Aplicații Multimedia

Laborator 2

Rețele de calculatoare. Protocolul TCP și UDP

Cuprins

Rețele de calculatoare

O rețea de calculatoare reprezintă o colecție de dispozitive interconectate, capabile să comunice între ele, efectuând un schimb de date. Aceasta este o definiție generală, în realitate comunicația printr-o rețea fiind un proces complex, ce îmbină diferite elemente, de la componente fizice precum cabluri, până la elemente logice și soluții software. Pentru a le reprezenta pe toate acestea, se poate folosi un model consacrat, și anume stiva TCP/IP

Nivelul Aplicație	Telnet, FTP, HTTP, SNMP
Nivelul Transport	TCP/UDP
Nivelul Rețea	IP, ICMP, Protocoale de rutare
Nivelul Fizic	Interfețe de rețea, Drivere

Fiecare nivel din stiva TCP/IP are o funcție specifică, după cum urmează:

- ***Nivelul Fizic*** definește modul în care un dispozitiv se va conecta fizic la rețea (Ethernet, Fibră optică, etc.). La acest nivel se folosește drept identificator adresa MAC (Media Access Control) care este asociată unei interfețe fizice de pe dispozitiv.
- ***Nivelul Rețea*** este cel în cadrul căruia se realizează logica de conexiune între componentele rețelei, definind căile pe care pachetele le pot urma pentru a ajunge de la o sursă la o destinație. La acest nivel se folosește drept identificator adresa IP (Internet Protocol), reprezentată pe 32 de biți grupați în 4 octeți (IPv4).
- ***Nivelul Transport*** este cel în cadrul căruia pachetele de date sunt construite, ceea ce presupune încapsularea datelor în mod specific protocolului utilizat.
- ***Nivelul Aplicație*** este responsabil de gestionarea și coordonarea schimbului de date, luând în calcul și formatul acceptat de aplicațiile care participă la comunicație. De exemplu, HTTP este o aplicație care rulează la acest nivel din stiva TCP/IP, în timp ce un browser web este o aplicație software care interacționează cu aplicația HTTP.

Internet Protocol (IP)

Adresele IP se pot împărți în trei clase principale în funcție de valoare primului octet după cum se poate observa în tabelul de mai jos.

Clasa	***Interval***	***Intervale private***
A	0-127	10.0.0.0 – 10.255.255.255
B	128-191	172.16.0.0 – 172.31.255.255
C	192-223	192.168.0.0 – 192.168.255.255

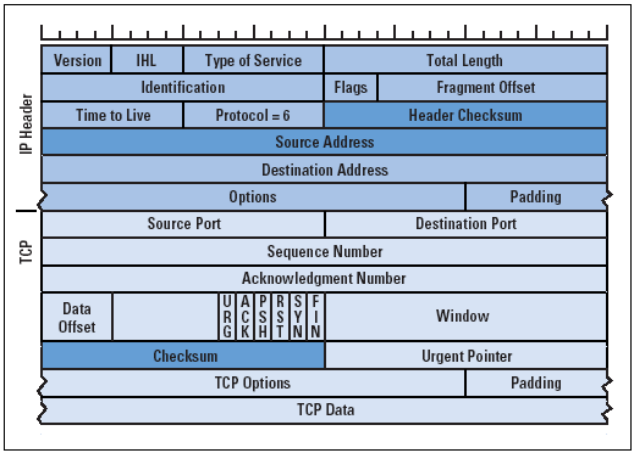
Intervalele private sunt cele care pot fi folosite pentru a crea adrese locale. Cu aceste adrese IP nu se iese direct în Internet, deoarece nu sunt unice (*n* companii diferite pot folosi simultan

rețeaua identificată prin 10.0.0.0). Pentru a ieși în Internet dintr-o rețea privată este nevoie de o metodă de mapare numită NAT (Network Address Translation) pentru a putea realiza corespondența dintre adresele private și cele publice.

Pe lângă aceste intervale private, un alt element special poate fi considerat a fi adresa de loopback (127.0.0.1 fiind cea mai folosită pentru IPv4). Adresa de loopback este cea prin care un dispozitiv poate să comunice cu el însuși, aceasta mai purtând și denumirea de ***localhost***.

Transmission Control Protocol (TCP)

TCP este un protocol orientat pe conexiune, ceea ce presupune asigurarea permanentă a faptului că o conexiune este pornită și funcționează, atât timp cât există un schimb de mesaje între două aplicații. TCP determină modul în care datele sunt fragmentate pentru a forma pachete ce pot fi trimise prin rețea. TCP lucrează împreună cu Internet Protocol pentru a defini un format al pachetelor și modul în care datele sunt încapsulate. Un astfel de pachet este format din metadate și conținutul concret al pachetului, iar structura este reprezentată în imaginea următoare:

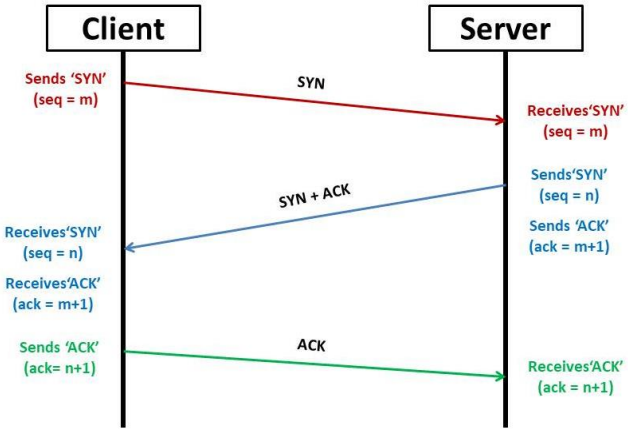


Structura pachetelor TCP/IP [sursă]

Metadatele vor oferi atât informații despre datele transmise (tipul de pachet, lungimea datelor, tipul datelor) cât și cu privire la sursa și destinația pachetului, dar și elemente de verificare precum sumele de control, pentru a putea verifica integritatea informațiilor. Pentru mai multe detalii privind fiecare câmp din structura pachetelor TCP/IP se poate accesa documentația de la link-ul: http://books.gigatux.nl/mirror/securitytools/ddu/ch06lev1sec3.html.

Pachetele pot avea un conținut diferit în funcție de ce protocol reprezintă. Astfel vom avea pachete de date și pachete de control. De exemplu protocolul ICMP (Internet Control Message Protocol) folosește doar pachete de control pentru a semnaliza și diagnostica probleme din rețea. Spre deosebire de acesta TCP folosește atât pachete de control cât și pachete de date. Pachetele de control sunt de obicei cele de ACK și SYN/ACK, tipul lor fiind specificat în

header-ul pachetului prin intermediul flag-urilor active. Aceste pachete sunt utilizate în cadrul operațiunii de Three Way Handshake, procedură utilizată pentru a stabili conexiunea dintre două dispozitive ce comunică.



Figură 1. Procedură 3-Way Handshake [sursă]

Pentru detalii suplimentare puteți accesa următorul articol: https://docs.microsoft.com/en-us/troubleshoot/windows-server/networking/three-way-handshake-via-tcpip, sau următorul video: https://youtu.be/F27PLin3TV0.

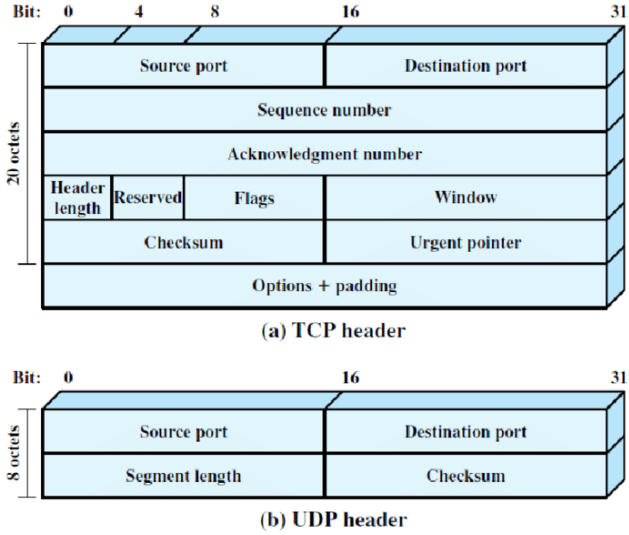
User Datagram Protocol (UDP)

UDP este un protocol ce poate fi utilizat pentru conexiuni care tolerează pierderi de pachete, care au cerințe de întârziere mici (cum ar fi transmisiile live/ streaming) sau care permit primirea pachetelor într-o ordine diferită față de cea în care au fost trimise. Astfel UDP este esențial pentru comunicații precum: VoIP (voice over IP), redări de video/audio.

Spre deosebire de UDP, TCP cere ca pachetele să fie primite în ordinea în care au fost trimise și retrimite pachetele care s-au pierdut. În plus TCP așteaptă până când toate pachetele dintr-un mesaj au fost primite, ceea ce îl face nepotrivit pentru aplicațiile mai sus menționate.

Luând un exemplu concret: considerând un eveniment transmis live, dacă s-ar folosi TCP s-ar produce întârzieri mari, cu toate că integritatea mesajelor este păstrată.

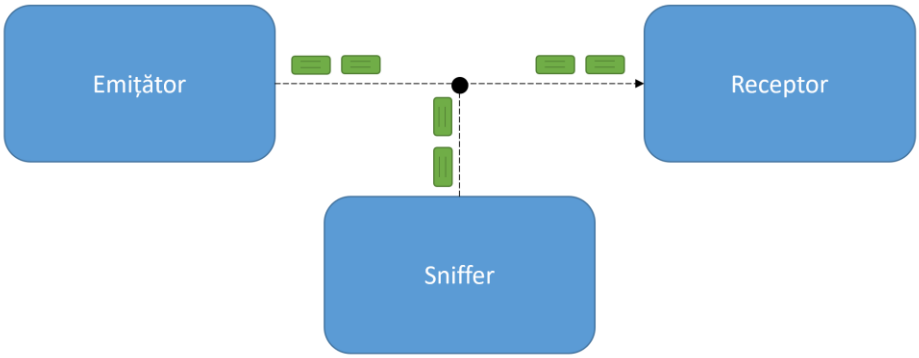
În plus, structura unui pachet UDP este mult mai simplă și dimensiunea este mai redusă față de TCP. În cazul ambelor protocoale header-ul de IP este identic, iar diferența apare la header-ul specific UDP.



Figură 2. Header TCP vs. Header UDP [sursă]

Wireshark

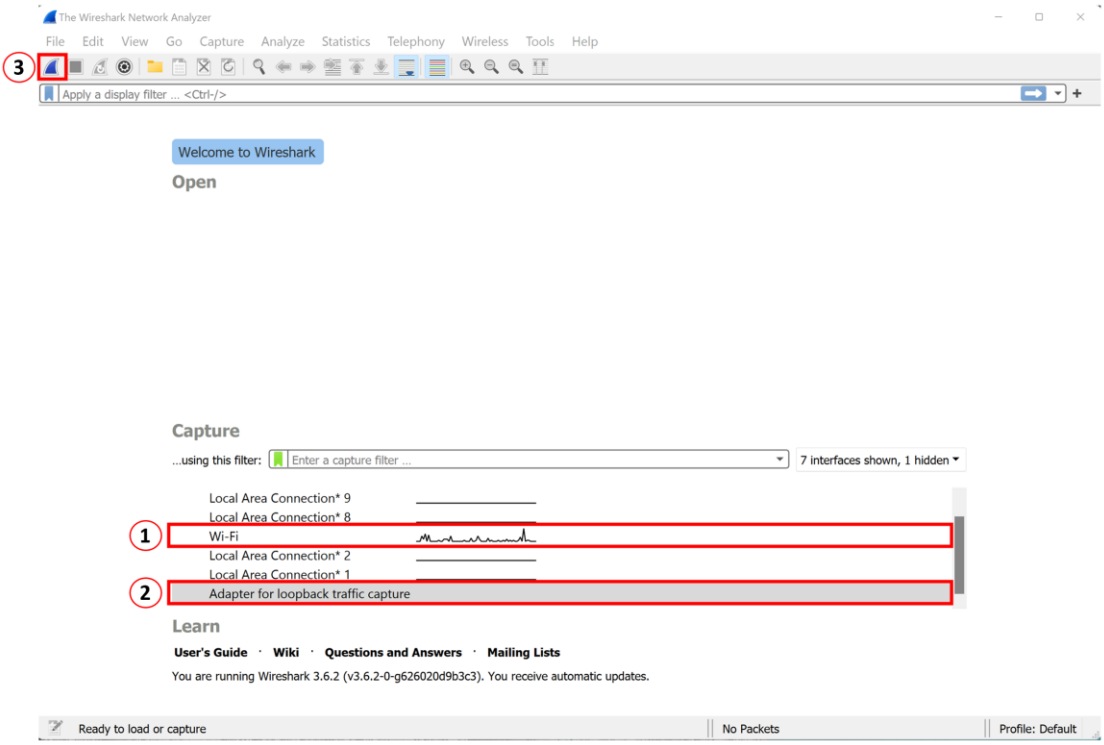
Wireshark este o aplicație utilizată pentru analiza traficului de rețea, acesta realizând o captură a traficului vehiculat, fără să interfereze cu acesta. Cu alte cuvinte, Wireshark realizează o copie a traficului de rețea, fiind un monitor pasiv. Acest tip de aplicație poartă denumirea de sniffer, acționând ca în imaginea de mai jos:



Sniffer al unei conexiuni

Wireshark poate fi descărcat de la adresa: https://www.wireshark.org/#download. Acesta se bazează pe biblioteca Libpcap, despre care puteți citi mai multe aici.

După deschiderea aplicației, se poate observa următorul ecran:



Pentru a analiza traficul, trebuie selectată o interfață care să fie analizată, în cazul de mai sus fiind evidențiate două interfețe importante pentru acest laborator:

1. Wi-Fi (în cazul conexiunilor wireless) sau Ethernet (pentru conexiunile wired) – se va analiza tot traficul către surse externe (pagini web, servere, etc.)
2. Adapter for loopback traffic capture – se va analiza traficul local, pe adresa de loopback (de exemplu, în cazul unui server și unui client pe aceeași mașină)

Începerea capturării traficului, după alegerea interfeței, se va face prin apăsarea butonului 3.

După capturarea unor pachete, se pot aplica filtre, care pot specifica protocolul căutat (tcp, udp, icmp), un port anume pentru un protocol (udp.port == 65784) sau o adresă IP după care se caută (ip.addr == 192.168.0.45). În figura de mai jos, de exemplu, se caută toate pachetele TCP de pe portul 22.