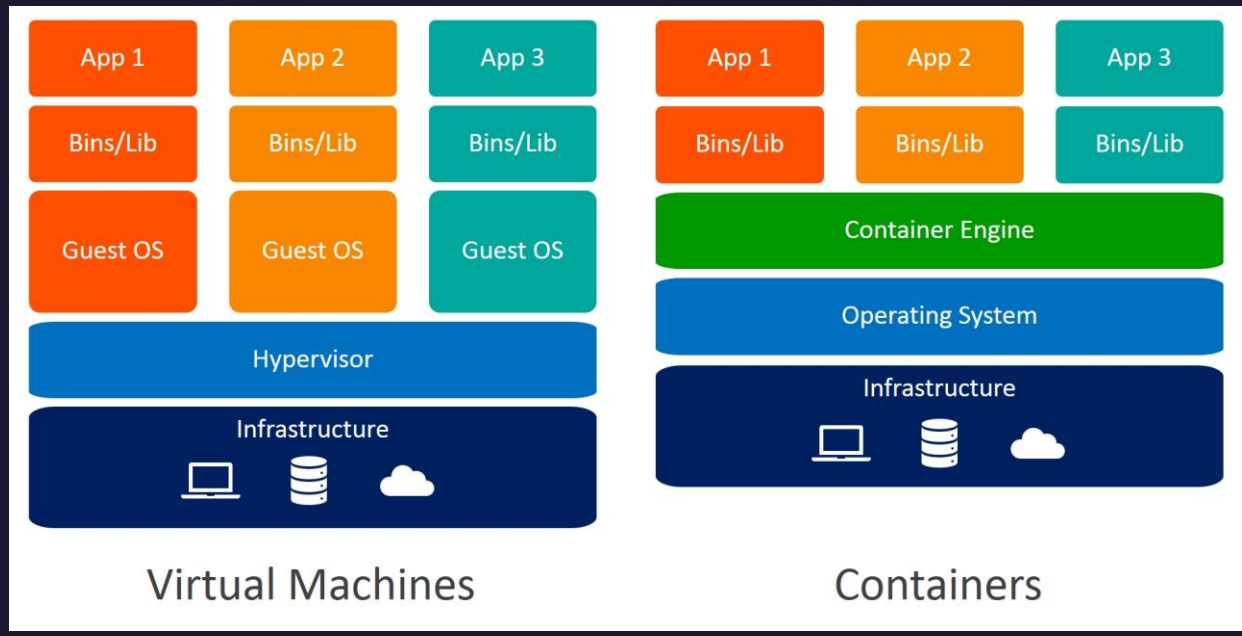
Laboratoare Administarea Retelelor de Calculatoare

Sisteme de containere Docker si LXD/LXC

Sistemele de Containere

- Exista doua moduri in care putem simula pe un server mai multe servere.
- Prima metoda si cea clasica este de a simulta complet un server incluzand componentele hardware, acest proces numindu-se virtualizare.
- Al doilea mod este de a simula doar de la kernelul sistemului de operare in sus aceste masini avand acces la acelasi kernel si avand denumirea de paravirtualizare sau containerizare si sta la baza unei infrastructuri de tip PaaS.



Sistemul LXD/LXC

- Vom aborda doua aplicatii care desi sunt din aceasi familie au abordari diferite.
- Incepem prin a face o masina virtuala Ubuntu Server.
- Si incepem instalarea lxd prin simpla comanda “apt install lxd” pe distributiile mai noi lxd vine preinstalat sau disonibil prin snap “snap install lxd”
- Apoi facem configuratia serviciului prin comanda “lxd init” si acceptand valorile default pentru inceput.

```
root@ubuntu22-server:/home/user# lxd init
Would you like to use LXD clustering? (yes/no) [default=no]:
Do you want to configure a new storage pool? (yes/no) [default=yes]:
Name of the new storage pool [default=default]:
Name of the storage backend to use (cephobject, dir, lvm, zfs, btrfs, ceph) [default=zfs]:
Create a new ZFS pool? (yes/no) [default=yes]:
Would you like to use an existing empty block device (e.g. a disk or partition)? (yes/no) [default=no]:
Size in GiB of the new loop device (1GiB minimum) [default=5GiB]:
Would you like to connect to a MAAS server? (yes/no) [default=no]:
Would you like to create a new local network bridge? (yes/no) [default=yes]:
What should the new bridge be called? [default=lxdbr0]:
What IPv4 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:
What IPv6 address should be used? (CIDR subnet notation, “auto” or “none”) [default=auto]:
Would you like the LXD server to be available over the network? (yes/no) [default=no]:
Would you like stale cached images to be updated automatically? (yes/no) [default=yes]:
Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:
```

Sistemul LXD/LXC

- Ruland comanda lxc fara argumente vedem in linii mari ce putem face.
- De mentionat ca lxd actioneaza mai mult ca o masina virtuala decat ca un PaaS oferind un acces mai putin restrictionat la sistem decat in cazul Docker.
- Acum suntem pregariti sa lansam un container.

```
  lxc [command]

Available Commands:
  alias       Manage command aliases
  cluster     Manage cluster members
  config      Manage instance and server configuration options
  console     Attach to instance consoles
  copy        Copy instances within or in between LXD servers
  delete      Delete instances and snapshots
  exec        Execute commands in instances
  export      Export instance backups
  file        Manage files in instances
  help        Help about any command
  image       Manage images
  import      Import instance backups
  info        Show instance or server information
  launch      Create and start instances from images
  list        List instances
  move        Move instances within or in between LXD servers
  network     Manage and attach instances to networks
  operation   List, show and delete background operations
  profile     Manage profiles
  project     Manage projects
  publish     Publish instances as images
  remote      Manage the list of remote servers
  rename      Rename instances and snapshots
  restart     Restart instances
  restore     Restore instances from snapshots
  snapshot    Create instance snapshots
  start       Start instances
  stop        Stop instances
  storage     Manage storage pools and volumes
  version     Show local and remote versions
  warning     Manage warnings
```

Sistemul LXD/LXC

- Acum putem lansa primul container ruland comanda “lxc launch <os:version> <container_name>”

```
root@ubuntu22-server:/home/user# lxc launch ubuntu:22.04 my-first-container
Creating my-first-container
Starting my-first-container
root@ubuntu22-server:/home/user#
```

- Ca sa vedem ce containere avem create putem rula comanda “lxc list”

```
root@ubuntu22-server:/home/user# lxc list
+--------------------+---------+----------------------+----------------------------------------------+-----------+-----------+
|        NAME        |  STATE  |         IPV4         |                     IPV6                     |   TYPE    | SNAPSHOTS |
+--------------------+---------+----------------------+----------------------------------------------+-----------+-----------+
| my-first-container | RUNNING | 10.79.154.186 (eth0) | fd42:5787:70cc:501e:216:3eff:fede:7713 (eth0) | CONTAINER | 0         |
+--------------------+---------+----------------------+----------------------------------------------+-----------+-----------+
root@ubuntu22-server:/home/user# _
```

- In cazul in care nu stim ce imagini avem disponib repository putem rula comanda “lxc image list images:”

```
abaddon@abaddon in ~ as 🧙 took 3s
λ lxc image list images: | more
+-----------------------------+--------------+--------+-----------------------------------------+--------------+-----------------+----------+------------------------------+
|            ALIAS            | FINGERPRINT  | PUBLIC |               DESCRIPTION               | ARCHITECTURE |      TYPE       |   SIZE   |         UPLOAD DATE          |
+-----------------------------+--------------+--------+-----------------------------------------+--------------+-----------------+----------+------------------------------+
| almalinux/8 (3 more)        | 6ef834f30365 | yes    | Almalinux 8 amd64 (20230331_02:28)      | x86_64       | VIRTUAL-MACHINE | 660.07MB | Mar 31, 2023 at 12:00am (UTC) |
+-----------------------------+--------------+--------+-----------------------------------------+--------------+-----------------+----------+------------------------------+
| almalinux/8 (3 more)        | 336e7e9978b8 | yes    | Almalinux 8 amd64 (20230331_02:28)      | x86_64       | CONTAINER       | 128.43MB | Mar 31, 2023 at 12:00am (UTC) |
+-----------------------------+--------------+--------+-----------------------------------------+--------------+-----------------+----------+------------------------------+
```

- Pentru a gasi o imagine specifica putem rula “lxc image list images:almalinux”

Sistemul LXD/LXC

- Si vedem o o lista de imagini cu almalinux pe care le putem folosi.
- Ca sa folosim una din ele de exemplu almalinux9 rulam comanda astfel “ lxc launch images:almalinux/9 <container_name>”
- Acum putem verifica fapul ca avem doua containere.

```
abaddon@abaddon in ~ as 🧙 took 2m16s
[[🧱][]] ⇒  λ lxc image list images:almalinux
```

ALIAS	FINGERPRINT	PUBLIC	DESCRIPTION	ARCHITECTURE	TYPE	SIZE	UPLOAD DATE
almalinux/8 (3 more)	6ef834f30365	yes	Almalinux 8 amd64 (20230331_02:28)	x86_64	VIRTUAL-MACHINE	660.07MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8 (3 more)	336e7e9978b8	yes	Almalinux 8 amd64 (20230331_02:28)	x86_64	CONTAINER	128.43MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/arm64 (1 more)	3598111df111	yes	Almalinux 8 arm64 (20230331_02:29)	aarch64	CONTAINER	124.97MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/cloud (1 more)	79c902459d6f	yes	Almalinux 8 amd64 (20230331_02:28)	x86_64	VIRTUAL-MACHINE	679.12MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/cloud (1 more)	0313713e1f07	yes	Almalinux 8 amd64 (20230331_02:28)	x86_64	CONTAINER	147.75MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/cloud/arm64	330bb39dcc1b	yes	Almalinux 8 arm64 (20230331_03:53)	aarch64	CONTAINER	143.83MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/cloud/ppc64el	b03e15ae6710	yes	Almalinux 8 ppc64el (20230331_02:28)	ppc64le	CONTAINER	152.27MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/8/ppc64el (1 more)	9fb4562d12db	yes	Almalinux 8 ppc64el (20230331_02:28)	ppc64le	CONTAINER	132.25MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9 (3 more)	54db8a1a21b3	yes	Almalinux 9 amd64 (20230331_02:28)	x86_64	CONTAINER	109.65MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9 (3 more)	357bf68374e9	yes	Almalinux 9 amd64 (20230331_02:28)	x86_64	VIRTUAL-MACHINE	562.90MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/arm64 (1 more)	c0ef66ae6e46	yes	Almalinux 9 arm64 (20230331_03:54)	aarch64	CONTAINER	105.71MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/cloud (1 more)	8bf404c790fb	yes	Almalinux 9 amd64 (20230331_02:28)	x86_64	CONTAINER	125.42MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/cloud (1 more)	ec78c68fc661	yes	Almalinux 9 amd64 (20230331_02:28)	x86_64	VIRTUAL-MACHINE	583.18MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/cloud/arm64	07d6931905d8	yes	Almalinux 9 arm64 (20230331_02:30)	aarch64	CONTAINER	121.06MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/cloud/ppc64el	70628f52f5bb	yes	Almalinux 9 ppc64el (20230331_02:28)	ppc64le	CONTAINER	127.43MB	Mar 31, 2023 at 12:00am (UTC)
almalinux/9/ppc64el (1 more)	050d10178f5b	yes	Almalinux 9 ppc64el (20230331_02:28)	ppc64le	CONTAINER	111.43MB	Mar 31, 2023 at 12:00am (UTC)

```
root@ubuntu22-server:/home/user# lxc launch images:almalinux/9 alma-test
Creating alma-test
Starting alma-test
root@ubuntu22-server:/home/user# _
```

```
root@ubuntu22-server:/home/user# lxc list
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
|        NAME        |  STATE  |         IPV4         |                     IPV6                      |   TYPE    | SNAPSHOTS |
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
| alma-test          | RUNNING | 10.79.154.212 (eth0) | fd42:5787:70cc:501e:216:3eff:fe1c:86df (eth0) | CONTAINER | 0         |
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
| my-first-container | RUNNING | 10.79.154.186 (eth0) | fd42:5787:70cc:501e:216:3eff:fede:7713 (eth0) | CONTAINER | 0         |
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
root@ubuntu22-server:/home/user# _
```

Sistemul LXD/LXC

- Pentru a intra intr-un container rulam comanda “lxc exec <nume_container> -- <shell>”.
- Vedem ca avem un alt prompt si un alt sistem de operare.
- Pentru a sterge un container “lxc delete <container_name> “ si flagul -–force daca daca avem containerul in executie
- Desigur aceasta este doar o introducere in acest sistem de containere si capacitatile sale sunt mult mai mari.

```
root@ubuntu22-server:/home/user# lxc exec alma-test -- bash
[root@alma-test ~]# cat /etc/os-release
NAME="AlmaLinux"
VERSION="8.7 (Stone Smilodon)"
ID="almalinux"
ID_LIKE="rhel centos fedora"
VERSION_ID="8.7"
PLATFORM_ID="platform:el8"
PRETTY_NAME="AlmaLinux 8.7 (Stone Smilodon)"
ANSI_COLOR="0;34"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:almalinux:almalinux:8::baseos"
HOME_URL="https://almalinux.org/"
DOCUMENTATION_URL="https://wiki.almalinux.org/"
BUG_REPORT_URL="https://bugs.almalinux.org/"

ALMALINUX_MANTISBT_PROJECT="AlmaLinux-8"
ALMALINUX_MANTISBT_PROJECT_VERSION="8.7"
REDHAT_SUPPORT_PRODUCT="AlmaLinux"
REDHAT_SUPPORT_PRODUCT_VERSION="8.7"
[root@alma-test ~]# _
```

```
root@ubuntu22-server:/home/user# lxc delete alma-test
Error: The instance is currently running, stop it first or pass --force
root@ubuntu22-server:/home/user# lxc delete alma-test --force
root@ubuntu22-server:/home/user# lxc list
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
|        NAME        |  STATE  |         IPV4         |                     IPV6                      |   TYPE    | SNAPSHOTS |
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
| my-first-container | RUNNING | 10.79.154.186 (eth0) | fd42:5787:70cc:501e:216:3eff:fede:7713 (eth0) | CONTAINER | 0         |
+--------------------+---------+----------------------+-----------------------------------------------+-----------+-----------+
root@ubuntu22-server:/home/user#
```

Sistemul Docker

- Pentru a instala docker in Ubuntu rulam comanda “apt install docker docker-compose”.
- Ruland docker fara agrumente vedem o lista de comenzi pe care le putem executa.
- Ca sa vedem sau sa cautam containere intram pe https://hub.docker.com/
-

```
  volume      Manage volumes

Commands:
  attach      Attach local standard input, output, and error streams to a running container
  build       Build an image from a Dockerfile
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
  images      List images
  import      Import the contents from a tarball to create a filesystem image
  info        Display system-wide information
  inspect     Return low-level information on Docker objects
  kill        Kill one or more running containers
  load        Load an image from a tar archive or STDIN
  login       Log in to a Docker registry
  logout      Log out from a Docker registry
  logs        Fetch the logs of a container
  pause       Pause all processes within one or more containers
  port        List port mappings or a specific mapping for the container
  ps          List containers
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rename      Rename a container
  restart     Restart one or more containers
  rm          Remove one or more containers
  rmi         Remove one or more images
  run         Run a command in a new container
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  search      Search the Docker Hub for images
  start       Start one or more stopped containers
  stats       Display a live stream of container(s) resource usage statistics
  stop        Stop one or more running containers
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
  top         Display the running processes of a container
  unpause     Unpause all processes within one or more containers
  update      Update configuration of one or more containers
  version     Show the Docker version information
  wait        Block until one or more containers stop, then print their exit codes
```

Sistemul Docker

- Acum putem rula un container de ex alpine ruland comanda “ docker --name <container name> <container_image>

```
root@ubuntu22-server:/home/user# docker run --name test alpine
root@ubuntu22-server:/home/user#
```

- Pentru a vedea ce containere sunt executate rulam comanda ps dar observam ca nu este nimic asa ca adaugam flag-ul -a si vedem ce am facut noi.

```
root@ubuntu22-server:/home/user# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@ubuntu22-server:/home/user# docker ps -a
CONTAINER ID   IMAGE     COMMAND     CREATED              STATUS                          PORTS     NAMES
e90a945cb0ef   alpine    "/bin/sh"   About a minute ago   Exited (0) About a minute ago             test
root@ubuntu22-server:/home/user# _
```

- Observam ca acest container a iesit cu un code 0 (adica succes) si nu a ramas in executie.

- Docker spre deosebire de lxd se ocupa de izolarea unei aplicatii si ruleaza cat timp aplicatia este in executie, pentru ca noi nu am executat nimic acesta a iesit.

Sistemul Docker

- Ca sa facem un container care sa ramana in exiecutie trebuie sa adaugam flagurile -ti
- Acum vedem ca suntem in container.
- Pentru a sterge putem folosi comanda “docker rm <container_name sau container id> si in cazul incare este in executie putem folosi flagul --force.

```
root@ubuntu22-server:/home/user# docker run -ti --name test2 alpine
/ # cat /etc/os-release
NAME="Alpine Linux"
ID=alpine
VERSION_ID=3.17.3
PRETTY_NAME="Alpine Linux v3.17"
HOME_URL="https://alpinelinux.org/"
BUG_REPORT_URL="https://gitlab.alpinelinux.org/alpine/aports/-/issues"
/ #
```

```
root@ubuntu22-server:/home/user# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@ubuntu22-server:/home/user# docker ps -a
CONTAINER ID   IMAGE     COMMAND     CREATED              STATUS                      PORTS     NAMES
5062da699a41   alpine    "/bin/sh"   About a minute ago   Exited (0) 9 seconds ago              test2
e90a945cb0ef   alpine    "/bin/sh"   6 minutes ago        Exited (0) 6 minutes ago              test
root@ubuntu22-server:/home/user# docker rm test
test
root@ubuntu22-server:/home/user# docker rm test2
test2
root@ubuntu22-server:/home/user#
```

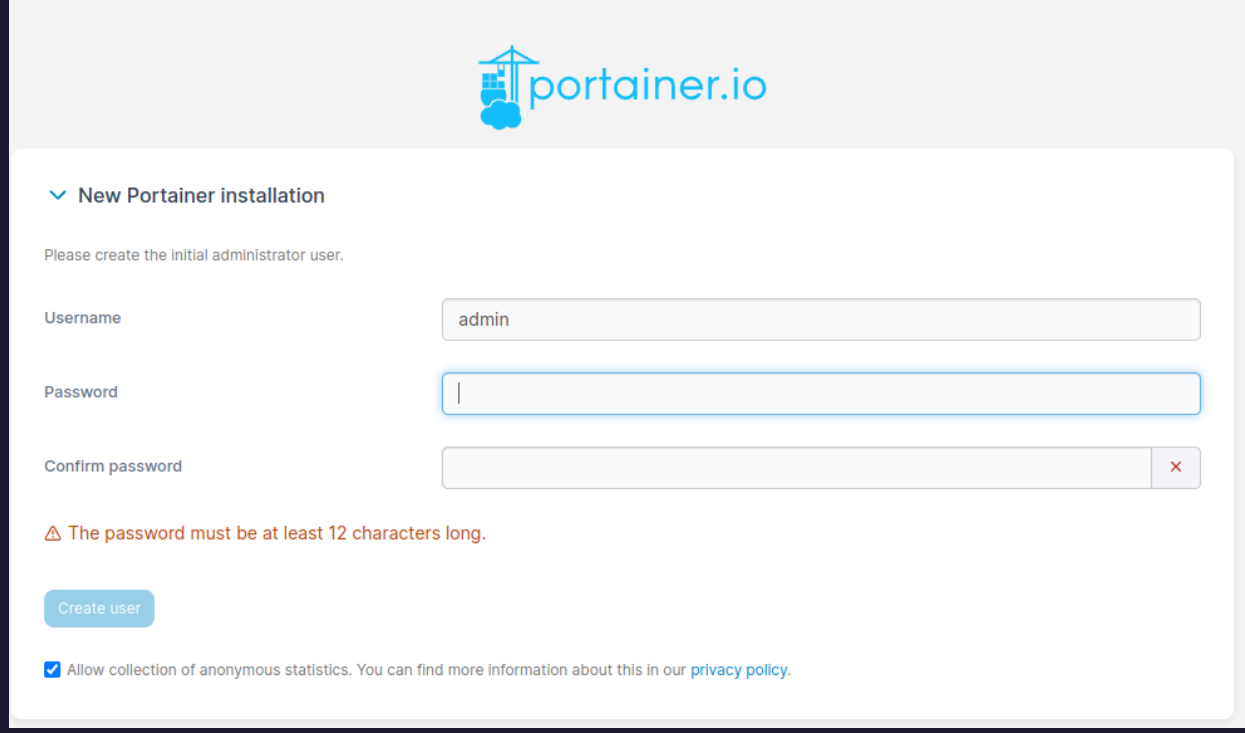
Sistemul Docker

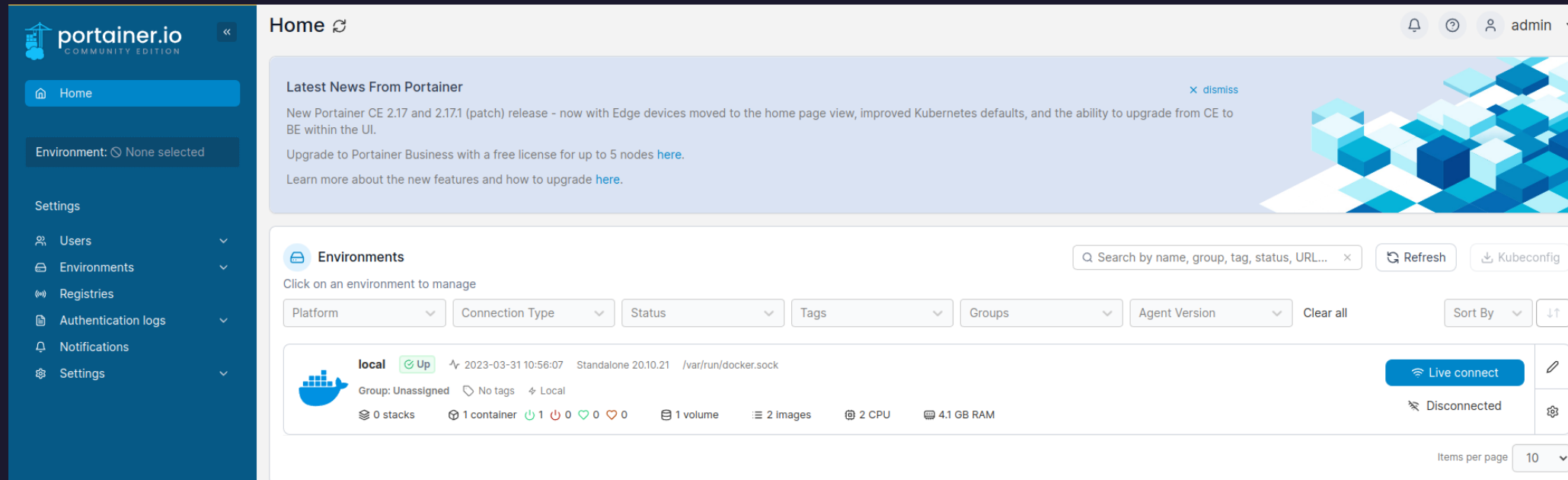
- Pentru docker exista un sistem de ochestrare numit pe care il vom folosi numit portainer.
- Nu avem nevoie decat sa rulam doua comenzi.
- “docker volume create portainer_data”
- “docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest”
- Flagul -d zice containerului sa intre in modul daemon (background si opusul lui -ti), flagul -p reprezinta maparea porturilor din container in gazda(ca sa poata fi folosite servicile furnizate de container) si flagul -v face maparea fisierelor/folderelor din gazda in container.

```
root@ubuntu22-server:/home/user# docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/por
ainer-ce:latest
Unable to find image 'portainer/portainer-ce:latest' locally
latest: Pulling from portainer/portainer-ce
772227786281: Pull complete
96fd13befc87: Pull complete
b733663f020c: Pull complete
9fbfa87be55d: Pull complete
Digest: sha256:9fa1ec78b4e29d83593cf9720674b72829c9cdc0db7083a962bc30e64e27f64e
Status: Downloaded newer image for portainer/portainer-ce:latest
d5a2f27ca4493428ce39403523a29408627c3feaba1214962018b614224ebc70
root@ubuntu22-server:/home/user# docker ps
CONTAINER ID   IMAGE                           COMMAND        CREATED         STATUS         PORTS
NAMES
d5a2f27ca449   portainer/portainer-ce:latest   "/portainer"   6 seconds ago   Up 4 seconds   0.0.0.0:8000→8000/tcp, :::8000→8000/tcp, 0.0.0.0:9443→9443/tcp, :::9443→9443/tcp, 9000/tcp
portainer
```

Sistemul Docker

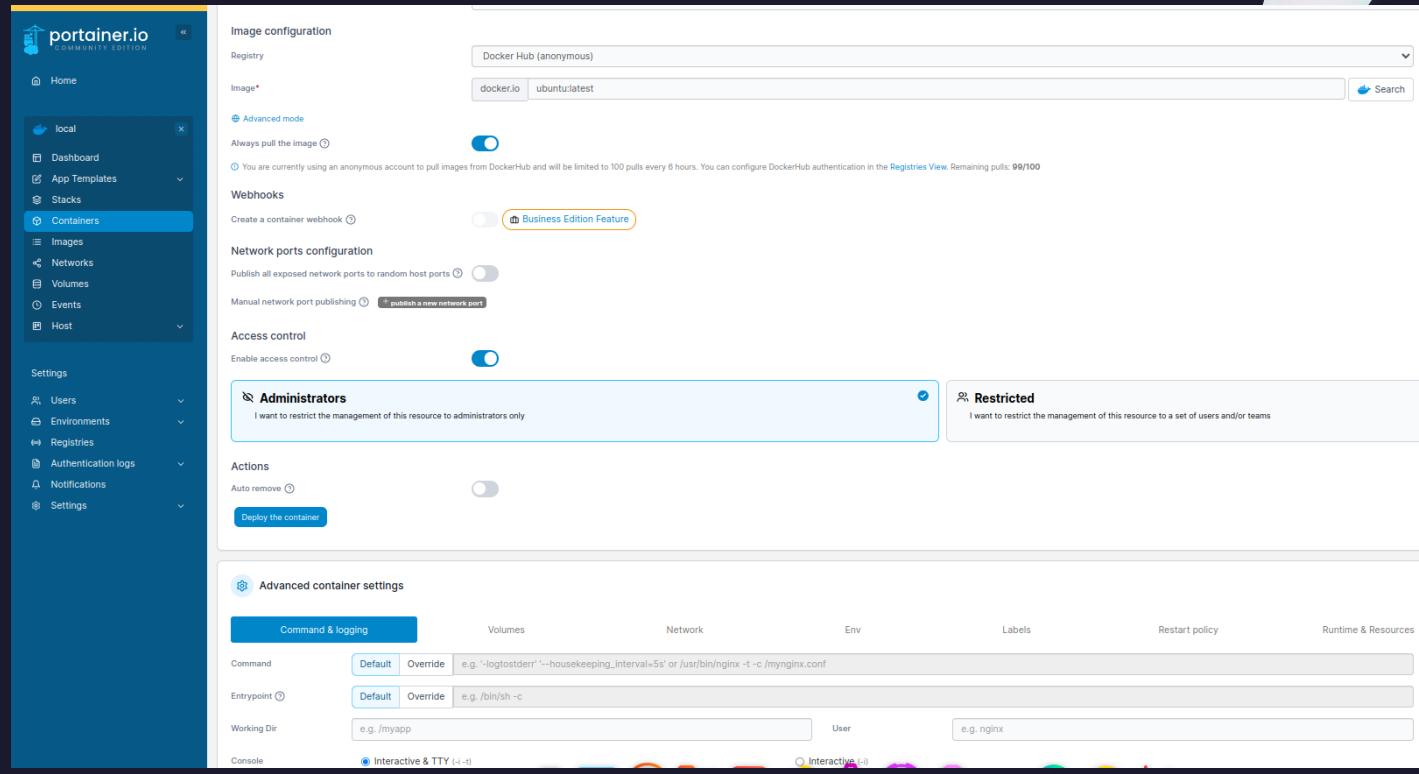
- Acum daca accesam link-ul “https://<masina_gazda_ip:9443/” vom ajunge in intrefata de administrare a dockerului.
- Si de aici procele sun mai simplificate.

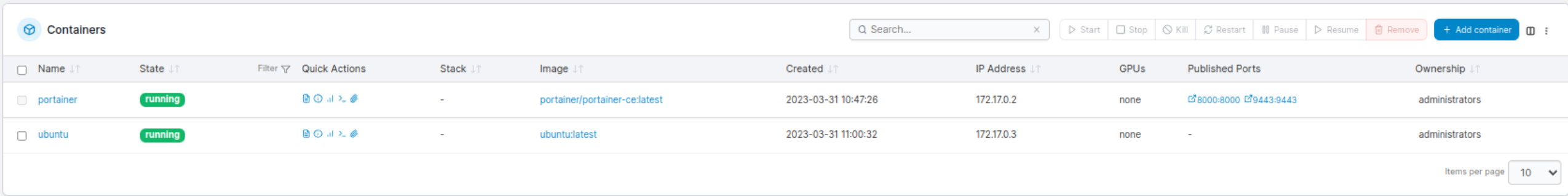




Sistemul Docker

- Pentru a face un container megem in Containers specificam numele, imagine si daca stim lisele de optiune din partea de jos.
- Dupa deploy vedem ca apare si ruleaza noua imagine





LXD/LXC

https://lxdware.com/exposing-the-host-lan-to-an-lxd-instance-with-macvlan/

https://linuxcontainers.org/lxd/docs/master/profiles/

https://askubuntu.com/questions/691039/adding-a-shared-host-directory-to-an-lxc-lxd-container

https://linuxcontainers.org/lxd/docs/master/howto/images_create/

https://ubuntu.com/blog/lxd-2-0-image-management-512

Docker

https://www.tecmint.com/name-docker-containers/

https://docs.docker.com/engine/reference/run/

https://docs.portainer.io/start/install-ce/server

https://medium.com/@BioCatchTechBlog/passing-arguments-to-a-docker-container-299d042e5ce

https://docs.docker.com/storage/volumes/