window size											
TCP checksum														Urgent pointer									
Options (if any)																							
Data																							

# TCP - המשך

- **Destination port** ו- **Source port** מגדירים ביחד TCP connection.
- **Sequence number** – מספר הסידורי של הבית הראשון ב-segment.
- **Acknowledgment number** – המספר הסידורי של הבית הבא לו מצפה השולח.
- **Window** – מאפשר לתחנה המקבלת לציין את גודל החלון המרבי אתו היא מוכנה לעבוד.
- **Urgent pointer** – מאפשר להכניס "ערוץ בקרה" בתוך ערוץ המידע, ולתת לערוץ זה עדיפות ביחס לזמן העברתו לאפליקציה.

# TCP - המשך

• **Flags** מכיל 6 דגלים:

❖ **URG** – מצוין שהשדה urgent pointer תקף.

❖ **ACK** - מצוין שהשדה acknowledgement number תקף.

❖ **PSH** – מבקש להעביר את המידע ללא שהות לאפליקציה (אחרת – ההחלטה היא בידי TCP המקבל).

❖ **RST, SYN, FIN** – מציינים סגמנטים לבנייה, סיום והפסקת TCP connection.



# Application Layer

- שיכבה זו מכילה את כל הפרוטוקולים ברמה הגבוהה  
(high level protocols).

- מעל: UDP  
DNS, SNMP, TFTP, DHCP

- מעל: TCP  
telnet, ssh, rlogin, ftp, http, POP, SMTP

# Dynamic Host Configuration Protocol (DHCP)

- פרוטוקול ברמת האפליקציה המאפשר לתחנה להתחבר לרשת באופן "אוטומטי".
- התחנה מחפשת DHCP server מקומי, שיעניק לה את הפרמטרים הדרושים לתקשורת כגון:
  - ❖ כתובת IP:
  - השרת יכול להעניק כתובת IP לפרק זמן קצוב מראש, או לתמיד. במידת הצורך, התחנה תבקש הארכה של פרק הזמן שהוקצב לה. (הארכה של ה-"lease")
  - ❖ כתובת IP של ה-default gateway:
  - כדי שהתחנה תדע לאן לשלוח חבילות שאינן מיועדות למחשבים הנמצאים ברשת שלה.
  - ❖ כתובת ה-IP של שרת DNS:
  - כדי שניתן יהיה לתרגם שמות אתרים לכתובות.

# DHCP - המשך

- כאשר שרת DHCP מקבל בקשה, הוא בוחר אינפורמציה כתובת IP ממאגר כתובות אשר מוגדרות במסד הנתונים שלו, ומספק זאת למחשב הפונה (DHCP client).
- אם ה-client מקבל את ההצעה כתובת ה-IP ניתנת לו לפרק זמן מוגדר.

# DHCP - המשך

• DHCP משתמש בתהליך בעל 4 שלבים:

– **IP lease request**

ה-client מאתחל גירסא מצומצמת של TCP/IP ומפיץ בקשה (broadcast) למיקום של שרת DHCP ואינפורמציית כתובות IP.

– **IP lease offer**

כל שרתי ה-DHCP שברשותם כתובות IP תקפות, שולחים הצעה אל ה-client.

– **IP lease selection**

ה-client בוחר את ה-IP addressing information מההצעה הראשונה שהוא מקבל ומפיץ בקשה לקבל את כתובת ה-IP מההצעה.

– **IP acknowledgment**

שרת ה-DHCP אשר יצר את ההצעה, מגיב להודעה שהגיעה מה-client ואילו יתר שרתי ה-DHCP "מושכים" את הצעתם. אינפורמציית ה-IP משויכת ל-client, ונשלח אישור.

# DHCP -IP lease request

- בפעם הראשונה כשמאותחל מחשב, הוא שולח בקשה לשכירת כתובת IP ע"י הפצת הבקשה לכל שרתי ה-DHCP.
- כיוון של-client אין כתובת IP וכמו כן אין הוא יודע את כתובת ה-IP של שרת ה-DHCP הוא משתמש בכתובת 0.0.0.0 ככתובת מקור ובכתובת: 255.255.255.255 ככתובת היעד (limited broadcast).
- בקשת השכירות נשלחת כהודעת DHCPDISCOVER. הודעה זו מכילה את הכתובת הפיזית (כתובת ה-MAC) ושם המחשב של ה-client, כדי ששרתי ה-DHCP ידעו ממי הגיעה הבקשה.

# DHCP -IP lease offer

- כל שרתי ה-DHCP אשר מקבלים IP lease request, ויש להם קונפיגורציה תקפה עבור ה-client, מפיצים הצעה שמכילה את האינפורמציה הבאה: כתובת ה-MAC של ה-client, הצעה לכתובת IP, Subnet mask, אורך תקופת השכירות, כתובת ה-IP של השרת המציע.
- מתבצע שימוש ב-broadcast מכיוון של-client עדיין אין כתובת IP.
- ההצעה נשלחת כהודעת DHCPOFFER.
- שרת ה-DHCP שומר את כתובת ה-IP כדי שהיא לא תוצע ל-client אחר.
- ה-client בוחר את כתובת ה-IP מההצעה הראשונה אותה הוא מקבל.

## DHCP -IP lease offer - המשך

- לאחר שה-client קיבל הצעה לפחות משרת DHCP אחד, הוא שולח broadcast לכל שרתי ה-DHCP שהוא כבר בחר הצעה מסוימת.
- ה-broadcast נשלח כהודעת DHCPREQUEST ומכילה את כתובת ה-IP של השרת אשר הצעתו התקבלה.
- כל יתר שרתי ה-DHCP מושכים את הצעותיהם, ובכך מאפשרים את חלוקת הכתובות שלהם לבקשות אחרות.



## מבנה האינטרנט

- האינטרנט בנוי מאוסף של רשתות המחוברות ע"י נתבים.
- מבחינה אדמיניסטרטיבית האינטרנט מחולק ל- Autonomous Systems (AS).
- כל AS מנוהל ע"י ישות אדמיניסטרטיבית אחת, ומורכב מכמה רשתות.
- בשל גודל האינטרנט לא ניתן להריץ פרוטוקול ניתוב אחד בין כל הנתבים באינטרנט.
- לכן, מריצים פרוטוקול ניתוב עצמאי בכל אחד מ-N ה-ASs ברשת ופרוטוקול אחד (נוסף) בין כל ה-ASs.

# פרוטוקולי ניתוב באינטרנט

- RIP) Routing Information Protocol)  
מוצא את הניתוב ה"זול ביותר" מרשת המקור לרשת היעד.
- OSPF) Open Shortest Path First)  
מוצא את הניתוב ה"זול ביותר" מרשת המקור לרשת היעד.
- BGP) Boarder Gateway Protocol)  
מוצא את הניתוב ה"מתאים ביותר" מרשת המקור לרשת היעד.

# DNS – Domain Name System

- מערכת מבוזרת לשמירת/הפצת מידע המקשר בין שמות וכתובות IP.
- עובדת מעל UDP.
- מערכת השמות היא היררכית.
- מערכת יעילה: רוב הפניות נענות ע"י השרת המקומי.
- מערכת כללית: תומכת בכל מבנה היררכי.
- מערכת אמינה: עובדת גם כאשר חלק מהשרתים אינם פעילים.
- מאפשרת סוגים שונים של שאילתות.

# Firewall

- Filter אשר בודק ומסנן מסגרות.
- ברמת ה-Network יכול לעשות הרבה אך לפעמים נדרש ידע שאינו ברמה זו.
- ברמת ה-TCP, UDP :Transport - סינון לפי ports.
- לפעמים נדרש ידע ברמת האפליקציה (ftp).