CCNA 1
Introduction to Networks

1. **Nivelul Transport**
2. TCP
3. UDP

A. Funcții Layer 4

Funcția principală = realizarea unei **conexiuni logice** între hosturi:

- **Multiplexare** – identificarea conversațiilor individuale (ex. browsing, email, chat, joc online) - **porturi**
- **Segmentare** – împărțirea datelor de la nivelul aplicație în bucăți / segmente mai mici; avantaj – detecția optimă și corecția erorilor
- **Reliable connections** – sequence numbers, aknowledgments – detectarea segmentelor lipsă
- **Flow control – windowing** – dimensionarea cantității de informație trimisă funcție de capacitatea de prelucrare a destinatarului

B. Porturi

Sunt folosite de nivelul transport pentru identificarea și controlul conversației – multiplexarea sesiunilor între hosturi

Port sursă - emițător

Port destinație - receptor

Număr pe 16 biți => **0 -> 65535**

Tipuri de porturi:

- **Well-known** - **0 -> 1023**, alocate de IANA pentru aplicații folosite în mod curent în Internet

Ex.	http	80		https	443	dns	53
	ftp	20, 21		dhcp	67, 68	pop3	110
	smtp	25		telnet	23	ssh	22

- **Registered** - **1024 -> 49151**, alocate de IANA pentru aplicații proprietare (ex. SQL, Oracle, RADIUS)

- **Dynamically assigned / Ephemeral** – **49152 -> 65535**, alocate aleator de sistemul de operare ca porturi sursă pentru diverse sesiuni

C. Categorii de aplicații

Aplicațiile uzuale în rețelele convergente (date, voce, video pe aceeași infrastructură) au de regulă cerințe:

- Voce — delay mic, jitter 0
- Video — mai puțin sensibil la jitter (folosește buffer), BW alocată în mod constant
- Date — nu sunt sensibile la delay; trebuie reansamblate în ordine la recepție; detectare/corectare de erori

Aplicații:

- *Time-sensitive*
- *Error-sensitive*

La nivelul 4 – Transport există două protocoale complementare: **TCP** și **UDP**

1. Nivelul Transport
2. **TCP**
3. UDP

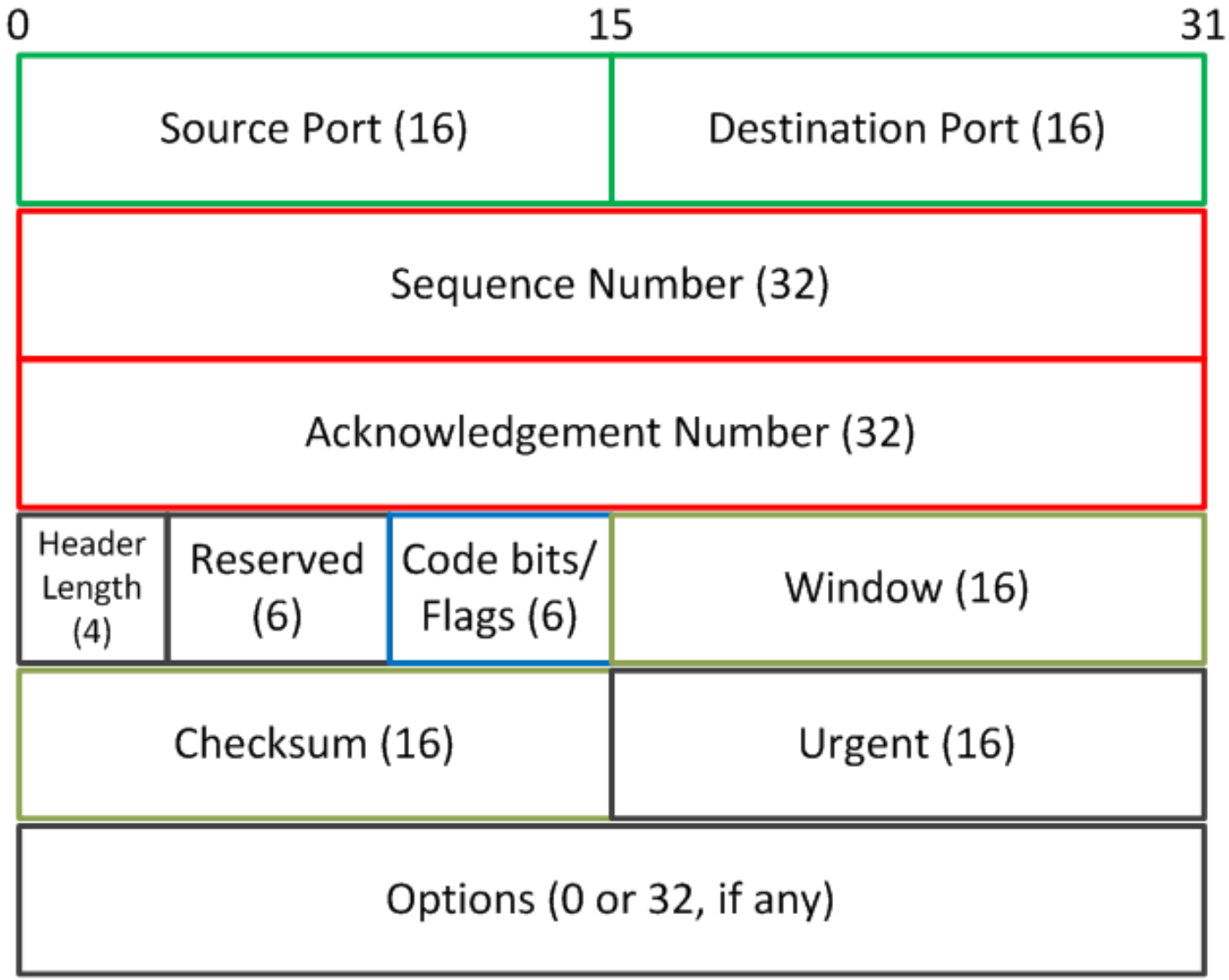
TCP = **T**ransmission **C**ontrol **P**rotocol

Se adresează tipurilor de date *sensibile la erori*, la modul în care sunt *reansamblate*.

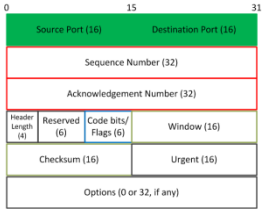
Caracteristici:

- Connection-oriented
- Reliable
- Error detection and correction
- Reordering of segments
- Windowing / Flow control

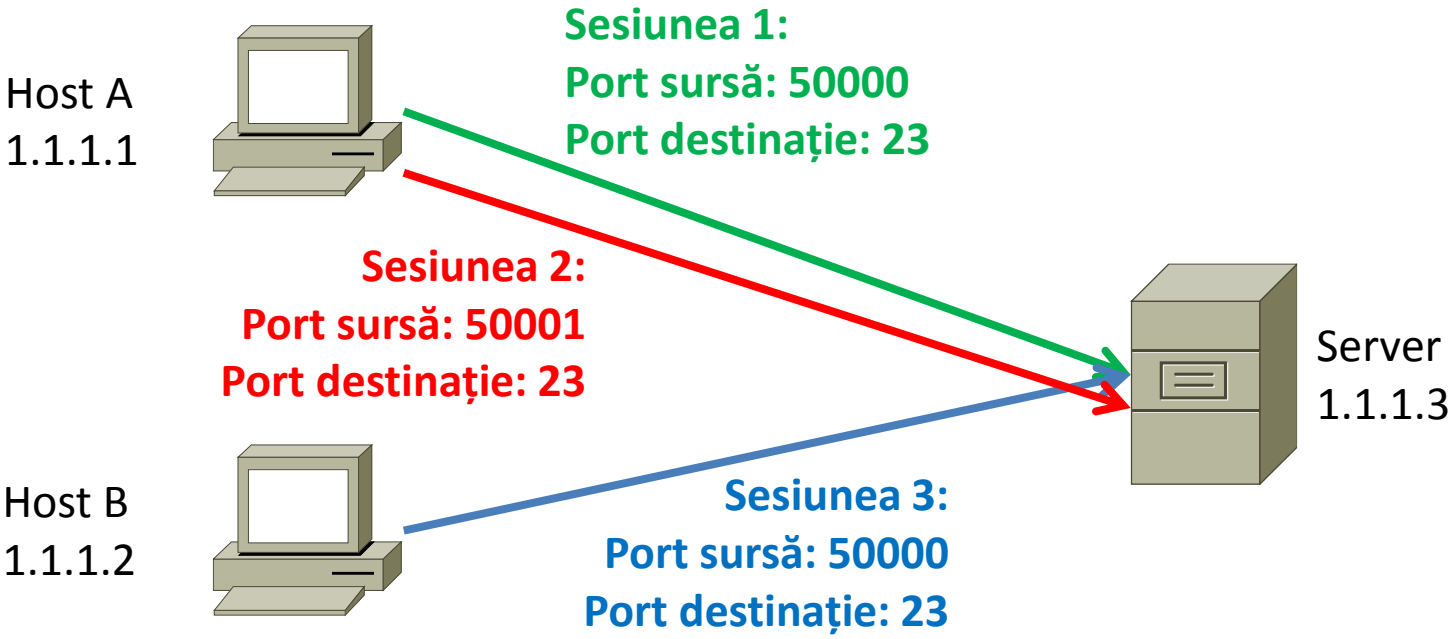
Header TCP


0
15
31
Source Port (16)
Destination Port (16)
Sequence Number (32)
Acknowledgement Number (32)
Header Length (4)
Reserved (6)
Code bits/ Flags (6)
Window (16)
Checksum (16)
Urgent (16)
Options (0 or 32, if any)

Port sursă / Port destinație:



- portul sursă se alocă *dinamic* (>1023)
- portul destinație este uzual un port *well-known* (instalate pe host odată cu instalarea stivei TCP/IP) sau registered (instalate odată cu aplicația)



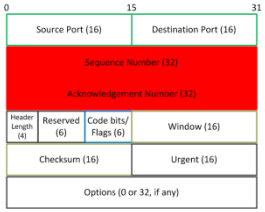
Indiferent de numărul de sesiuni, un host poate să le *diferențieze* folosind informația de la nivelul transport (porturi) și de la nivelul Internet (adresa IP)

IP:port = socket

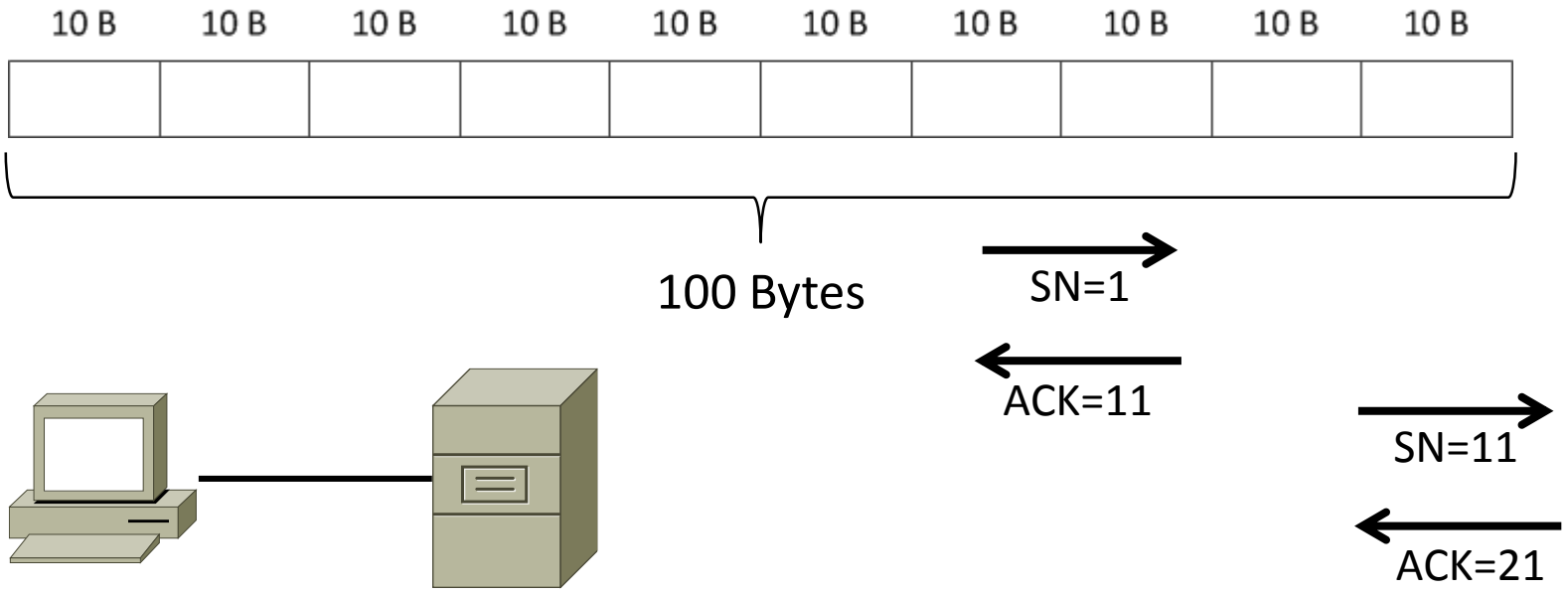
ex. **1.1.1.3:23**

Aplicație / Protocol	Port
FTP	20 (date), 21 (control)
Telnet	23
SSH	22
SMTP	25
POP3	110
HTTP	80
Secure HTTP (HTTPS)	443
DNS	53
DHCP	67 (server), 68 (client)

Sequence Number / Acknowledgement Number



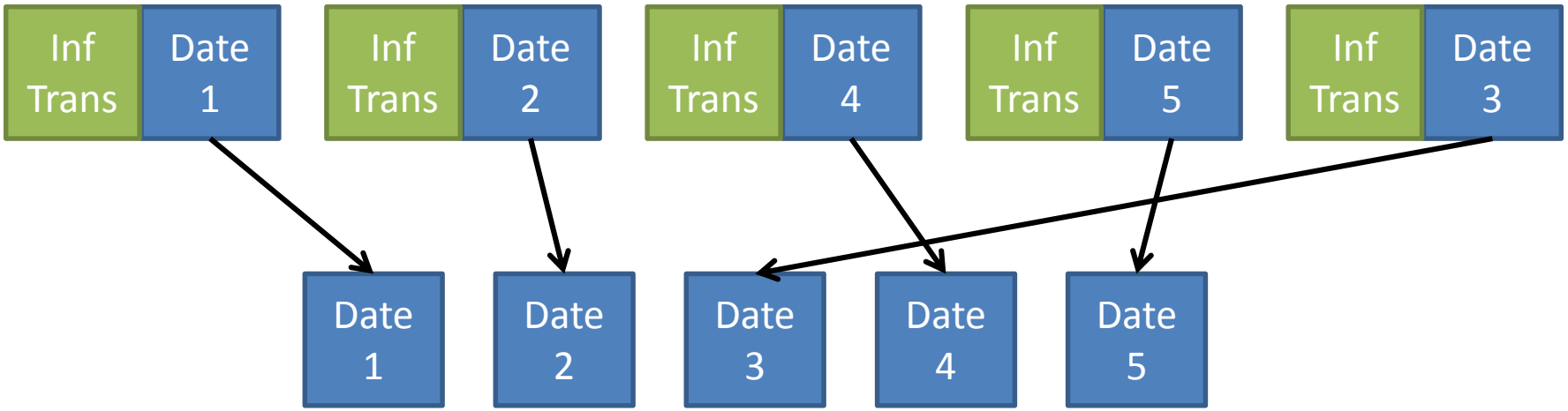
- numerotarea segmentelor ce provin dintr-un PDU mai mare
- sequence number = numărul *primului byte din segment*
- acknowledgement number = seq. number al *următorului segment* așteptat



Sequence Number / Acknowledgement Number (cont.)

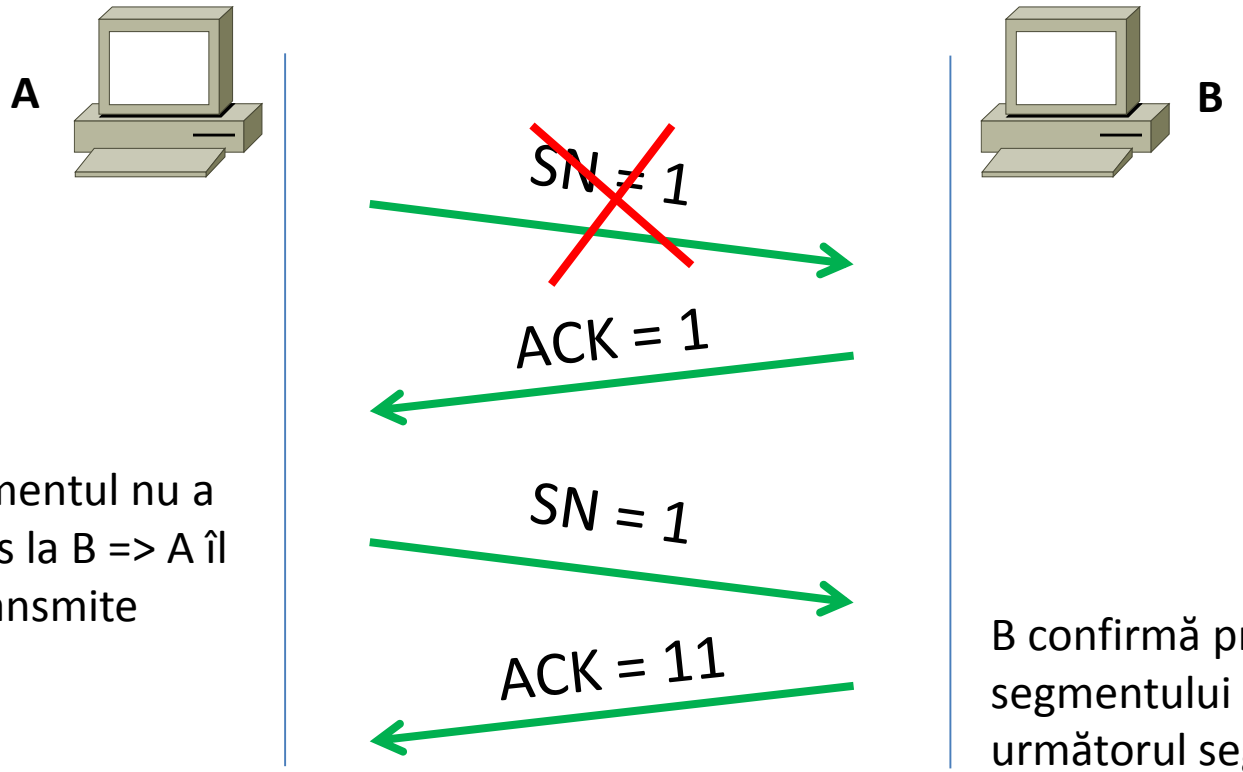
- reansamblarea datelor în *ordinea corectă*

(vs. în ordinea sosirii)

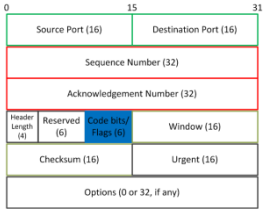


Sequence Number / Acknowledgement Number (cont.)

- confirmare de primire a datelor (ack. number)
- se poate solicita și retransmiterea segmentelor pierdute sau care prezintă erori



Code bits / Flags

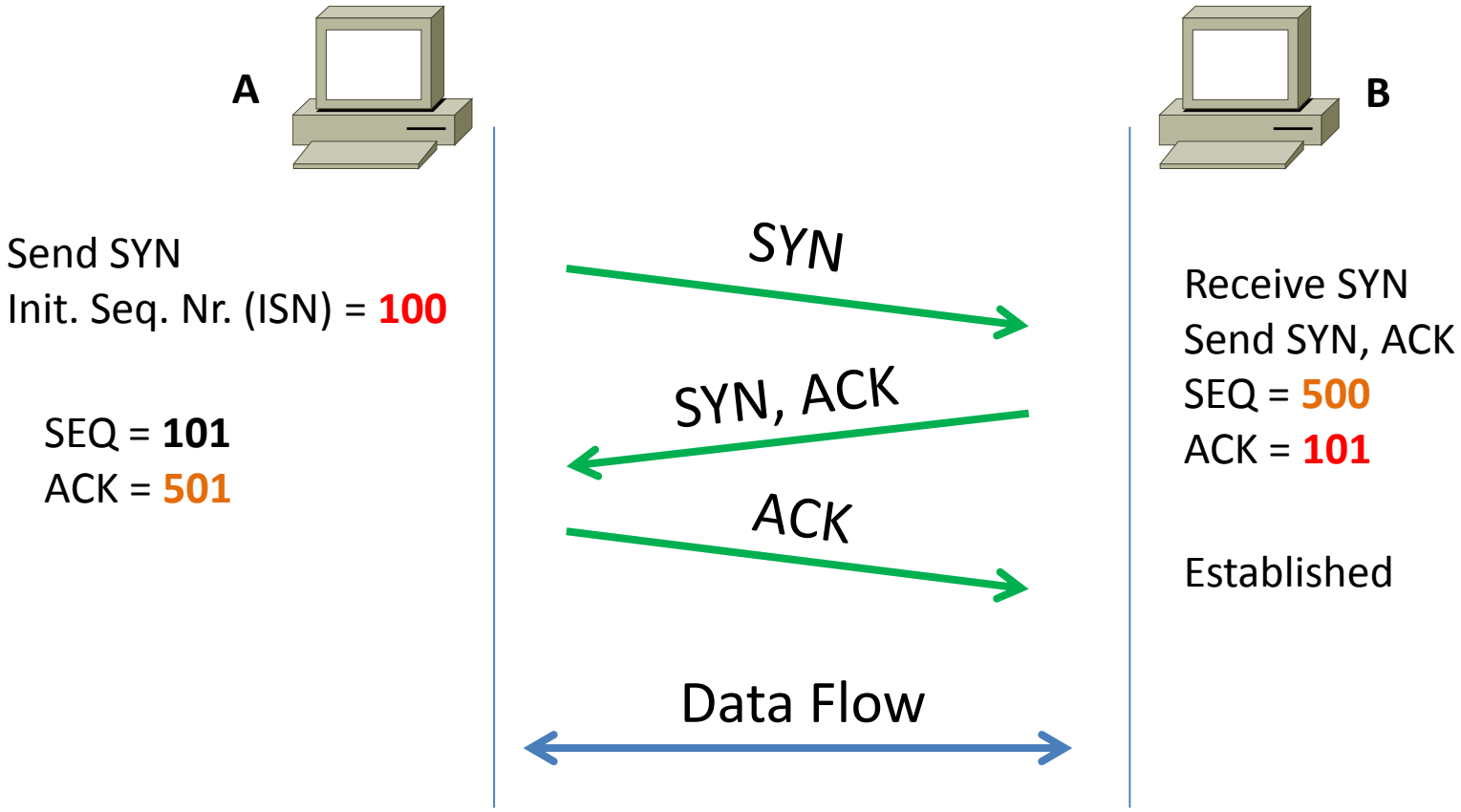


Flag = un bit ce poate avea valoare **0** – nesetat,

sau **1** – setat

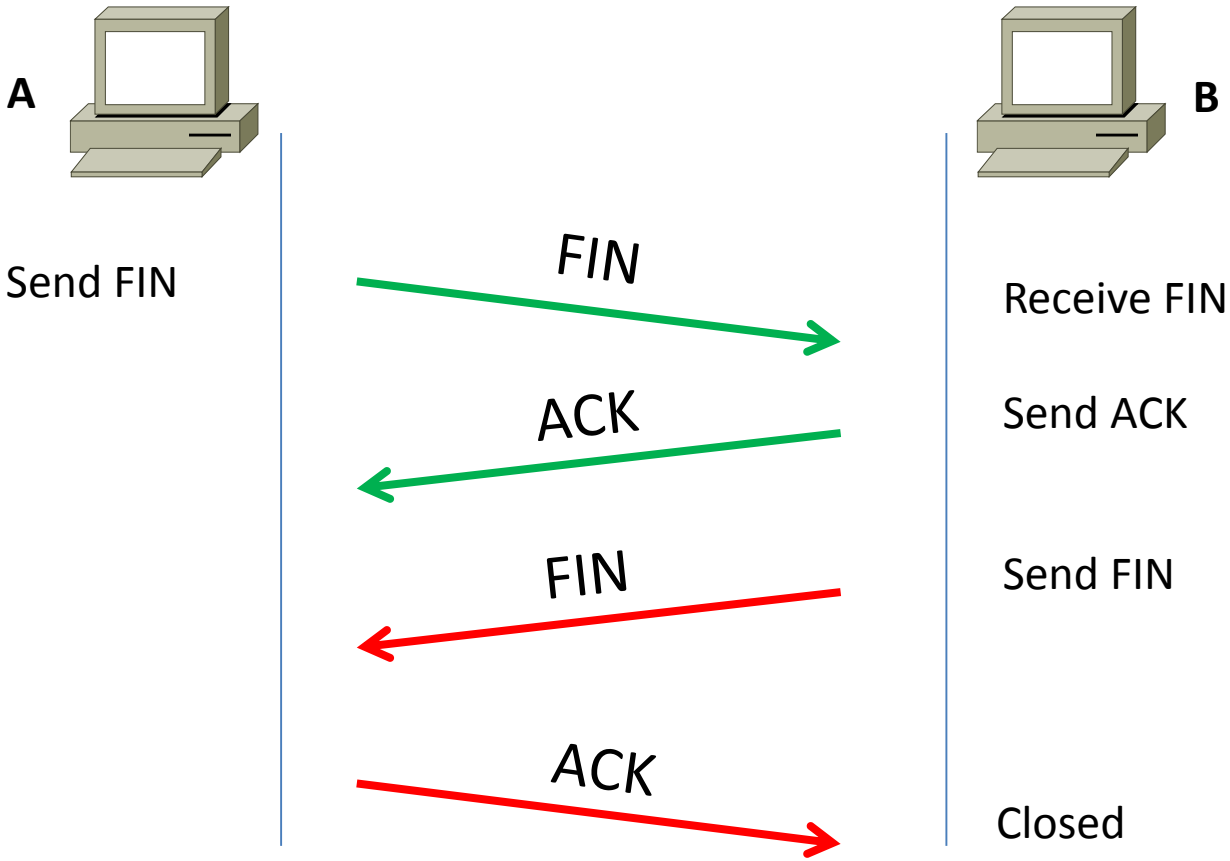
- **ACK** (Acknowledge) – folosit în realizarea conexiunii inițiale și ulterior în toate segmentele de acknowledge (expectational ack.).
- **SYN** (Synchronize) – folosit pentru deschiderea conexiunii.
- **RST** (Reset) – folosit în închiderea conexiunii.
- **FIN** (Finish) – folosit în închiderea conexiunii (graceful)

Stabilirea conexiunii logice - **Three-Way Handshake**

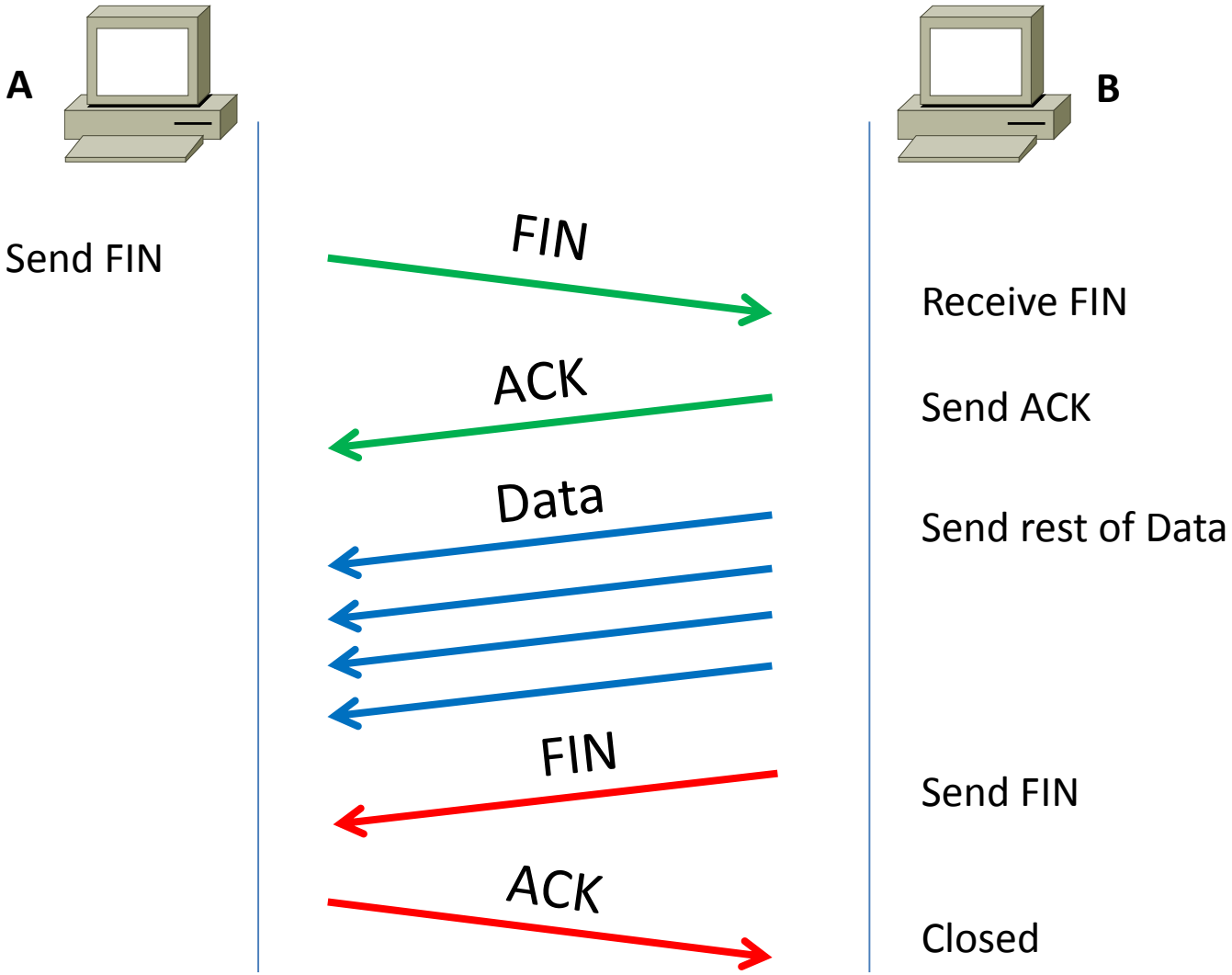


Închiderea conexiunii logice - graceful

1. ***Four-way handshake*** – B **nu mai are** date de transmis

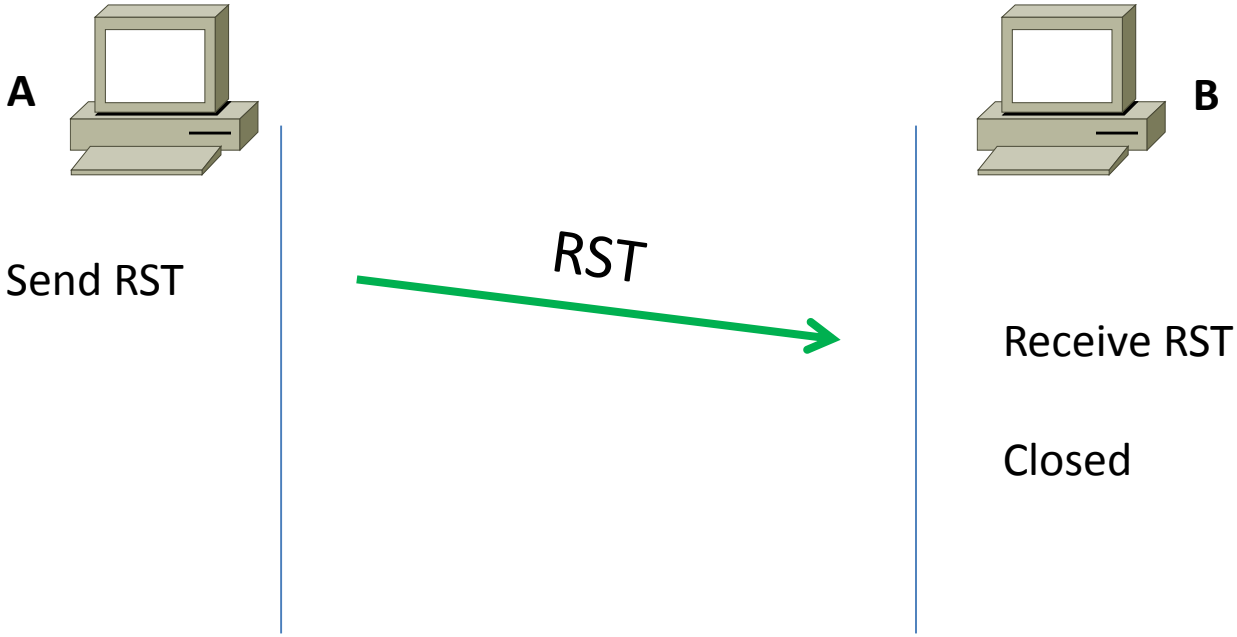


2. **Four-way handshake** – B **mai are** date de transmis



Închiderea conexiunii logice – non-graceful

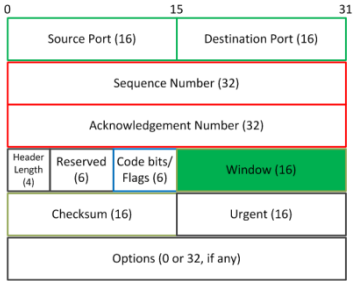
Hostul A închide conexiunea, fără să aștepte o confirmare de la B

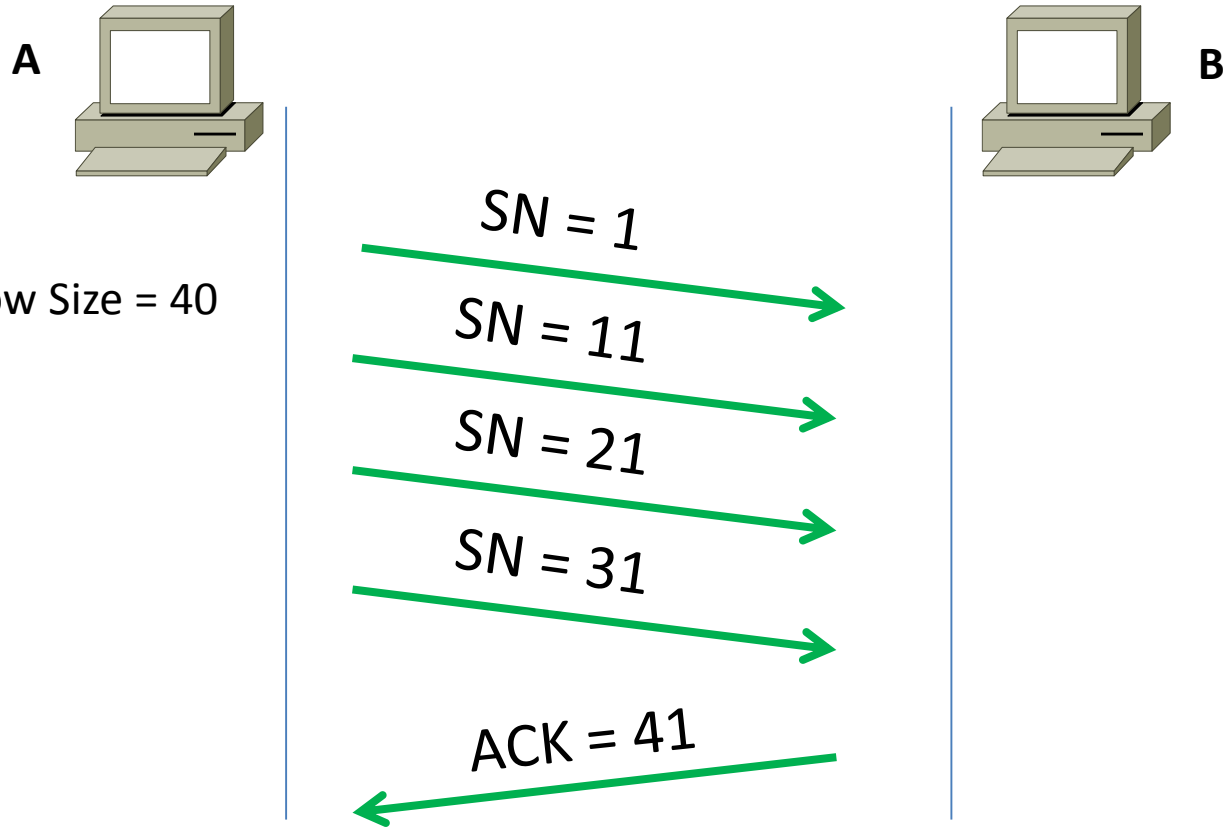


Flow Control – Window Size

- câți biți se trimit până se așteaptă un ACK

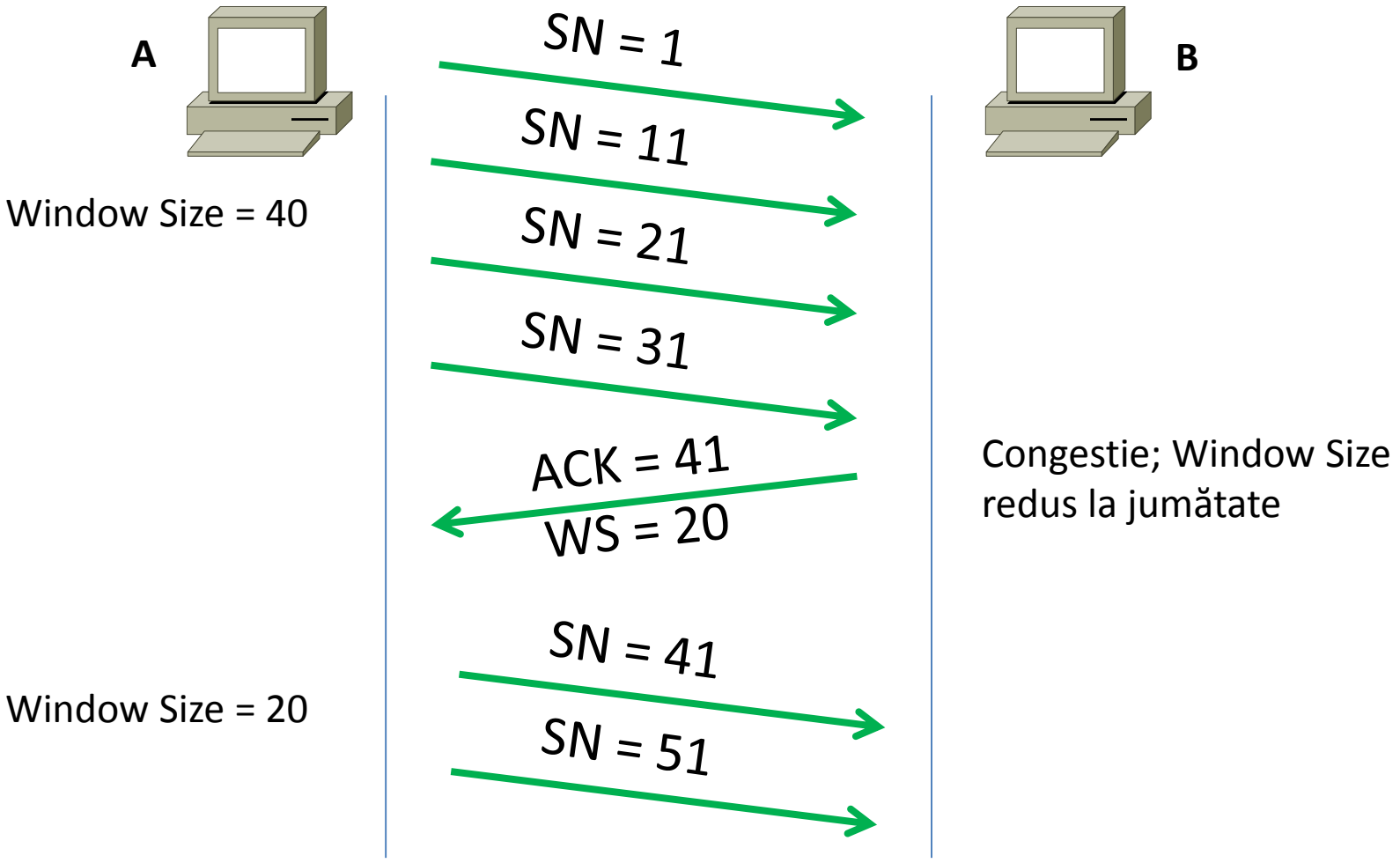
Ex. 100 B în 10 segmente de 10 B





Flow Control – Window Size (cont.)

- Flow control – în cazul congestiei se reduce window size (by default la 1/2)



1. Nivelul Transport
2. TCP
3. **UDP**

UDP = User Datagram Protocol

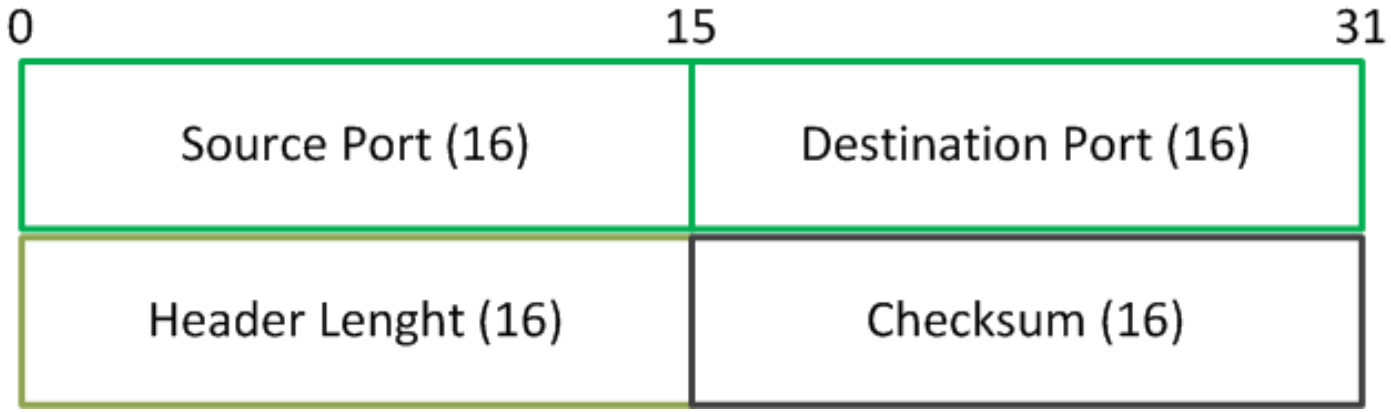
Se adresează tipurilor de date sensibile la delay (voce, video).

Datele sunt reansamblate în ordinea în care sosesc segmentele.

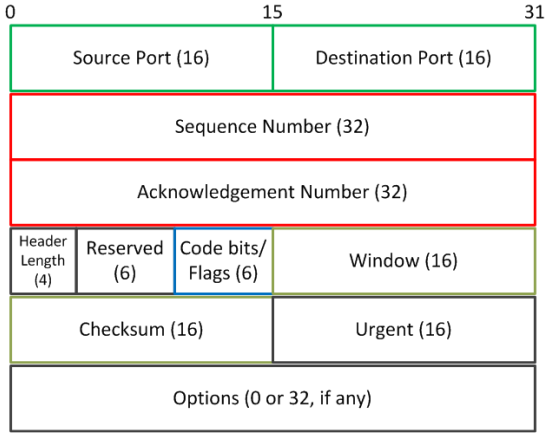
Caracteristici:

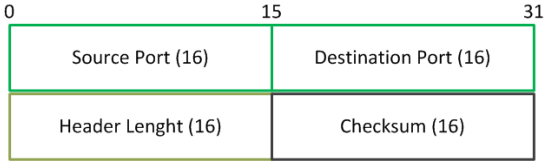
- Connectionless
- Unreliable
- No error detection / recovery
- Low overhead

Header UDP


0
15
31
Source Port (16)
Destination Port (16)
Header Lenght (16)
Checksum (16)

TCP vs. UDP





DNS (Zone transfer)	DNS (query)
FTP	VoIP (SIP)
POP3	Video Streaming (RTP)
SMTP	DHCP
HTTP(S)	RIP