

# Capitolul 0: Cerințe preliminare – Virtualizare

*Sisteme de Operare – Laborator 0C preliminar*

v2.0 – Ubuntu 24.04 LTS

## 0.1 Introducere în virtualizare

Termenul virtualizare își are originea în latinescul *virtus* — noțiune asociată cu ceva potențial, cu acel „ceva care există doar ca posibilitate, fără a se materializa în mod direct”. Într-un context tehnic, virtualizarea desemnează totalitatea mecanismelor prin care sistemele și aplicațiile rulează pe platforme software intermediare, în loc să interacționeze nemijlocit cu hardware-ul fizic. Resursele fizice — procesoare, memorie, stocare — sunt abstractizate și prezentate sistemelor de operare oaspete sub forma unor echivalente virtuale, gestionarea acestei traduceri revenind unui strat software specializat numit hypervisor.

Să presupunem, de pildă, un calculator echipat cu Windows 10 pe care trebuie rulat și un sistem Linux, fără ca recurgerea la dual-boot să fie acceptabilă din motive de flexibilitate. Soluția constă în instalarea unei mașini virtuale: o aplicație (hypervisorul) va prelua o parte din resursele gazdei — un fișier-disc de câțiva gigaocteți în locul unei partiții fizice, o fracțiune din RAM, unul sau mai multe nuclee de procesor — și le va prezenta sistemului oaspete ca și cum ar fi hardware real. Efectul net: pe o singură mașină fizică coexistă mai multe sisteme de operare independente, fiecare cu mediul său de execuție izolat.

## 0.2 Mașini virtuale și containere

Virtualizarea se realizează prin două mecanisme fundamentale. Primul presupune utilizarea unui hypervisor de tip 1 (denumit și bare-metal), instalat direct pe hardware, fără un sistem de operare gazdă intermediar; el mediază accesul mașinilor virtuale la resurse. Al doilea tip — hypervisorul de tip 2 (hosted) — se instalează ca o aplicație într-un sistem de operare existent (gazda), adăugând un strat suplimentar de abstractizare.

Figura 1. Hypervisor de tip 1 (bare-metal)

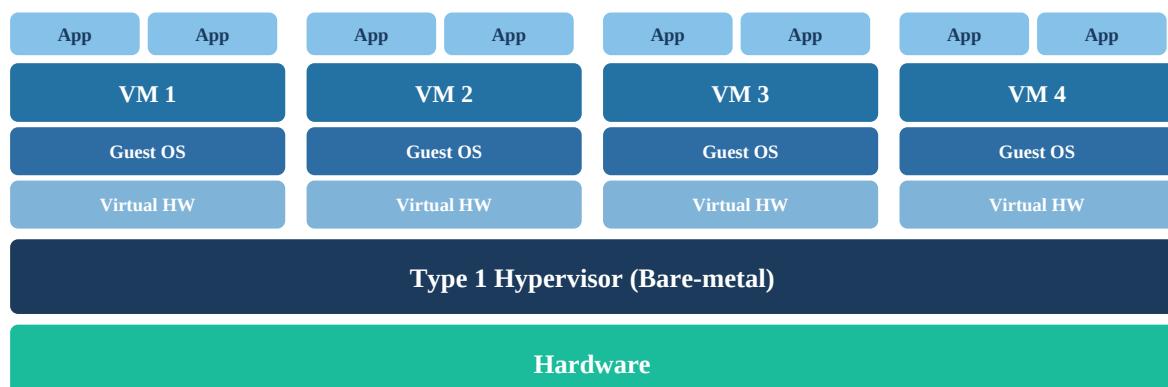
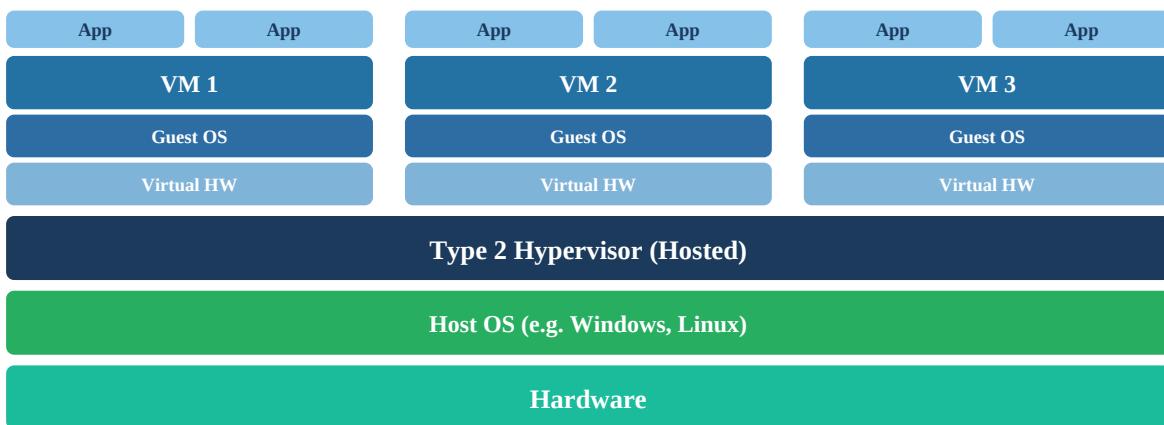


Figura 2. Hypervisor de tip 2 (hosted)



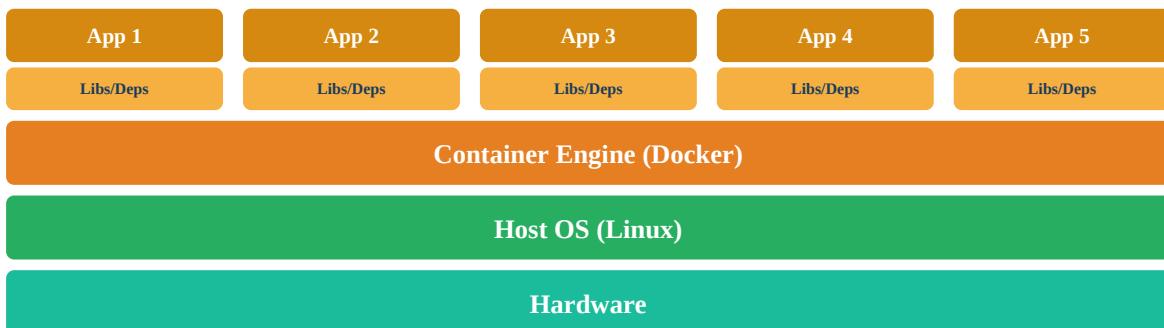
Sistemul de operare care rulează într-o mașină virtuală se numește sistem de operare oaspete (guest OS). Acesta se instalează, de regulă, dintr-o imagine preconfigurată, deși sunt posibile și instalări clasice de pe suporturi optice, prin PXE sau alte mecanisme de boot în rețea. Configurarea ulterioară — patch-uri de securitate, setări de rețea, permisiuni — este sarcina administratorului de sistem.

Un avantaj major al virtualizării constă în utilizarea eficientă a resurselor. Într-un scenariu tradițional, un server fizic de înaltă performanță găzduiește un singur sistem de operare pe care rulează multiple aplicații; în cazul unei avarii, toate devin indisponibile. Prin virtualizare, fiecare aplicație primește propriul mediu izolat, cu setări specifice, iar căderea unui VM afectează doar serviciul respectiv — celelalte continuă să funcționeze, realizându-se astfel high availability.

## Containerizarea

O evoluție semnificativă a paradigmelor de virtualizare o constituie containerizarea. Spre deosebire de mașinile virtuale, un container nu include un sistem de operare oaspete complet; în schimb, aplicația containerizată depinde de un motor de containere (e.g. Docker Engine) care comunică direct cu nucleul sistemului de operare gazdă, fără intermedierea unui hypervisor. Dezvoltatorul împachetează în container aplicația împreună cu toate dependențele necesare, obținând astfel portabilitate și reproductibilitate.

*Figura 3. Arhitectura containerizată (Docker)*



## Comparație între mașini virtuale și container

Criteriu	Mașini virtuale	Containere
Izolare	Completa (SO separat)	La nivel de proces (kernel partajat)
Consum resurse	Ridicat (SO complet per VM)	Reducit (fără SO suplimentar)
Timp pornire	Minute	Secunde
Portabilitate	Limitată de hypervisor	Ridicată (imagini standardizate)
Securitate	Izolare puternică	Necesită configurare atentă

Scalabilitate	Moderată	Ridicată (orchestrare K8s)
---------------	----------	----------------------------

## 0.3 Soluții de virtualizare

Alegerea unei soluții de virtualizare trebuie făcută cu atenție, întrucât compatibilitatea între produse diferite este adesea problematică. Mașinile virtuale se implementează (deploy) pornind de la o imagine care conține sistemul de operare oaspete, aplicațiile și configurațiile realizate de administrator.

VMware — Una dintre cele mai vechi și mai stabile platforme de virtualizare, deținută în prezent de Broadcom (achiziționată în 2023). Produsele vSphere și ESXi sunt apreciate pentru fiabilitate și suport enterprise.

KVM (Kernel-based Virtual Machine) — Hypervisor open-source integrat direct în nucleul Linux. Oferă scalabilitate superioară soluțiilor proprietare, fără costuri de licențiere. Nu trebuie confundat cu dispozitivul KVM switch (keyboard-video-mouse).

Xen — Hypervisor open-source de tip 1, proiect colaborativ al Linux Foundation. Designul microkernel îi conferă performanțe ridicate cu consum redus de resurse.

Hyper-V — Soluția de virtualizare Microsoft, integrată în Windows Server și Windows 10/11 Pro. Permite rularea distribuțiilor Linux în medii Windows.

VirtualBox — Produs open-source Oracle, ușor de instalat și configurat pe Linux, macOS sau Windows. Recomandat pentru utilizare desktop, mai puțin pentru medii de producție la scară largă.

Docker — Motor de containere care permite crearea de aplicații containerizate — componente autonome ce nu necesită un SO gazdă dedicat. Containerul facilitează împachetarea codului și a dependențelor. Proiect de vârf al Cloud Native Computing Foundation.

Kubernetes — Platformă de orchestrare a containerelor, dezvoltată inițial de Google și transformată în proiect open-source în 2014. Organizează resursele în clustere, noduri și pod-uri, oferind management automatizat, balansare de încărcare și scalare orizontală.

## 0.4 IaaS — Infrastructure as a Service

Concepțele prezentate mai sus stau la baza modelului IaaS (Infrastructure as a Service), o formă de cloud computing în care resurse de calcul — instanțe virtuale, stocare bloc, rețea — sunt livrate la cerere prin intermediul internetului. Sistemele de calcul se află în centre de date externe, sunt gestionate de hypervisor și interconectate prin rețele software-defined, comunicând prin API-urile unor platforme de orchestrare precum OpenStack, Apache CloudStack sau OpenNebula.

## 0.5 Configurarea VM-urilor și a containerelor

La crearea unei mașini virtuale noi, aceasta trebuie configurată pentru a comunica cu rețeaua externă: hostname unic, adresă IP, mască de subrețea, DNS și rută implicită. Imaginea de bază este stocată ca un set de fișiere gata de rulat, însă îi lipsesc informații esențiale — setări de rețea, permișuni, rute către baze de date, chei SSH — care sunt injectate prin scripturi de inițializare.

Utilitatea precum cloud-init simplifică acest proces: apeleză API-uri, citesc metadate de pe un server dedicat și configurează instanțele automat. În Kubernetes, containerele de inițializare (Init Containers) rulează înainte de pornirea aplicației principale.

Cheile SSH sunt folosite pentru administrarea securizată a containerelor și VM-urilor; se generează automat la instalare sau manual cu ssh-keygen. Un alt element critic este UUID-ul, stocat în fișierul D-Bus machine ID (/etc/machine-id) — un număr de 128 de biți cu probabilitate de coliziune neglijabilă.

## Extensiile de virtualizare

Funcționarea eficientă a hypervisorilor presupune suport hardware: extensiile Intel VT-x sau AMD-V, activate din BIOS/UEFI. Verificarea se realizează prin inspecția fișierului /proc/cpuinfo: prezența flag-ului vmx (Intel) sau svm (AMD) confirmă activarea acestora, iar lm indică o arhitectură pe 64 de biți.

### Verificarea extensiilor de virtualizare

```
$ grep -cE "vmx|svm" /proc/cpuinfo  
8  
$ lscpu | grep Architecture  
Architecture: x86_64
```

## Glosar de termeni

Container de aplicație — Aplicație împachetată care accesează direct resursele calculatorului prin intermediul unui motor de containere, fără un SO oaspete complet.

D-Bus machine ID — Fișier de configurare conținând UUID-ul unic al sistemului, generat la instalare sau la prima pornire.

Guest drivers / Guest agents — Componente software instalate în SO-ul oaspete care adaugă funcționalități precum suport de mouse, sincronizarea orei, partajare de foldere între gazdă și oaspete.

SSH host keys — Perechi de chei criptografice generate aleator, folosite pentru autentificarea securizată între sisteme.

Mașină virtuală (VM) — Sistem de calcul virtualizat care oferă acces la un SO complet, instalat pe hardware virtual furnizat de hypervisor.

## Partea practică: Instalare Ubuntu 24.04 LTS

### A. Pregătirea VirtualBox

Indiferent de distribuția aleasă, configurarea instanței în VirtualBox urmează aceiași pași. În paralel cu configurarea, se poate descărca imaginea Ubuntu Server 24.04 LTS de pe <https://ubuntu.com/download/server>.

Pas 1. Deschideți VirtualBox și apăsați New. În asistentul de creare:

- Name: srv1033 (ex.: srv + grupa de studiu)
- Type: Linux
- Version: Ubuntu (64-bit)
- Memory: minimum 1024 MB (recomandat 2048 MB)
- Hard disk: VDI, alocare dinamică, 15–20 GB

Pas 2. În Settings → Storage, atașați fișierul .iso descărcat la controlerul IDE (pictograma discului gol).

Pas 3. În Settings → Network, pentru Adapter 1 selectați Bridged Adapter în loc de NAT, astfel încât VM-ul să primească o adresă IP din rețeaua locală, direct de la serverul DHCP.

### B. Instalarea Ubuntu Server 24.04 LTS

Ubuntu, derivat din Debian, este una dintre cele mai răspândite distribuții Linux. Versiunea 24.04 LTS (Noble Numbat) beneficiază de suport extins de cinci ani (până în aprilie 2029). Instalatorul modern, Subiquity, înlocuiește vechiul debian-installer cu o interfață TUI intuitivă.

Pas 1. Porniți VM-ul. La ecranul de boot, selectați Try or Install Ubuntu Server.

Pas 2. Selectați limba (English recomandat pentru medii server) și layout-ul de tastatură.

Pas 3. La tipul instalării, alegeți Ubuntu Server (fără mediu desktop).

Pas 4. Configurarea rețelei: dacă adaptorul este setat la Bridged, interfața va primi automat o adresă IP prin DHCP. Verificați și confirmați.

Pas 5. Configurarea proxy: lăsați câmpul gol (cu excepția cazului în care vă aflați în spatele unui server proxy).

Pas 6. Mirror-ul Ubuntu: păstrați valoarea implicită (<http://ro.archive.ubuntu.com/ubuntu>).

Pas 7. Configurarea discului: selectați Use an entire disk. Acceptați schema de partiționare propusă (LVM implicit).

Pas 8. Confirmați scrierea pe disc (acțiune ireversibilă pentru conținutul VM-ului).

Pas 9. Creați contul de utilizator:

#### Exemplu de credențiale

Your name: Student SOP  
Server name: srv1033  
Username: 1033\_climantonio  
Password: student

Pas 10. La ecranul SSH, bifați Install OpenSSH server. Nu selectați pachete suplimentare din Featured Server Snaps.

Pas 11. Așteptați finalizarea instalării, apoi selectați Reboot Now. La prompt, scoateți imaginea ISO din Storage și apăsați Enter.

## C. Configurare post-instalare

### C.1. Primul login și comenzi de bază

După repornire, veți fi întâmpinați de ecranul de login (tty/getty). Introduceți credențialele create la instalare.

#### Comenzi de verificare inițială

```
$ whoami  
1033_climantonio  
$ hostname  
srv1033  
$ date  
Wed Feb 12 10:30:00 EET 2025  
$ ip addr show | grep inet  
    inet 127.0.0.1/8 scope host lo  
        inet 192.168.1.105/24 brd 192.168.1.255 scope global dynamic enp0s3
```

### C.2. Conectarea prin SSH

Pentru a facilita lucrul, conectați-vă la VM prin SSH de pe sistemul gazdă. Pe Linux/macOS, utilizați direct terminalul; pe Windows, folosiți ssh din PowerShell/CMD sau aplicația PuTTY.

#### Conectare SSH și escaladare de privilegii

```
# Conexiune SSH de pe sistemul gazdă  
$ ssh 1033_climantonio@192.168.1.105  
1033_climantonio@192.168.1.105's password: *****  
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-xx-generic x86_64)  
  
# Obținerea drepturilor de root  
$ sudo -i  
[sudo] password for 1033_climantonio: *****  
root@srv1033:~#
```

*Notă importantă: În Ubuntu, contul root este dezactivat implicit. Privilegiile administrative se obțin prin comanda sudo, care solicită parola utilizatorului curent (nu a root-ului).*

### C.3. Setarea hostname-ului

Hostname-ul se configura la instalare, dar poate fi modificat ulterior cu hostnamectl:

#### Configurare hostname

```
$ hostnamectl  
Static hostname: srv1033  
...  
Operating System: Ubuntu 24.04 LTS  
Kernel: Linux 6.8.0-xx-generic  
  
# Modificare hostname  
$ sudo hostnamectl set-hostname srv1033.clim.local  
$ cat /etc/hostname  
srv1033.clim.local
```

### C.4. Adăugarea unei adrese IP statice

Ubuntu 24.04 utilizează Netplan pentru configurarea rețelei. Fișierele de configurare se află în /etc/netplan/.

**Configurare IP static cu Netplan**

```
# Vizualizare configurare curentă
$ cat /etc/netplan/50-cloud-init.yaml

# Exemplu de configurare cu IP static
$ sudo nano /etc/netplan/01-static.yaml

network:
  version: 2
  ethernets:
    enp0s3:
      addresses:
        - 192.168.1.105/24
      routes:
        - to: default
          via: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]

$ sudo netplan apply
```

**C.5. Managementul pachetelor**

Ubuntu folosește managerul de pachete APT (Advanced Package Tool). Actualizarea regulată a sistemului este esențială pentru securitate.

**Gestiune pachete cu APT**

```
# Actualizare liste de pachete
$ sudo apt update

# Aplicare actualizări
$ sudo apt upgrade -y

# Instalare pachete utile
$ sudo apt install -y vim tmux mc htop net-tools
```