

C4

Rețele de calculatoare

Sergiu Nisioi
sergiu.nisioi@unibuc.ro

Anul II, FMI, UniBuc, 2021-2022

De data trecută

HTTP/1.1

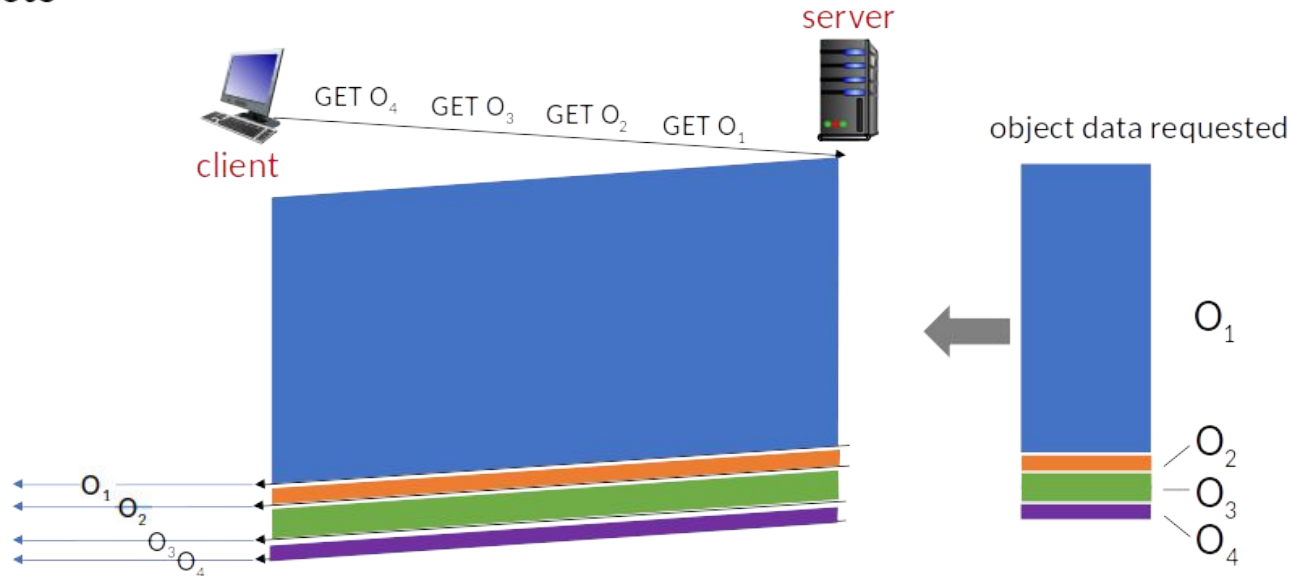
- **FCFS first-come-first-served scheduling**, serverul răspunde la obiectele cerute în ordine
- obiecte mai mici riscă să stea blocate până la transmisia obiectelor mari, **head-of-line blocking (HOL)**
- retransmiterea segmentelor de TCP blochează transmisia tuturor obiectelor

HTTP/2 [[RFC 7540](#), 2015]

- obiectele cerute sunt returnate în funcție de prioritatea specificată de client (NU FCFS)
- posibilitatea de a trimite (push) obiecte care nu au fost cerute în prealabil către client
- divizarea obiectelor în frame-uri

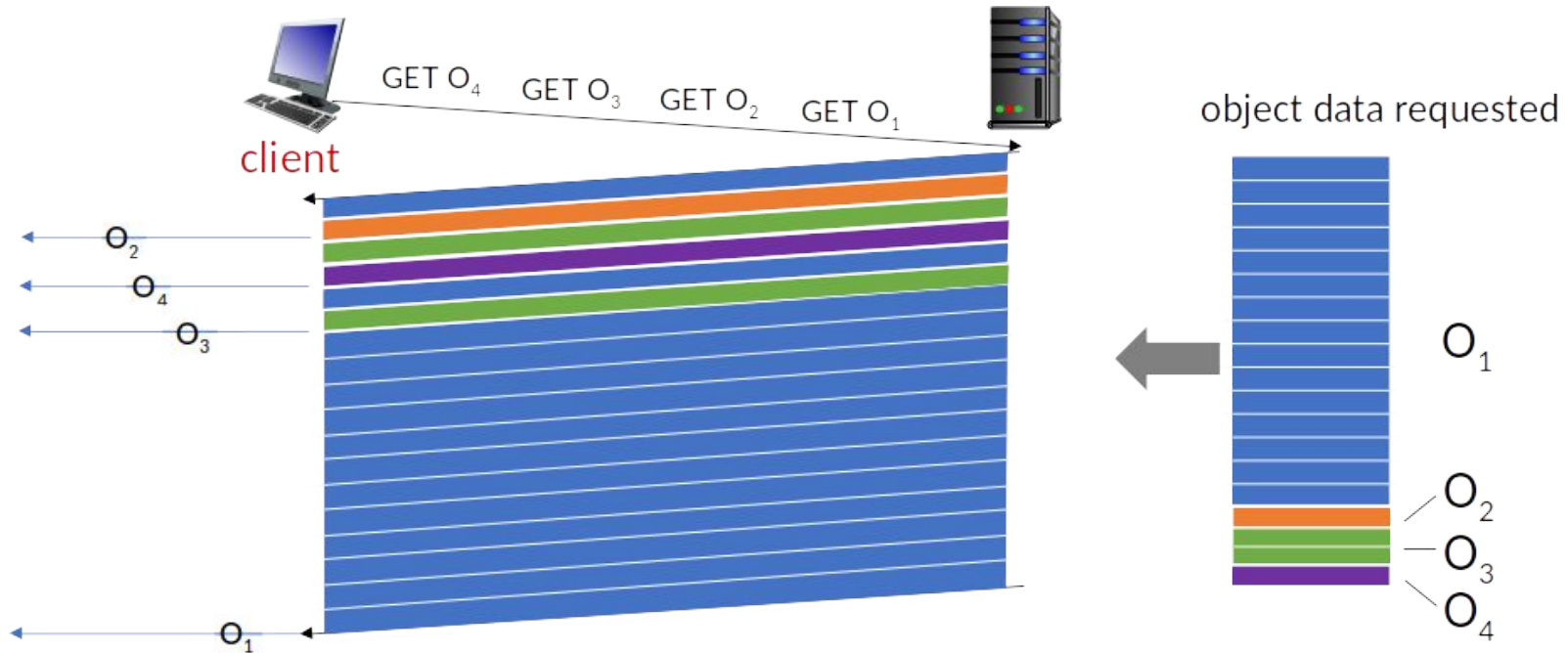
De data trecută: HOL blocking

HTTP 1.1: client requests 1 large object (e.g., video file) and 3 smaller objects



objects delivered in order requested: O₂, O₃, O₄ wait behind O₁

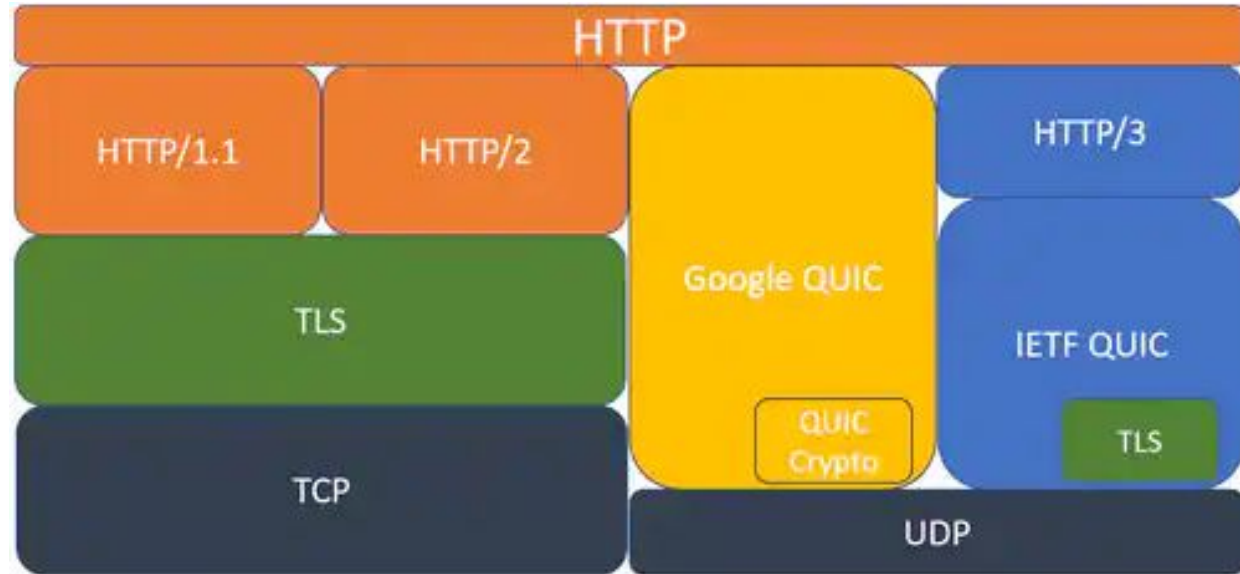
De data trecută: HTTP/2 HOL blocking solution



O₂, O₃, O₄ delivered quickly, O₁ slightly delayed

HTTP/3 și QUIC

- [RFC9000](#)
- TLS inclus
- siguranța trimiterii
- controlul fluxului
- controlul congestiei
- load balancing
- fluxuri independente
- conexiunea se bazează pe un ID, independentă de IP



<https://www.akamai.com/blog/performance/http3-and-quic-past-present-and-future>

<https://blog.cloudflare.com/the-road-to-quic/>
<https://www.youtube.com/watch?v=idViw4anA6E>

De data trecută



User



Insecure Connection



Normal HTTP



User



Encrypted Connection



SSL Certificate



Secure HTTPS

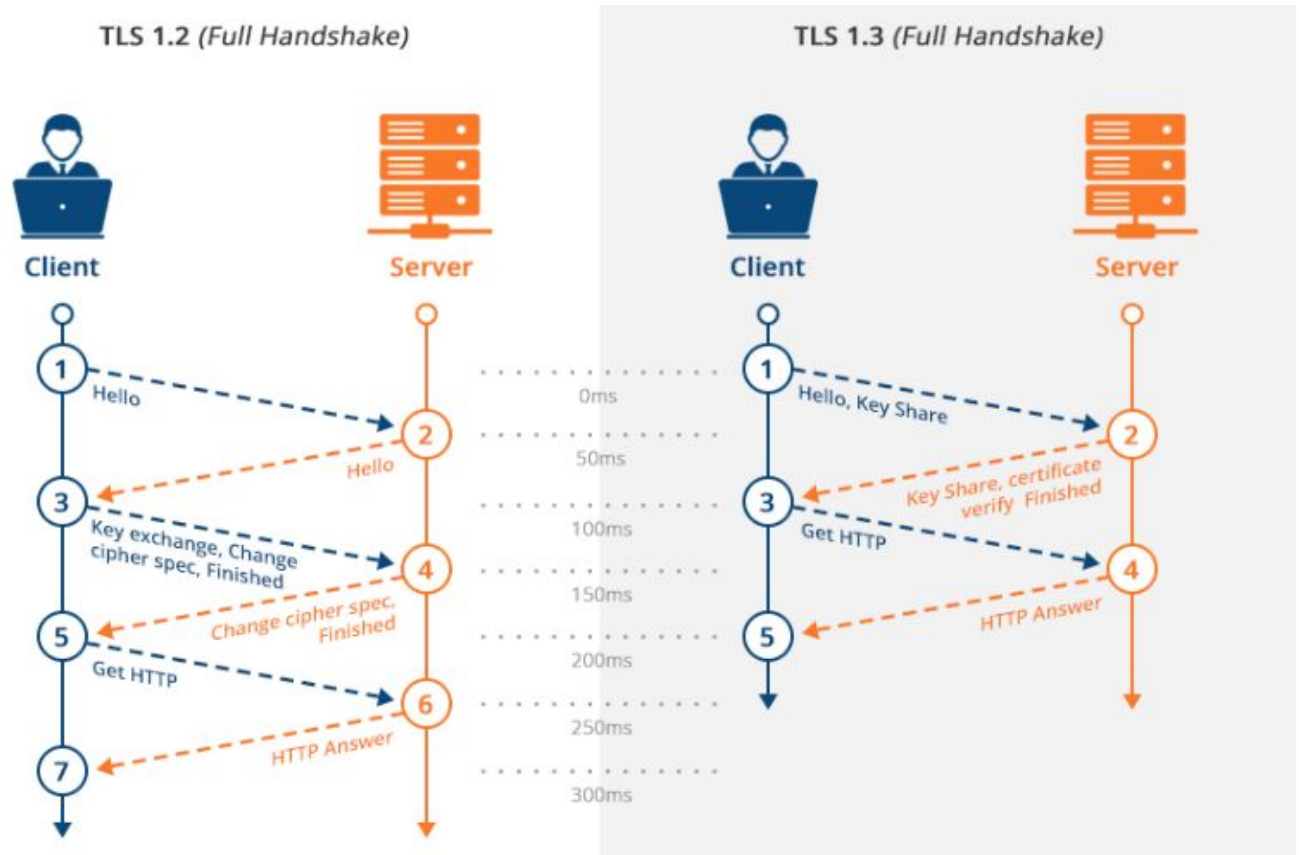
<https://howhttps.works/>

Despre metodele de criptare, puteți citi și [în bibliografie](#)

TLS 1.2 vs 1.3

mai detaliat:

<https://www.davidwong.fr/tls13/>



<https://www.wipro.com/blogs/suresha-ejari/five-ways-tls-1-3-will-take-your-privacy-and-performance-readiness-to-the-next-level/>

Exercițiu

Un alt serviciu de cloud este digitalocean, folosind acest link obțineți un credit de 100 de dolari pentru două luni:

<https://m.do.co/c/420cdd035b02>

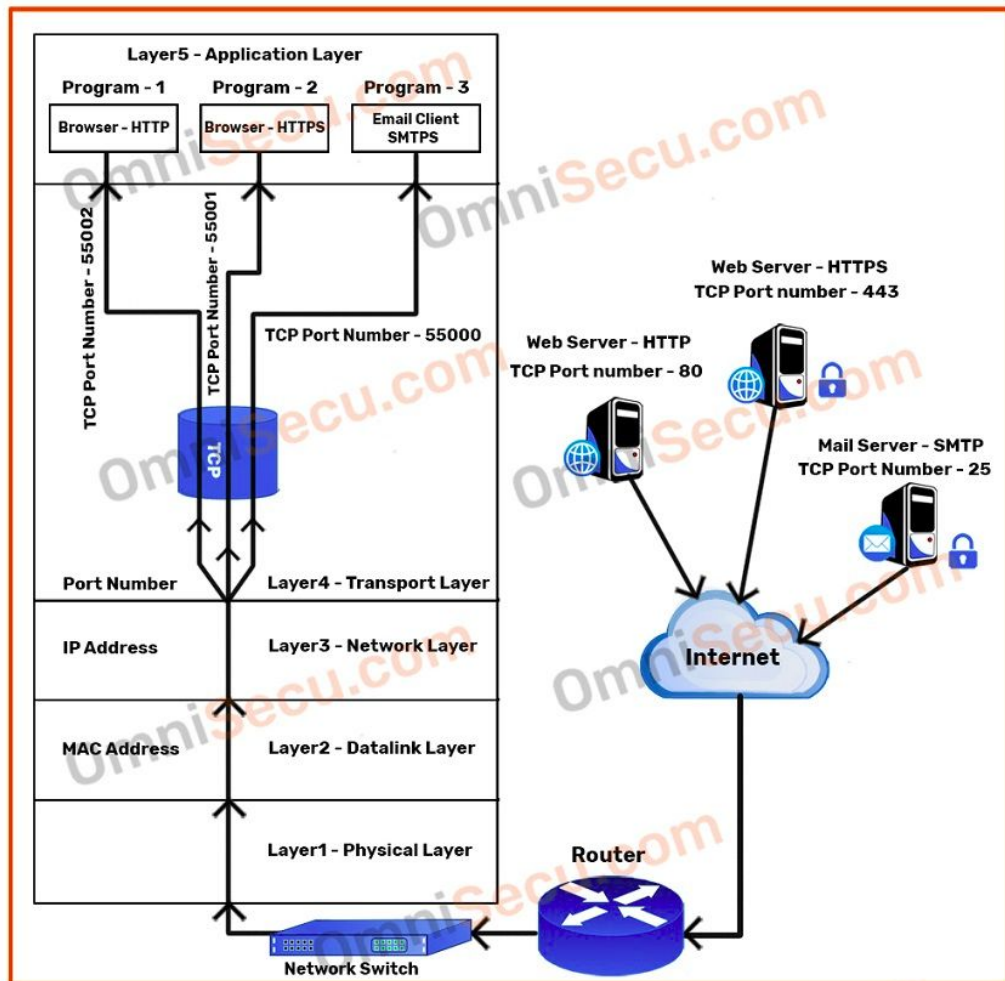
- dacă folosiți link-ul de mai sus, susțineți echipa de practică de la HLT <https://nlp.unibuc.ro> ;-)
- pe DigitalOcean aveți posibilitatea să creați un droplet care poate fi ținut live pe toată perioada în care aveți credit disponibil
- generați o pereche de chei publică-privată pentru a vă conecta la server

[un tutorial de generarea a cheilor este aici](#)

Porturi

- când ajunge mesajul la adresa IP destinație, care este procesul căruia trebuie să îi fie înmănat mesajul
- la nivelul aplicație și transport **porturile** reprezintă niște numere prin care se identifică aplicațiile între ele

De ce mai avem nevoie de porturi, dacă avem PID?



Secure Shell - SSH

<https://www.openssh.com/history.html>



- SSH este o aplicație client-server care permite instanțierea unui **shell** pe un calculator care se află într-o altă locație pe rețea
- alternativă la telnet și rlogin (aplicații nesecurizate)
- folosește protocolul **TCP** pentru transport. De ce?
- de obicei **portul 22** este rezervat pentru SSH
- permite și crearea unui **tunel** prin care să se transmită date în mod securizat
- cea mai sigură metodă este conectarea prin pereche cheie publică-cheie privată; merge și prin conexiune pe bază de parolă

[un tutorial pentru crearea unui tunel este aici](#)

Server deschis pe 0.0.0.0



- am executat [simple_flask.py](#) pe un server setând `host=0.0.0.0` și `port=8001`
- 0.0.0.0 nu este o adresă rutată, ci reprezintă *orice adresă*

Ce protocol la nivelul transport este folosit?

- am configurat în security groups ca portul **8001** să fie deschis pentru conexiuni din toată lumea
- serverul este accesibil de oriunde din lume, dar nu e securizat și îl folosesc mai mult pentru teste
-

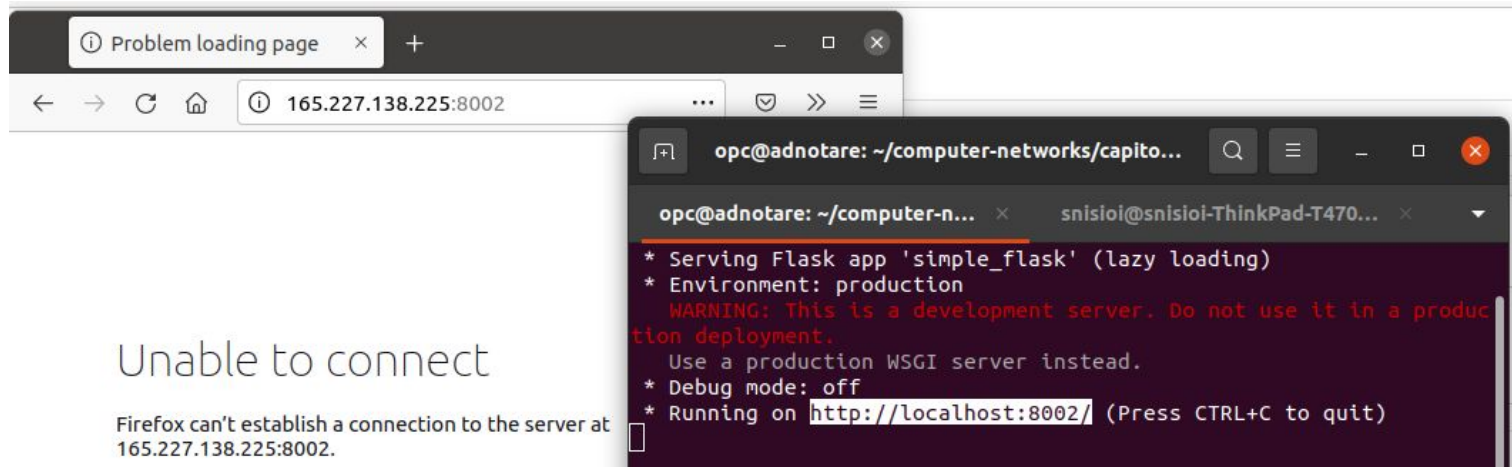
Cum facem să nu fie accesibil pentru oricine?
Cum îl accesăm doar noi?

A terminal window screenshot showing the execution of a Flask application. The prompt is 'opc@adnotare: ~/computer-networks/capitolul2/src\$'. The user runs 'vim simple_flask.py' and then 'python3 simple_flask.py'. The output shows Flask running in production mode on all addresses (0.0.0.0). It displays warnings about using a development server and provides the URL 'http://165.227.138.225:8001/'. Below the warnings, it shows two HTTP GET requests from 188.26.90.49, one for '/' and one for '/favicon.ico', both returning 200 status codes.

```
opc@adnotare: ~/computer-networks/capitolul2/src$ vim simple_flask.py
opc@adnotare:~/computer-networks/capitolul2/src$ python3 simple_flask.py
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://localhost:8002/ (Press CTRL+C to quit)
opc@adnotare:~/computer-networks/capitolul2/src$ python3 simple_flask.py
* Serving Flask app 'simple_flask' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses.
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://165.227.138.225:8001/ (Press CTRL+C to quit)
188.26.90.49 - - [07/Mar/2022 13:22:22] "GET / HTTP/1.1" 200 -
188.26.90.49 - - [07/Mar/2022 13:22:23] "GET /favicon.ico HTTP/1.1" 200 -
```

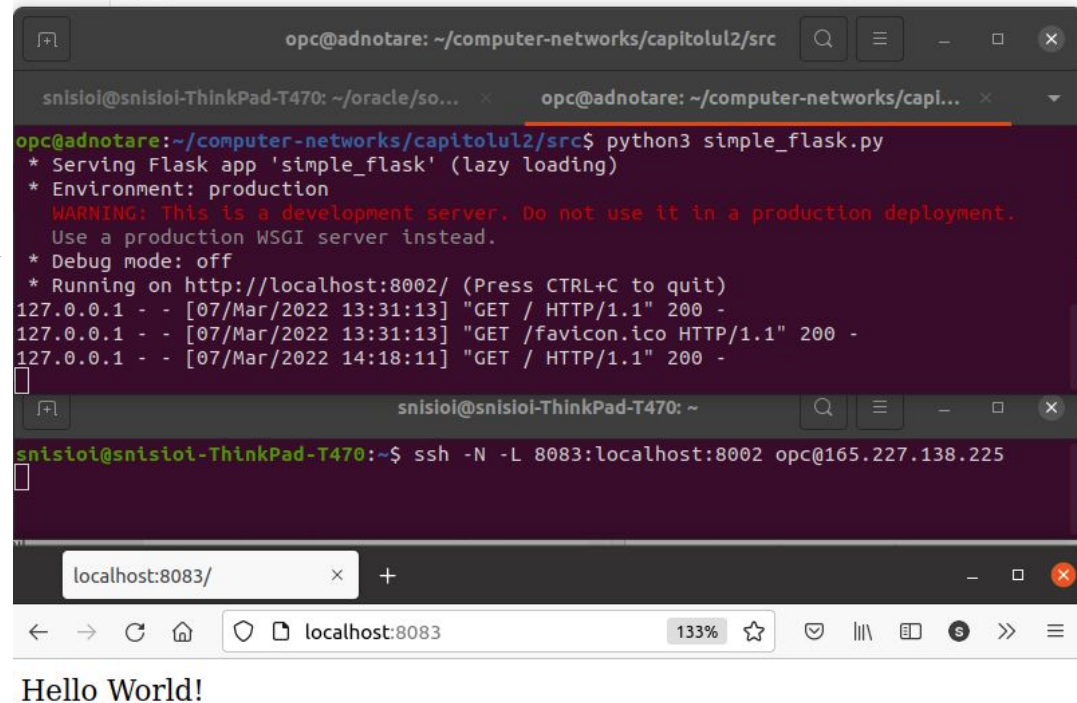
Server deschis pe localhost

- **localhost** este un **nume** rezervat pentru o clasă de adrese IPv4 **127.0.0.0/8** și singura adresă IPv6 0:0:0:0:0:0:0:1, scrisă cu notația **CIDR** **::1/128**
- orice server deschis pe localhost va fi deschis pe interfața de **loopback**
- nu va fi accesibil de nicăieri în afara host-ului local



Aplicație server deschisă pe localhost pe server, accesibilă local prin tunel SSH

- simple_flask.py deschis cu localhost:8002 pe server-ul remote (serverul se numește adnotare)
- prin `ssh -L`, redirectionăm mesajele care vin pe portul local **8083** către adresa **localhost:8002** de pe server
- putem accesa aplicația din browserul local folosind portul local pe care tocmai l-am alocat (8083)



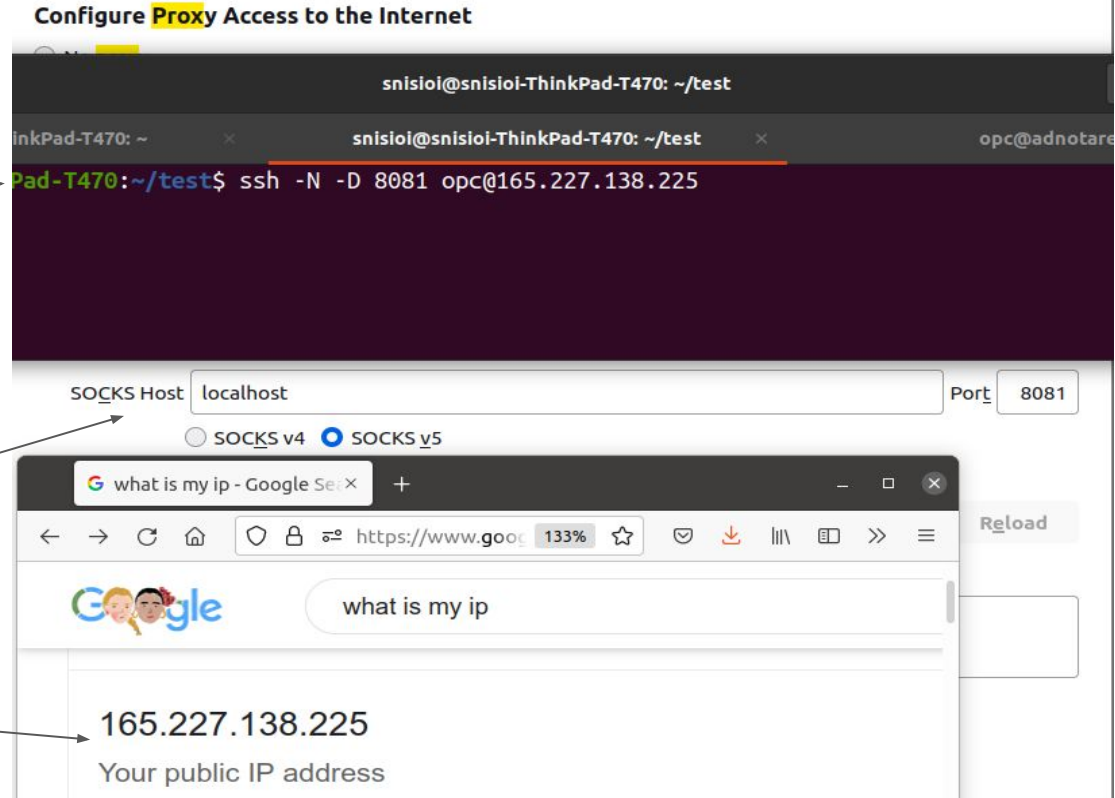
The image shows a terminal window and a web browser. The terminal window, titled 'opc@adnotare: ~/computer-networks/capitolul2/src', shows the execution of 'python3 simple_flask.py'. The output indicates that the Flask app 'simple_flask' is running on http://localhost:8002/. Below this, there are three GET requests from 127.0.0.1, all returning 200 status codes. The web browser, titled 'snisioi@snisioi-ThinkPad-T470: ~', has the address bar set to 'localhost:8083/'. The browser displays the text 'Hello World!'.

```
opc@adnotare: ~/computer-networks/capitolul2/src
snisioi@snisioi-ThinkPad-T470: ~/oracle/so... x  opc@adnotare: ~/computer-networks/capi... x
opc@adnotare:~/computer-networks/capitolul2/src$ python3 simple_flask.py
* Serving Flask app 'simple_flask' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://localhost:8002/ (Press CTRL+C to quit)
127.0.0.1 - - [07/Mar/2022 13:31:13] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [07/Mar/2022 13:31:13] "GET /favicon.ico HTTP/1.1" 200 -
127.0.0.1 - - [07/Mar/2022 14:18:11] "GET / HTTP/1.1" 200 -
snisioi@snisioi-ThinkPad-T470: ~
snisioi@snisioi-ThinkPad-T470:~$ ssh -N -L 8083:localhost:8002 opc@165.227.138.225
localhost:8083/
Hello World!
```

Dynamic Port Forwarding (nerecomandat)

Putem transforma server-ul într-un **proxy** securizat prin care să trimitem toate mesajele.

- dynamic port forwarding deschide un canal de comunicare de pe adresa **localhost:8081** către server, encapsulând orice mesaj de la nivelele inferioare
- la nivelul browserului putem seta **SOCKS** proxy ca fiind **localhost:8081**
- dacă verificăm în browser care este adresa IP, vom vedea că este chiar adresa serverului pe care am instanțiat conexiunea SSH cu dynamic port forwarding



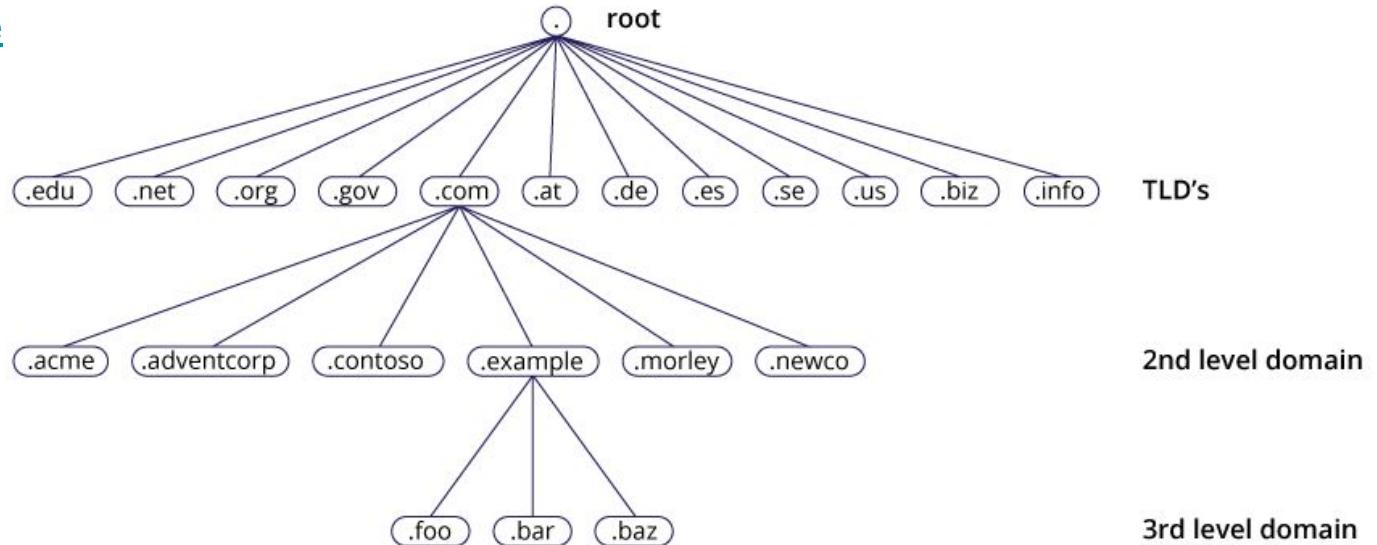
Domain Name System (DNS). Ce e?

- traduce un **nume** în adresa **IP** corespunzătoare
- crearea de alias pentru un **nume (hostname-ul** poate fi ceva complicat de reținut)
- numele este dat de un registrar: <http://www.internic.net>
- în mod tradițional, presupune că adresa IP este statică (nu se schimbă)
- alias pentru un numele unui server de e-mail
- servicii web replicate - mai multe IP-uri pot corespunde unui singur nume
- certificatele SSL sunt obținute pentru nume, nu pentru adrese IP
- poate fi folosit pentru blocarea accesului la site-uri în funcție de numele lor
- puteți citi mai multe și [în bibliografie](#)

Domain Name System

Tutoriale

- [Tutorial clar](#)
- [Tutorial ok](#)
- Cursuri [Jim Kurose](#)

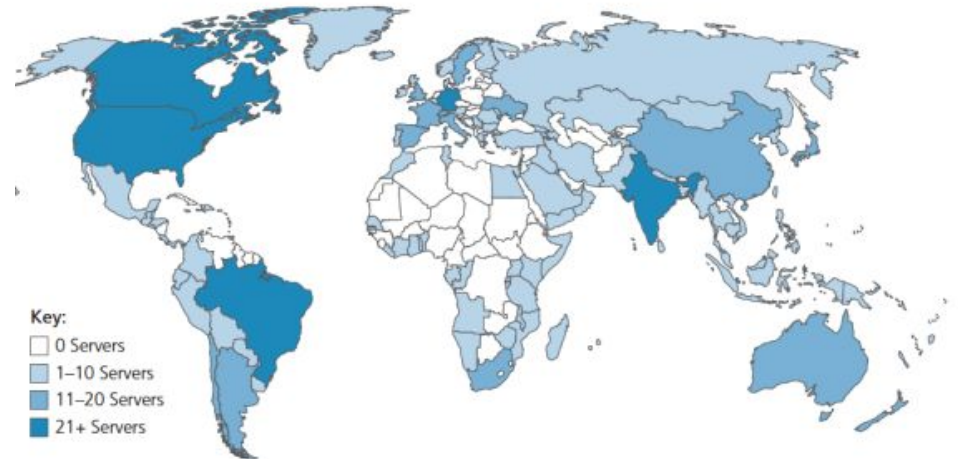


Root Servers

- absolut necesare, sunt folosite în ultimă instanță, atunci când nu există informația în cache
- oferă o funcție esențială internetului
- [ICANN](#) (Internet Corporation for Assigned Names and Numbers) manages root DNS domain
- [FAQs](#)
- sunt clonate în toată lumea și afișează aceeași adresă IP!

a.root-servers.net.
b.root-servers.net.
...
m.root-servers.net.

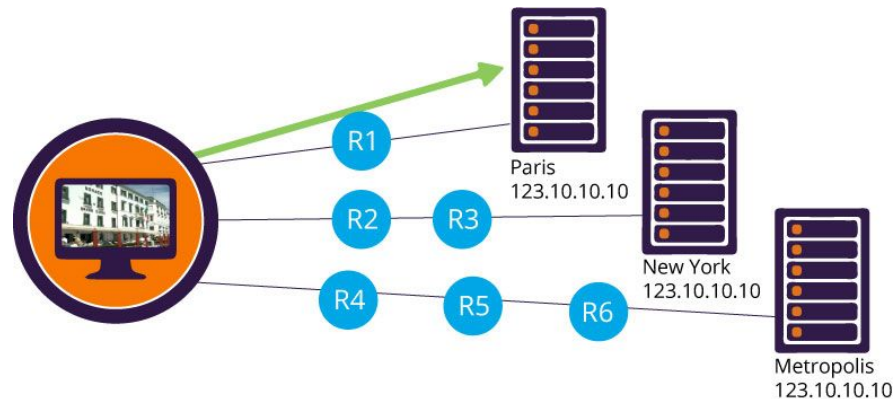
13 logical root name “servers”
worldwide each “server” replicated
many times (~200 servers in US)



<https://root-servers.org/>

Anycast

- severe care există în mai multe regiuni ale lumii, anunță că au aceeași adresă IP
- când un client face cererea de rezolvare a numelui, aceasta este redirectionată către adresa IP cea mai apropiată dpdv geografic



<https://www.imperva.com/blog/how-anycast-works/>

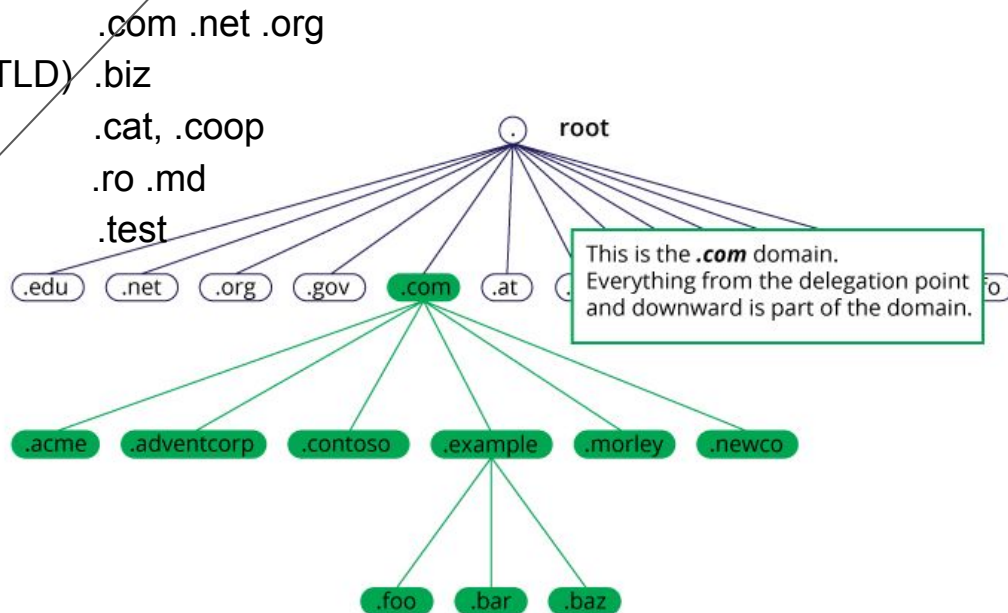
Top-level Domain (TLD)

Mai multe tipuri de TLD:

- Infrastructure top level domains (ARPA)
- Generic top level domains (gTLD):
- Restricted generic top level domains (grTLD)
- Sponsored top level domains (sTLD)
- Country code top level domains (ccTLD)
- Test top level domains (tTLD)
- TLD servers pentru **.ro**:

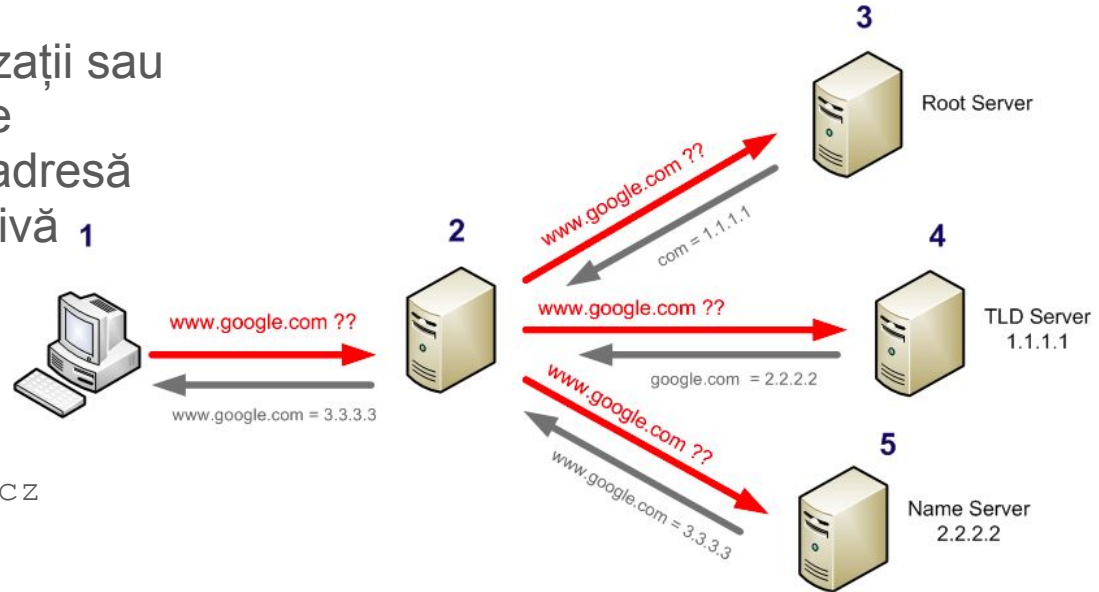
- sec-dns-b.rotld.ro.
- dns-at.rotld.ro
- sec-dns-a.rotld.ro.
- dns-ro.denic.de.
- dns-c.rotld.ro.
- primary.rotld.ro.

FQDN - fully qualified domain name; se termină cu punct la sfârșit și indică calea absolută



Server DNS Autoritativ

- servere DNS ale unei organizații sau **autoritatea** care se ocupă de maparea între un nume și o adresă IP pentru organizația respectivă 1
 - ns1.unibuc.ro.
pentru fmi.unibuc.ro
 - ns1.google.com.
pentru google.com
 - ns.ms.mff.cuni.cz.
pentru ufal.mff.cuni.cz
- sunt în general menținute de organizații sau chiar de ISP
- când cumpărăm un domeniu, *name server*-ul acelei firme de la care cumpărăm domeniul este autoritativ



<https://www.servermx.com/en/blog/Authoritative-vs-Non-authoritative-DNS-answers-00001/>



Server DNS local

1. cererile către DNS sunt întâi și întâi adresate DNS-ului local
2. acesta răspunde din cache-ul său sau din perechi nume - adresă obținute de-a lungul timpului
3. dacă nu are un răspuns, trimite cererea mai departe în ierarhia DNS
4. exemplu: router-ul de acasă poate avea un DNS server local
5. un **system resolver** server rulează și pe localhost, menținând într-un cache informațiile acumulate de-a lungul timpului

Execrțiu de configurarea a unui server DNS pe calculatorul local:

<https://www.fosslinux.com/7631/how-to-install-and-configure-dns-on-ubuntu.htm>

Intrări DNS (DNS records) mai bine ne uităm în [acest tutorial](#)

Type ?	Name ?	Data ?	TTL ?		
A	@	3.92.198.40	600 seconds	Delete	Edit
NS	@	ns31.domaincontrol.com.	1 Hour	Can't delete	Can't edit
NS	@	ns32.domaincontrol.com.	1 Hour	Can't delete	Can't edit
CNAME	www	chlorophylla.net.	1 Hour	Delete	Edit
CNAME	_domainconnect	_domainconnect.gd.domaincontrol.com.	1 Hour	Delete	Edit
SOA	@	Primary nameserver: ns31.domaincontrol.com.	1 Hour	Delete	Edit

Intrarea Start of Authority / Zone File

\$ORIGIN chlorophylla.net.

; SOA Record

@ 3600 IN SOA ns31.domaincontrol.com. dns.jomax.net. (
2022022803 ← serial nr. se incrementează la fiecare modificare a fișierului
28800 ← rata de refresh pentru serverul secundar (8h)
7200 ← timeout de retry pentru serv. secundar (2h)
604800 ← perioada de expirare a serv. secundar (7zile), apoi mai este autoritativ
3600 ← cache time pentru o eroare (1h), când nu găsește numele în acest fișier
)

; A Record

@ 600 IN A 18.206.88.223

; NS Record

@ 3600 IN NS ns31.domaincontrol.com. ← FQDN
@ 3600 IN NS ns32.domaincontrol.com.

server DNS
primar

adresa de
email

referinta la origin

DNS dinamic

-  Schimbare parolă PPPoE
-  Schimbare date WIFI și parolă
-  Vizualizare loguri conectare
-  Control parental
-  DNS dinamic
-  Administrare porturi

- un serviciu oferit de ISP sau de organizații care vă permite să faceți o mapare între adresele IP variabile pe care le primiți și un nume dat
1. obțineți un nume prin DNS dinamic (sau chiar un nume de la un DNS)
 2. configurați pe router-ul de acasă laptop-ul vostru să primească același IP de fiecare dată (IP static din subnet)
 3. configurați pe router-ul de acasă **port forwarding** astfel încât să se redirecționeze mesajele primite pe portul extern către portul deschis de laptop-ul vostru

Service Name:

[VIEW COMMON SERVICES](#)

Device IP Address:

[VIEW CONNECTED DEVICES](#)

External Port:

Internal Port:

Protocol:

Exemplu blocare prin DNS

- site-ul <https://rt.com> este momentan blocat, astfel că intrarea DNS corespunzătoare site-ului se redirecționează către altă pagină
- numele către **rt.com** nu mai este "rezolvat" sau se redirecționează către o pagină interpusă
- încercați `dig rt.com @dns1.rdsnet.ro` care (prind digi) vă va redirecționa către <http://81.196.9.130/>
- putem folosi <https://dns.yandex.com/> DNS: `dig rt.com @77.88.8.8`
- accesarea direct în browser a adresei `185.178.208.5` nu este permisă din motive de protecție împotriva [DDoS](#)
- trebuie să modificăm DNS-ul computerului și să facem flush la DNS cache (diferit în funcție de SO)



Acces neautorizat

Accesul dumneavoastră către acest site a fost restricționat în baza Deciziei Președintelui Autorității Naționale pentru Administrare și Reglementare în Comunicații nr.145/2021

Exercițiu

[descrie si aici](#)

1. Înscrieți-vă pe github pentru a obține student developer pack:
<https://education.github.com/pack>
2. Obțineți un domeniu gratuit timp de un an
3. Folosiți-l pentru a vă mapa IP-ul public de pe AWS sau de la orice alt provider de cloud
4. **Obțineți un certificat valid prin LetsEncrypt pentru domeniul vostru**
5. Configurați diferite intrări DNS și urmăriți timpul de propagare

Name.com

About Name.com

Best-in-class domains, email, and hosting

Benefit

One free domain name and free Advanced Security (SSL, privacy protection, and more).

Get access by connecting your GitHub account on [Name.com](#) >

Get help at [Name.com support](#)

Exemple de comenzi dig

- comenzi DNS folosind dig:

<https://github.com/senisioi/computer-networks/tree/2022/capitolul2#dns>

<https://www.cloudns.net/blog/linux-dig-command-install-use/>

Sfârșit