

Rețele de Calculatoare

Nivelul Transport

Sumar al laboratorului

1

Încapsularea datelor
Antetul Nivelului Transport

2

Nivelul Transport
Generalități

3

Numere de porturi

4

Protocoalele
TCP
UDP

5

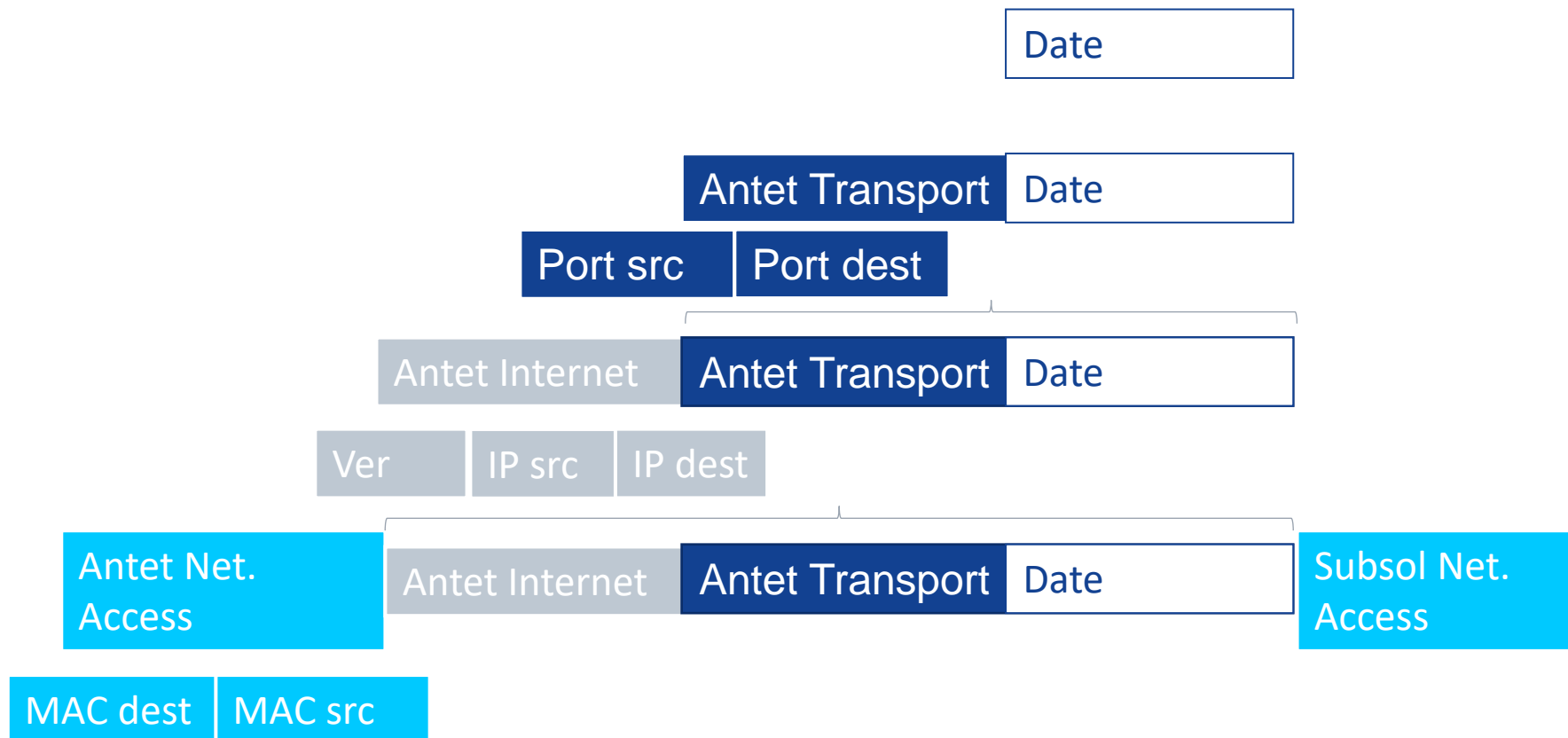
Sesiunile de comunicare
Procesul de windowing
Asamblare/Reasamblare
Unde folosim protocoalele



Procesul de încapsulare

Recapitulare

Aplicații	4
Transport	3
Internet	2
Network Access	1



Nivelul Transport

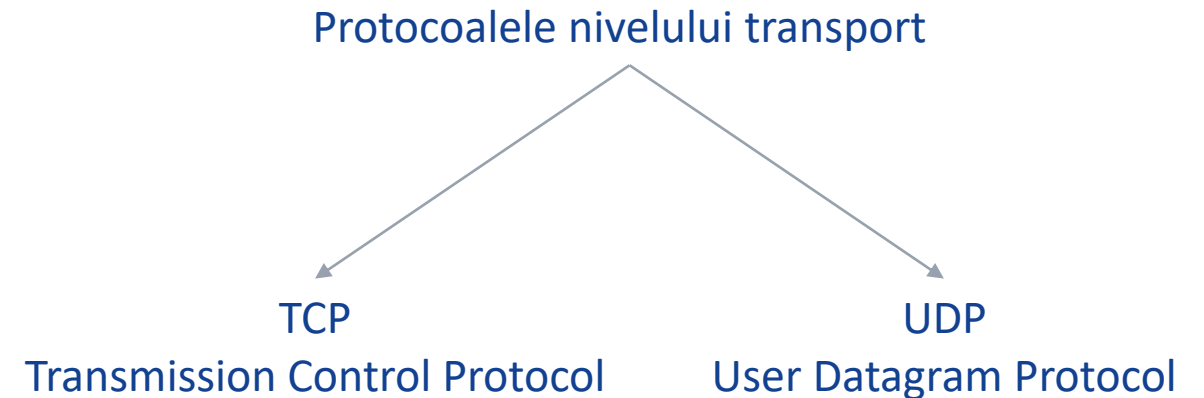
La ce este folositor?

Rolul nivelului Transport:

- Stabilirea comunicațiilor temporare între 2 aplicații
- Transportul datelor între ele
- Oferă servicii de multiplexare, fiabilitate, controlul fluxurilor

Multiplexarea conversațiilor:

- Împarte datele în segmente mai scurte;
- Etichetează segmentele în concordanță cu conversația



Nivelul Transport

Multiplexarea

E-mail

To

example@computer_networks.com

Cc

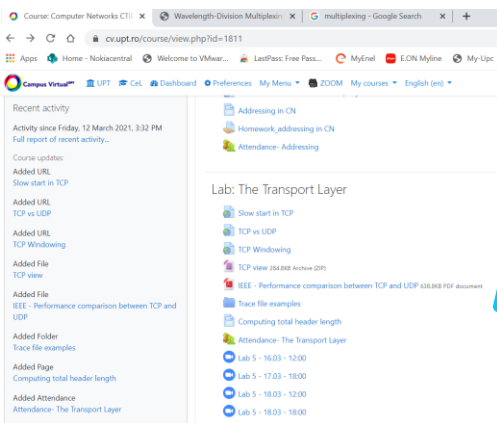
Subject

Example of email

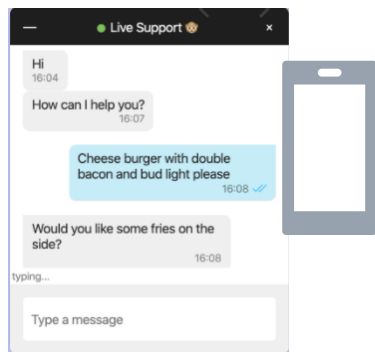


Multiplexarea este folosită între dispozitivele noastre și ruter-ul de gateway

HTML page

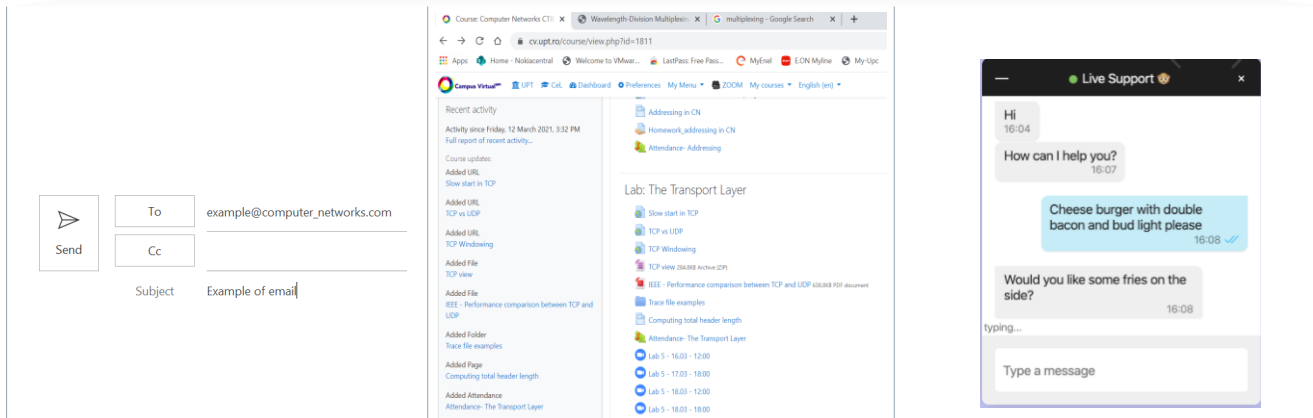


Web chat



Similar este folosită și de către diferite aplicații și nivelul Transport

Numere de porturi



- Grupuri de numere de porturi:
- Porturi bine cunoscute: 0-1023
 - Porturi rezervate: 1024-49151
 - Porturi dinamice: 49152-65535

În programare se folosește mult mai mult conceptul de **SOCKET**

Diverse aplicații	Electronic mail		HTML pages		Internet Chat	
Protocolul	POP3		HTTP		IM	
Transport	App port	Data	App port	Data	App port	Data
Numere de porturi	110		80		531	

ATENȚIE:
un socket = adresa IP+număr de port

Numere de porturi

Cum le vedem

Identificator unic
pentru fiecare
aplicație și sesiune
de comunicare

Portul sursă:
ales aleator de
către inițiator

Portul destinație:
folosit pentru
identificarea
aplicației la
recepție

Văzute ca și
pereche sursă-
destinație

Local Port	Remote Address	Remote Port
14766	localhost	52780
14766	localhost	52793
52563	131.228.2.175	https
52594	52.85.10.88	https
52598	172.217.20.226	https
52600	172.217.22.225	https
52601	172.217.23.1	https
52641	34.98.67.61	https

Protocolul UDP

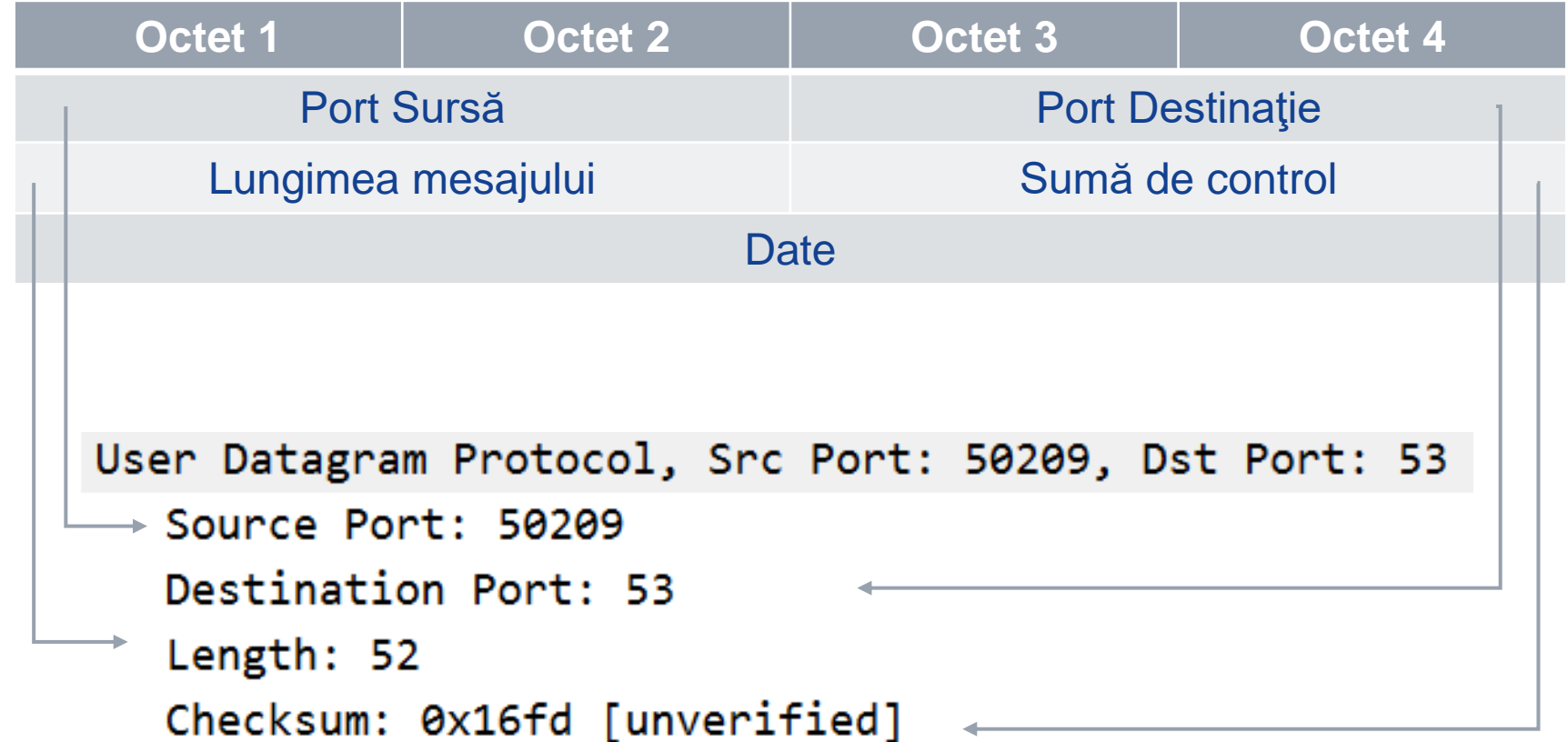
Antetul ptotocolului

Adaos scăzut de biți la date

Lungimea mesajului este formată din:

- Antet
- Date

Nu oferă fiabilitate prin structura sa



Protocolul TCP

Antetul protocolului

Transmission Control Protocol, Src Port: 50450, Dst Port: 443, Seq: 9256, Ack: 7701, Len: 0

Source Port: 50450

Destination Port: 443

[Stream index: 31]

[TCP Segment Len: 0]

Sequence number: 9256 (relative sequence number)

[Next sequence number: 9256 (relative sequence number)]

Acknowledgment number: 7701 (relative ack number)

1000 = Header Length: 32 bytes (8)

> Flags: 0x010 (ACK)

Window size value: 1025

[Calculated window size: 262400]

[Window size scaling factor: 256]

Checksum: 0x86c4 [unverified]

[Checksum Status: Unverified]

Urgent pointer: 0

> Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), SACK

> [SEQ/ACK analysis]

> [Timestamps]

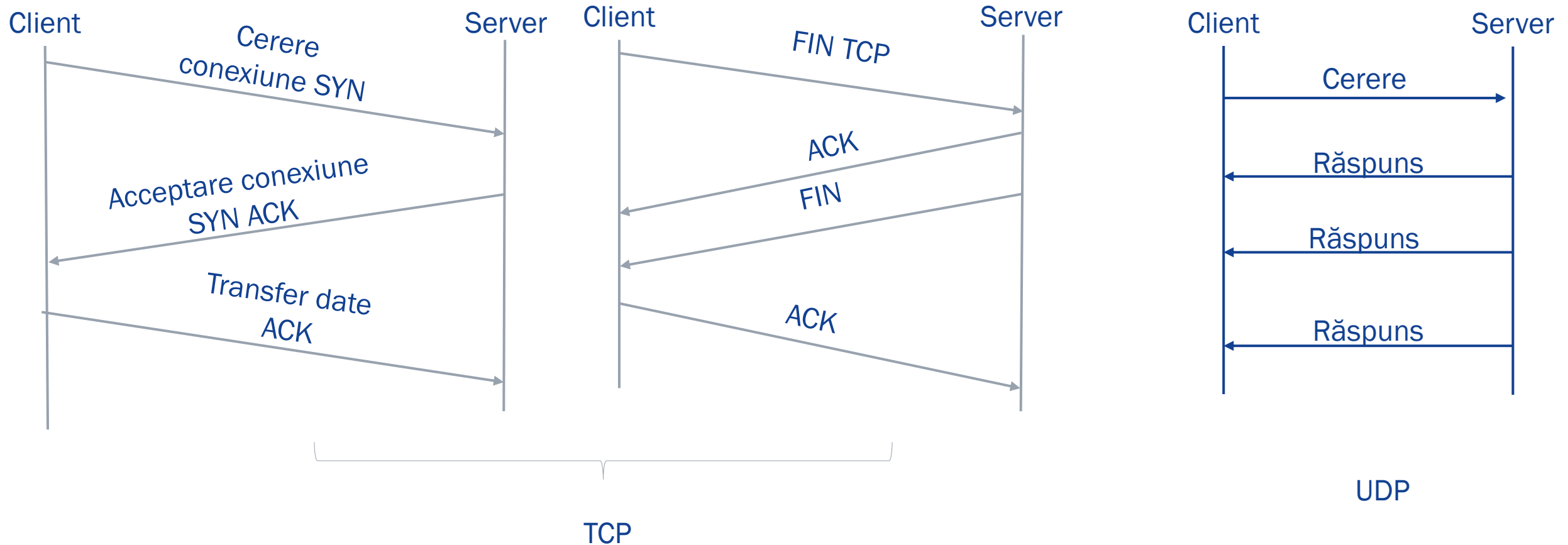
Octet 1	Octet 2	Octet 3	Octet 4
Port sursă		Port destinație	
Număr de secvență			
Număr de acknowledgement			
Lungimea antetului	Alți parametri	Dimensiunea ferestrei	
Sumă de control		Pointer de urgență	
		Opțiuni	
		Date	

Sesiunile de comunicare

Cum le deschidem și cum le închidem

Stabilirea conexiunii

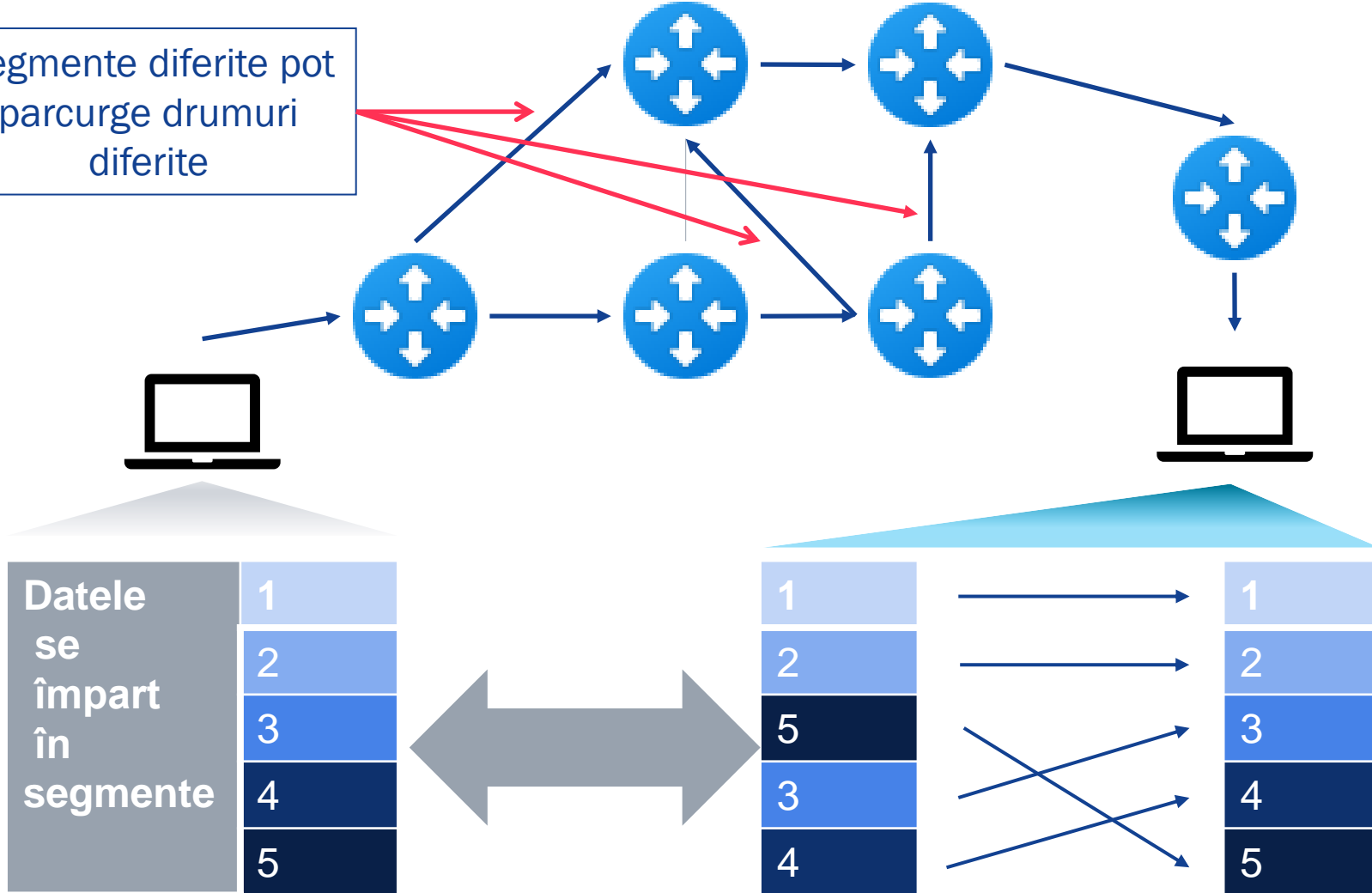
Încheierea conexiunii



Sesiunile de comunicare

Asamblarea și reasamblarea datelor

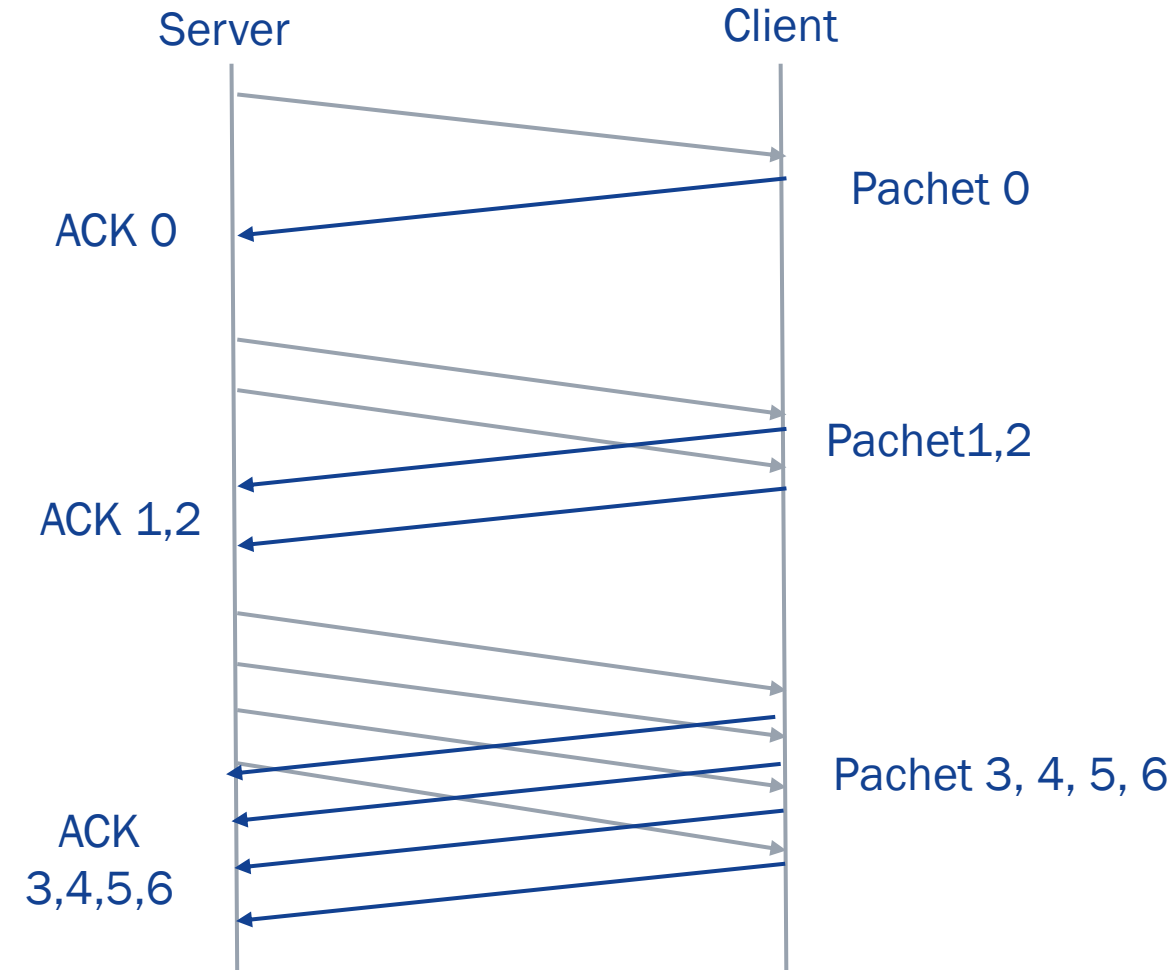
Segmente diferite pot
parcure drumuri
diferite



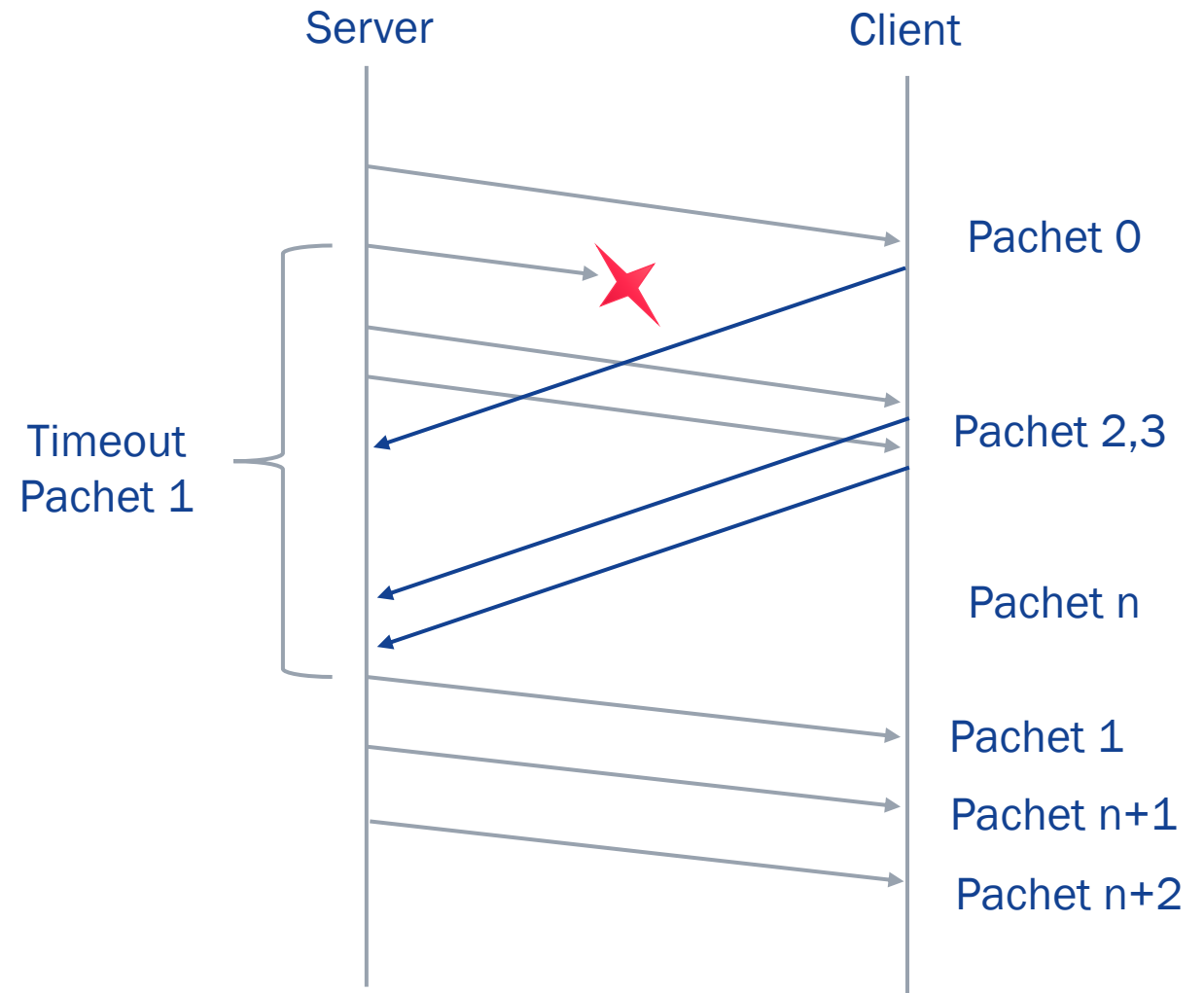
1. La începerea transmisiunii primul segment de date primește un număr de secvență (ISN)
2. Acest ISN se incrementează cu numărul de octeți transmiși
3. TCP-ul pune într-un buffer datele până ce toate segmentele au fost recepționate
4. La nivelul Aplicații se transmit datele complet recepționate și asamblate în mod corect

Sesiunile de comunicare

Procesul de windowing



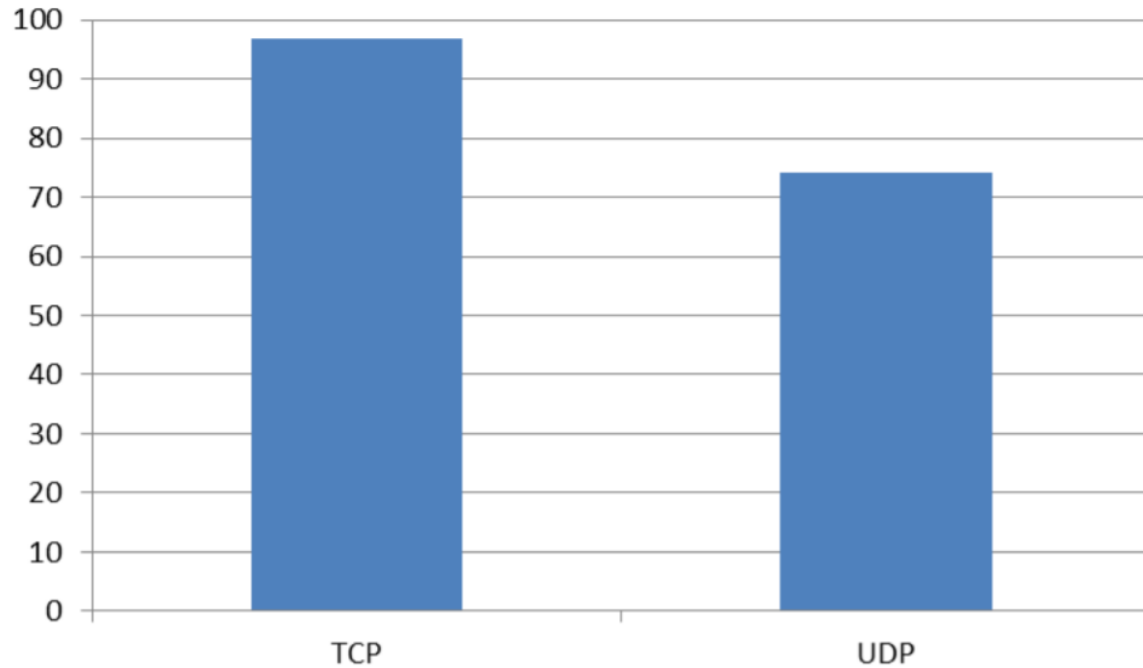
Transmisie fără erori



Transmisie cu erori

Cum alegem între ele?

TCP sau UDP



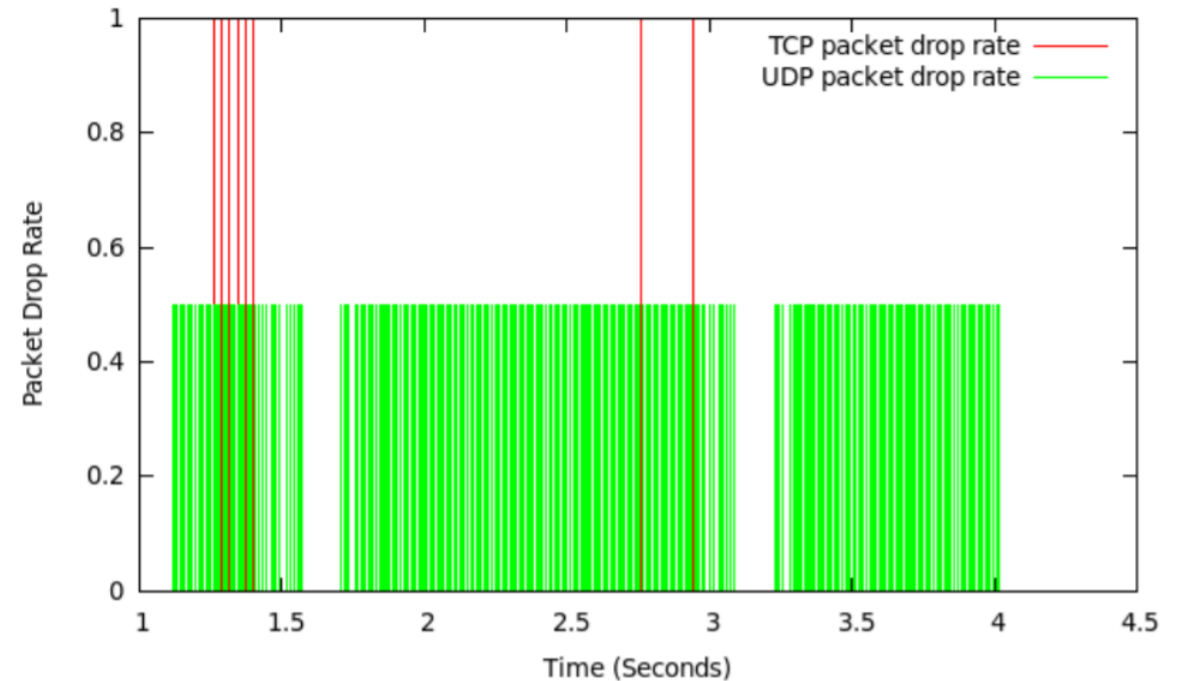
Conform articolului:

[Performance Comparison of Transport Layer Protocols](#)

By: Ali Hussein Wheeb University of Baghdad, Iraq

Rata de livrare a pachetelor

Rata de drop a pachetelor



Protocolul UDP

Unde îl folosim?

QUERY-RESPONSE PROTOCOLS



DNS

NTP

BROADCASTING SERVICES



MULTICAST

SERVICE DISCOVERY
SYSTEM

ROUTING PROTOCOL

TIME-SENSITIVE APPLICATIONS



IPTV



VOIP



ONLINE GAMES



Caracteristici:

- Simplu
- Rapid

Nu este orientat pe conexiune – nu oferă servicii de retransmisie complicate și resecvențiere

Fiabilitatea se asigură la nivel de aplicație

Sursa:
<https://finematics.com/udp-vs-tcp/>

Protocolul TCP

Unde îl folosim?

ONE OF THE MOST COMMONLY USED PROTOCOLS



ON THE INTERNET

HTTP



FTP



SSH



SMTP



COMPLEX
CONNECTION-ORIENTED
PROTOCOL



Caracteristici:

- Stabilește o sesiune de comunicare
- Asigură livrarea datelor
- Control al livrării
- Adaos crescut de biți la date

Sursa:
<https://finematics.com/udp-vs-tcp/>

Cum vedem porturile?

Windows

```
C:\Users\cmisici>netstat
```

Active Connections

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:7700	N-20L6PF1K8TWG:54606	ESTABLISHED
TCP	127.0.0.1:14766	N-20L6PF1K8TWG:49451	ESTABLISHED
TCP	127.0.0.1:14766	N-20L6PF1K8TWG:52409	TIME_WAIT
TCP	127.0.0.1:14766	N-20L6PF1K8TWG:52412	TIME_WAIT
TCP	127.0.0.1:14766	N-20L6PF1K8TWG:52413	TIME_WAIT
TCP	127.0.0.1:14766	N-20L6PF1K8TWG:52421	FIN_WAIT_2
TCP	127.0.0.1:49451	N-20L6PF1K8TWG:14766	ESTABLISHED
TCP	127.0.0.1:52421	N-20L6PF1K8TWG:14766	CLOSE_WAIT
TCP	127.0.0.1:54606	N-20L6PF1K8TWG:7700	ESTABLISHED
TCP	127.0.0.1:54921	N-20L6PF1K8TWG:54922	ESTABLISHED
TCP	127.0.0.1:54922	N-20L6PF1K8TWG:54921	ESTABLISHED
TCP	127.0.0.1:54923	N-20L6PF1K8TWG:54924	ESTABLISHED

Utilizând comanda netstat din cmd

Funcție: afișează conexiunile active prin:

1. Protocolul ce le folosește
2. Socket-ul folosit (aici vedem și portul)
3. Statusul conexiunii

Dacă dorim să vedem și PID-ul procesului adaugăm opțiunea -aon

Sursă și inspirație: <https://www.howtogeek.com/howto/28609/how-can-i-tell-what-is-listening-on-a-tcpip-port-in-windows/>

Cum vedem porturile? Windows

Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	State
chrome.exe	11924	TCP	n-2016pf1k8twg.nsn-intra.net	50381	wl-in-f188.1e100.net	5228	ESTABLISHED
chrome.exe	11924	TCP	n-2016pf1k8twg.nsn-intra.net	52426	199.232.18.49	https	ESTABLISHED
chrome.exe	11924	TCP	n-2016pf1k8twg.nsn-intra.net	52429	ec2-52-203-83-16...	https	ESTABLISHED
chrome.exe	11924	TCP	n-2016pf1k8twg.nsn-intra.net	52431	ec2-52-203-83-16...	https	ESTABLISHED

Utilizând utilitarul TCPview – atașat în CV

Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	State	Sent Packets	Sent Bytes
Teams.exe	24632	TCP	n-2016pf1k8twg.nsn-intra.net	50383	52.114.104.98	https	ESTABLISHED	5	893
Teams.exe	24500	TCP	n-2016pf1k8twg.nsn-intra.net	65426	52.113.205.20	https	ESTABLISHED	8	1,478
Teams.exe	24500	UDP	N-20L6PF1K8TWG	53414	*	*			
Teams.exe	24500	UDP	N-20L6PF1K8TWG	60227	*	*			

Cum vedem porturile?

Linux

```
admin@tecmint ~ $ sudo netstat -lntup
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:80             0.0.0.0:*               LISTEN      1423/nginx -g daemo
tcp        0      0 127.0.1.1:53           0.0.0.0:*               LISTEN      2992/dnsmasq
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      1409/sshd
tcp        0      0 127.0.0.1:631          0.0.0.0:*               LISTEN      2738/cupsd
tcp        0      0 0.0.0.0:443            0.0.0.0:*               LISTEN      1423/nginx -g daemo
tcp6       0      0 :::80                  :::*                    LISTEN      1423/nginx -g daemo
tcp6       0      0 :::22                  :::*                    LISTEN      1409/sshd
tcp6       0      0 :::1:631               :::*                    LISTEN      2738/cupsd
tcp6       0      0 :::443                  :::*                    LISTEN      1423/nginx -g daemo
udp        0      0 0.0.0.0:631            0.0.0.0:*               2740/cups-browsed
udp        0      0 0.0.0.0:5353           0.0.0.0:*               1022/avahi-daemon:
udp        0      0 0.0.0.0:36390          0.0.0.0:*               2992/dnsmasq
udp        0      0 0.0.0.0:59072          0.0.0.0:*               1022/avahi-daemon:
udp        0      0 127.0.1.1:53           0.0.0.0:*               2992/dnsmasq
udp        0      0 0.0.0.0:68             0.0.0.0:*               2982/dhclient
udp        0      0 192.168.43.31:123      0.0.0.0:*               1465/ntpd
udp        0      0 127.0.0.1:123          0.0.0.0:*               1465/ntpd
udp        0      0 0.0.0.0:123            0.0.0.0:*               1465/ntpd
udp6       0      0 :::43740               :::*                    1022/avahi-daemon:
udp6       0      0 :::5353                 :::*                    1022/avahi-daemon:
udp6       0      0 fe80::dd8c:3d40:817:123 :::*                    1465/ntpd
udp6       0      0 :::1:123               :::*                    1465/ntpd
udp6       0      0 :::123                 :::*                    1465/ntpd
admin@tecmint ~ $
```

Utilizând comanda netstat din terminal

Funcție: afișează conexiunile active prin:

1. Protocolul ce le folosește
2. Socket-ul folosit (aici vedem și portul)
3. Statusul conexiunii

Sursă și inspirație: <https://www.tecmint.com/find-listening-ports-linux/>



That's all for today, see you next time!

Nu uitați de tema de casă
termen de predare până în data de 25.11.2023.