



Entorno Híbrido de Pruebas y Monitoreo-Proyecto2

Jaime Mendez

Universidad Santo Tomas de Aquino

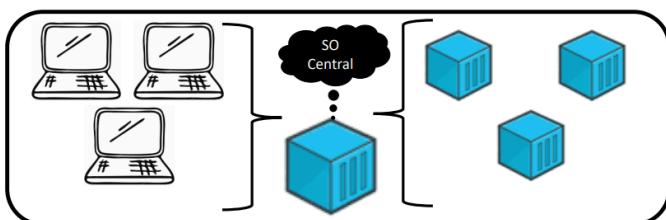
Bogotá, Colombia

[*jaimemendezj@usantotomas.edu.co*](mailto:jaimemendezj@usantotomas.edu.co)

I. INTRODUCCIÓN

El presente informe documenta el desarrollo y análisis de un entorno híbrido de pruebas, siguiendo las directrices del proyecto para el segundo corte de Sistemas Operativos. El objetivo principal fue construir una infraestructura compuesta por máquinas virtuales y contenedores, aplicando y reforzando conocimientos sobre la administración y análisis de diferentes sistemas operativos Linux, así como la implementación de herramientas de monitoreo centralizado.

Para lograr esto, se utilizó QEMU/KVM para la virtualización de tres sistemas operativos distintos: Rocky Linux, Manjaro Linux y Arch Linux. Simultáneamente, se empleó Docker para desplegar tres contenedores con imágenes de sistemas operativos diferentes: una base Arch Linux (representando a Garuda), Alpine Linux y Debian. Sobre cada una de estas seis instancias (3 VMs y 3 contenedores), se realizaron análisis específicos de servicios, procesos, red, y gestión de archivos/discos, utilizando herramientas nativas y apropiadas para cada sistema operativo, como se detalla en las Actividades 1 y 2.



Posteriormente, se construyó y configuró un contenedor Docker central basado en Fedora Linux, equipado con herramientas de diagnóstico de red y sistemas. Este contenedor sirvió como punto central para realizar análisis de

red y procesos sobre el resto del entorno híbrido, tal como se describe en la Actividad 4.

Finalmente, como objetivo de integración (Actividad 5), se procedió a desplegar una pila de monitoreo compuesta por Prometheus, Grafana y Zabbix, utilizando Docker Compose en el entorno central. El propósito fue configurar la recolección de métricas y estados de las máquinas virtuales y contenedores mediante agentes (Node Exporter, Zabbix Agent) y visualizar esta información de forma centralizada.

Este documento detallará el proceso de instalación, configuración, los análisis realizados, los desafíos encontrados durante la implementación y los resultados observados en cada una de las actividades propuestas.

II. DESARROLLO DE CONTENIDOS

A partir de esta sección, se desarrollan los contenidos del tema, de una forma ordenada y secuencial.

A. Configuración del Entorno de Virtualización y actividades (QEMU - Actividad 1)

Breve descripción de la instalación de Rocky, Manjaro, Arch.

Configuración de Red Host: Detalla la creación del bridge br0, su IP (192.168.100.1/24), muestra salida de ip addr show br0 del host.

```
[1/2] ju  
gixer155abs@gixer155:  
  
<network>  
  <name>red_gemu</name>  
  <bridge name="virbr1" stp="on" delay="0"/>  
  <mac address="52:54:00:dc:61:22"/>  
  <ip address="192.168.100.1" netmask="255.255.255.0">  
    <dhcp>  
      <range start="192.168.100.2" end="192.168.100.254"/>  
    </dhcp>  
  </ip>  
</network>
```

Configuración de Red VMs: Explica la asignación de MACs únicas (resolución del conflicto), configuración de IPs estáticas (192.168.100.10 a .12), gateway (192.168.100.1). Incluye capturas de ip a de las VMs mostrando las IPs finales.

```
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROJECT0 & docker network create --subnet=172.28.0.0/24 red-monitored  
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Head 'http://%2Fvar%2Frun%  
2fdocker.sock_ping': dial unix /var/run/docker.sock: connect: permission denied  
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROJECT0 & sudo docker network create --subnet=172.28.0.0/24 red-monitored  
d640649821c012b6992345154b4a0133d2455fa9e22b019
```

Instalación de las VMs: Este fue el proceso que se llevo acabo con la instalacion de las maquinas virtuales por medio de QEMU desde la terminal y luego obteniendo la interfaz grafica

-ARCHLINUX:El proceso que se llevo acabo con la instalacion de Arch-Linux, descargando su imagen ISO, y abriendo una terminal desde el directorio en cuestion

```
gixer155ab@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO2$ sudo qemu-system-x86_64 \
-m 4096 -smp 2 \
-cpu host \
-enable kvm \
-hda ./vm/arch/rancher.qcow2 \
-cdrom ./arch/rancher-2025.04.01-x86_64.iso \
-boot d \
-nicdev bridge,br=vhbr1,id=net0 \
-device e1000,netdev=net0
```

Con las siguientes specs:

-m 4096 -smp 2: Asigna 4GB de RAM y 2 núcleos de CPU.

-cpu host -enable-kvm: Usa la CPU del host y la aceleración KVM (buena práctica).

-hda ./VM_arch/archlinux.qcow2: Intenta usar este archivo como disco duro principal. Aquí está el problema.

-cdrom ./archlinux-2025.04.01-x86_64.iso: Monta la ISO de Arch en la unidad de CD virtual.

-boot d: Indica a la VM que intente arrancar desde la unidad de CD (d) primero.

-netdev bridge,br=virbr1,id=net0: Configura la red para usar un bridge llamado virbr1. (Nota: antes hablamos de br0, virbr0)

-device e1000,netdev=net0: Crea una tarjeta de red virtual tipo

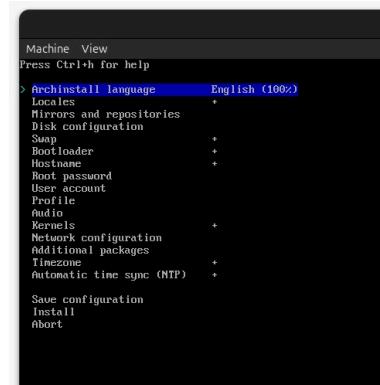
é 1000 conectada al netdev definido antes.



Esto nos confirma que el comando se ejecutó de forma correcta. Nos muestra el inicio del instalador.



Aquí usamos el método rápido y eficaz de instalación de Arch con el comando “Archinstall”, en vez de hacerlo de la forma tradicional y engorrosa. Ya solo es seguir los pasos.



Bash con las configuraciones de Archinstall, configuramos lo necesario, como repositorio (Colombia), zona horaria,

Archinstall language: Idioma del instalador (ya está en English).

Locales: Configuración regional (idioma del sistema instalado, formato de teclado, etc.)

Mirrors and repositories: Selección de los servidores desde donde se descargará los paquetes (importante para la velocidad de instalación)

Disk configuration: ¡Crucial! Aquí partitionarás tu disco duro virtual (el archivo .qcow2) seleccionarás el sistema de

archivos (ej. ext4, btrfs), y dónde montar cada partición (ej. /, /boot, swap). archinstall puede ofrecer opciones automáticas o manuales.

Swap: Configurar (o no) una partición/archivo de intercambio (swap).

Bootloader: Elegir e instalar el gestor de arranque (ej. GRUB, systemd-boot) necesario para iniciar el sistema operativo.

Hostname: Definir el nombre de red de la máquina (ej. Arch-VM).

Root password: Establecer la contraseña para el usuario administrador root.

User account: Crear una cuenta de usuario normal

(recomendado) y establecer su contraseña.

conjunto de paquetes base (ej. Desktop con un entorno como KDE o Gnome, Minimal, Server).

Audio. Configurar el servidor de audio (ej. PipeWire, PulseAudio).

Kernel: Seleccionar el kernel de Linux a instalar (ej. linux, linux-lts).

Network configuration: Configurar la red (usar DHCP o IP estática). Para este proyecto, aquí configurarías la IP estática (192.168.100.12/24, gateway 192.168.100.1, DNS 192.168.100.1, 8.8.8.8).

Additional packages: Especificar paquetes extra que quieras instalar además de los del perfil seleccionado.

Timezone: Definir la zona horaria (ej. America/Bogota).

Automatic time sync (NTP): Habilitar la sincronización automática de la hora.

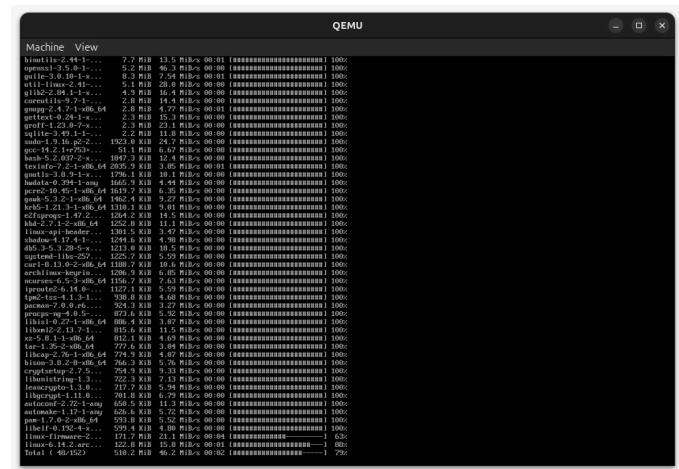
Save configuration: Guardar las opciones elegidas en un archivo JSON para reutilizarlas.

Install: ¡Iniciar el proceso de instalación una vez que todo lo anterior esté configurado!

Abort: Cancelar la instalación.

Confirmamos y listo.

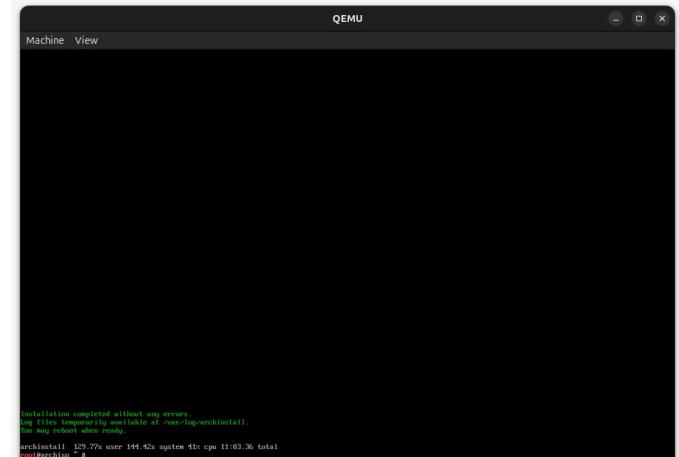
QEMU



Aqui vemos el log generado por esa instalacion

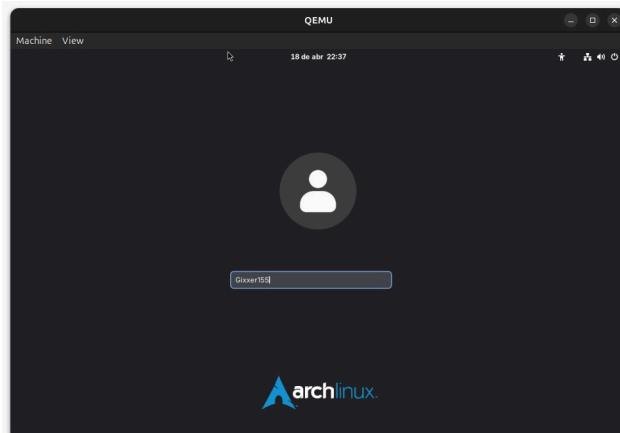


Nos pregunta si queremos hacer configuraciones adicionales, le presionaras que no para este caso.



Aqui empezara la instalacion con cuenta regresiva

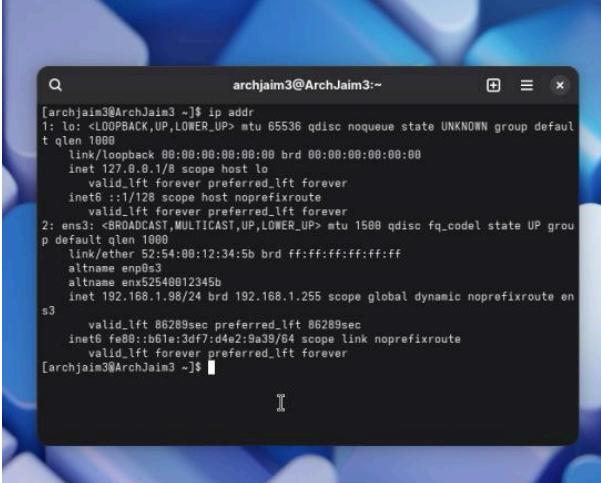
Y listo volvemos a reiniciar y cambiamos el boot que en vez de iniciar en cd-rom ahora que lo haga desde la partición local donde se encuentra el sistema operativo



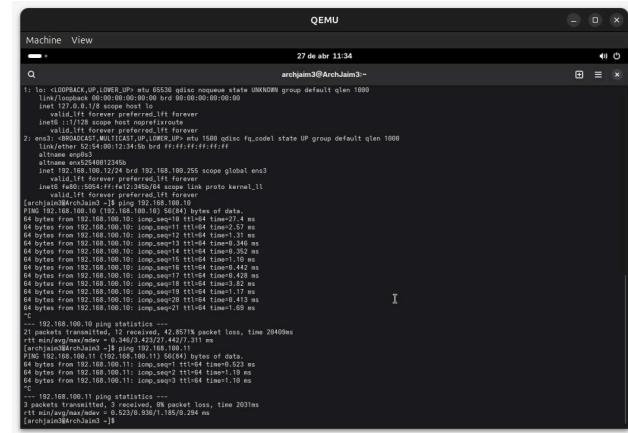
Aquí observamos que Archlinux inicio correctamente e iniciamos con las credenciales de usuario y contraseña tal como pusimos en el instalador.



Ahora iniciamos nuevamente ARCH pero cambiando el directorio en vez de cd-rom donde estaba la ISO, ahora inicia en la partición virtual que creamos



Configuramos y asignamos la IP 192.168.1.98

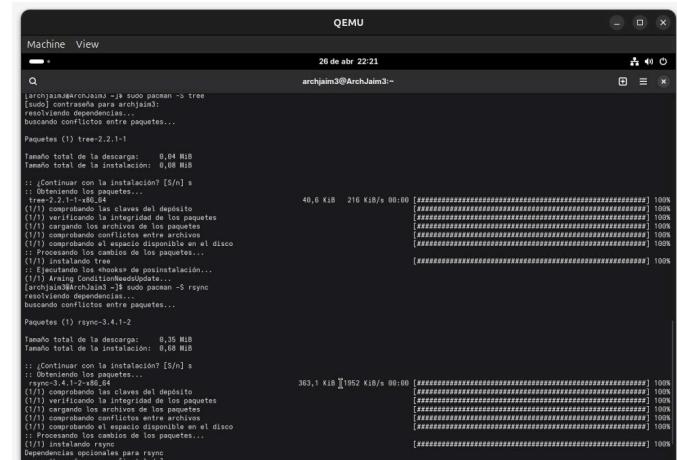


Aquí comprobamos conexión haciendo ping con las otras máquinas VM. Con pequeño cambio de IP's luego de evaluar el gateway

Desarrollar análisis de gestión de archivos y discos con comandos como "ncdu", "baobab", "tree" y "rsync".



Aquí instalamos el NCDU



Aquí instalamos TREE y rsync

Hacemos lo mismo con baobab

```
QEMU

Machine View
27 de abr 12:19
archjaim3@ArchJaim3:~>

root:~> ls -l / | head -n 10
total 1.1 GiB
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /usr
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /var
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /boot
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /home
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /etc
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /run
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /root
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /lib
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /tmp
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /bin
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /opt
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /mnt
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /proc
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /sys
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /dev
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /lib64
drwxr-xr-x 1 root root 4096 Mar 27 12:19 /lib

root:~> free -h
              total        used        free      shared  buff/cache available
Mem:       128.0 MiB     128.0 MiB          0 B     0 B          0 B          0 B
Swap:            0 B            0 B            0 B            0 B            0 B

root:~> df -h
Filesystem      Size  Used  Avail Mount point
/dev/sda1        128G  128G     0B  /
```

```
QEMU
Machine View
27 de abr 12:27
archjalm3@ArchJalm3:~$ tree .
.
+- dep_fv_kbL_v3492_bin.zst
+- dep_fv_kbL_v701_bin.zst
+- dep_fv_release_bin.zst -> avs/vskl/dsp_basefw/bin.zst
+- dep_fv_release_silicon.zst -> dep_fv_kbL_v3492_bin.zst
+- dep_fv_release_w985_bin.zst
+- fw.st 0f28.bin->48MHz_12a.master.zst
+- fw.st 0f28.silicon.zst
+- fw.st 0f28.xspd.bin.zst
+- fw.st 0f28_zip.zst
+- ikt-0040-0041.doc.zst
+- ikt-0040-0041.sfl.zst
+- ikt-0040-0041.sfl.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-1920.sfl.zst
+- ikt-0040-1950.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-1950.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0040-2130.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-2130.sfl.zst -> ikt-0040-0040.sfl.zst
+- ikt-0040-4150.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-4150.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0041-0041.doc.zst
+- ikt-0041-0041.sfl.zst
+- ikt-0093-0041.doc.zst
+- ikt-0093-0041.sfl.zst
+- ikt-0093-0291.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-0291.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0093-1050.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-1050.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0093-4150.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-4150.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0180-0041.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-0041.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0180-1950.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-1950.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0180-4150.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-4150.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0190-0041.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0190-0041.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0190-0041.pci.zst -> ikt-0093-0041.ddc.zst
+- ikt-0190-0041.pci.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0190-0041.web.zst -> ikt-0093-0041.web.zst
.
archjalm3@ArchJalm3:~$ ls -l
.
+- dep_fv_kbL_v3492_bin.zst
+- dep_fv_kbL_v701_bin.zst
+- dep_fv_release_bin.zst -> avs/vskl/dsp_basefw/bin.zst
+- dep_fv_release_silicon.zst -> dep_fv_kbL_v3492_bin.zst
+- dep_fv_release_w985_bin.zst
+- fw.st 0f28.bin->48MHz_12a.master.zst
+- fw.st 0f28.silicon.zst
+- fw.st 0f28.xspd.bin.zst
+- fw.st 0f28_zip.zst
+- ikt-0040-0041.doc.zst
+- ikt-0040-0041.sfl.zst
+- ikt-0040-1920.sfl.zst
+- ikt-0040-1950.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-1950.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0040-2130.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-2130.sfl.zst -> ikt-0040-0040.sfl.zst
+- ikt-0040-4150.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0040-4150.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0041-0041.doc.zst
+- ikt-0041-0041.sfl.zst
+- ikt-0093-0041.doc.zst
+- ikt-0093-0041.sfl.zst
+- ikt-0093-0291.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-0291.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0093-1050.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-1050.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0093-4150.doc.zst -> ikt-0093-0041.doc.zst
+- ikt-0093-4150.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0180-0041.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-0041.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0180-1950.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-1950.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0180-4150.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0180-4150.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0190-0041.doc.zst -> ikt-0040-0041.doc.zst
+- ikt-0190-0041.sfl.zst -> ikt-0040-0041.sfl.zst
+- ikt-0190-0041.pci.zst -> ikt-0093-0041.ddc.zst
+- ikt-0190-0041.pci.sfl.zst -> ikt-0093-0041.sfl.zst
+- ikt-0190-0041.web.zst -> ikt-0093-0041.web.zst
```

Aquí vemos la ejecución de NCDU, donde vemos como el directorio raíz de la partición virtual y los tamaños que ocupan dentro de ellos.

Aquí ejecutamos TREE, que nos permite ver todos los directorios existentes, todas las subcarpetas y archivos, TODO, es por eso que la salida de este comando es immenso, este comando se podría usar en directorio específico para ver su contenido.

```
QEMU  
Machine View  
27 de abr 12:20  
archimpi@ArchJaimi3:~  
Budu 2.7 - Use the arrow keys to navigate, press ? for help  
... /usr/lib  
285.0 MiB [=====] /firmware  
142.0 MiB [=====] /graphics  
147.9 MiB [=====] /geometry  
134.3 MiB [=====] /gpu  
30.8 MiB [=====] /host  
84.1 MiB [=====] /isoboot  
83.1 MiB [=====] /isowebkit  
53.4 MiB [=====] /lib  
48.0 MiB [=====] /glu  
45.8 MiB [=====] /dri  
40.8 MiB [=====] /pvr  
39.4 MiB [=====] /libglx  
39.3 MiB [=====] /libglx.so.19.0  
33.8 MiB [=====] /libjavascriptcoregtx-4.1.so  
32.5 MiB [=====] /libjavascriptcoregtx-6.0.so  
32.0 MiB [=====] /dgb  
30.4 MiB [=====] /libaudio.so  
29.1 MiB [=====] /libavcodec.so.54.0  
22.9 MiB [=====] /jude  
22.7 MiB [=====] /l3d  
21.0 MiB [=====] /gtTremor-1.0  
21.0 MiB [=====] /libtremor-1.0.33  
19.1 MiB [=====] /libuvashPython-3.3-3.1.so  
19.1 MiB [=====] /libvulkan_intel.so  
18.9 MiB [=====] /libuvashPython-3.3-3.1.so  
18.1 MiB [=====] /asm  
17.7 MiB [=====] /system  
17.0 MiB [=====] /libuvashppc64.so.61.19.190  
15.0 MiB [=====] /libvulkan_intel.so  
14.9 MiB [=====] /libuvashppc64.so  
13.9 MiB [=====] /libosmesa.so.6.0.0  
12.8 MiB [=====] /libvulkan_radeon.so  
12.7 MiB [=====] /libuvashppc64.so.5.0  
12.1 MiB [=====] /asm  
11.9 MiB [=====] /libuvashppc64.so.6.0.0  
10.2 MiB [=====] /girepository-1.0  
9.9 MiB [=====] /libgkt-4.0.so.1880.4  
9.8 MiB [=====] /libuvashppc64.so.6.0.0  
Total disk usage: 2.3 GiB Apparent size: 2.2 GiB Items: 20.111
```

```
QEMU
Machine View
27 de abr 12:29
archjain3@ArchJain3:~/home
Q archjain3@ArchJain3:~/home
└── ibt-0093-1059.doc.zst -> ibt-0093-0041.doc.zst
├── ibt-0093-1059.sfl.zst -> ibt-0093-0041.sfl.zst
├── ibt-0093-1059.sil.zst -> ibt-0093-0041.sil.zst
├── ibt-0093-1450.sfl.zst -> ibt-0093-0041.sfl.zst
├── ibt-0093-1450.sil.zst -> ibt-0093-0041.sil.zst
├── ibt-0100-0041.sfl.zst
├── ibt-0100-0041.sil.zst
└── ibt-0100-1050.doc.zst -> ibt-0040-0041.doc.zst
    ├── ibt-0100-1050.sfl.zst -> ibt-0040-0041.sfl.zst
    ├── ibt-0100-1050.sil.zst -> ibt-0040-0041.sil.zst
    ├── ibt-0100-1450.doc.zst -> ibt-0040-0041.doc.zst
    ├── ibt-0100-1450.sfl.zst -> ibt-0040-0041.sfl.zst
    ├── ibt-0100-1450.sil.zst -> ibt-0040-0041.sil.zst
    └── ibt-0100-0041.inf.zst
    └── ibt-0100-0041.pcl.doc.zst -> ibt-0093-0041.doc.zst
        ├── ibt-0100-0041.sil.zst
        └── ibt-0100-0041.usb.doc.zst
[archjain3@ArchJain3 ~]$ tree Desktop
Desktop [error opening dir]
B directories, 0 files
[archjain3@ArchJain3 ~]$ ls
Desktop [archjain3@ArchJain3 ~]$ ls -l
total 0
Desktop [archjain3@ArchJain3 ~]$ cd ..
[archjain3@ArchJain3 ~]$ tree home
home [error opening dir]
I
└── archjain3
    ├── Documentos
    ├── Escritorio
    ├── Imagenes
    ├── Música
    ├── Plantillas
    └── Pública

10 directories, 0 files
[archjain3@ArchJain3 ~]$
```

Aquí entramos a la primera carpeta y vemos su contenido con más detalles

Aqui lo usamos en un directorio especifico

The screenshot shows a Linux desktop interface with a terminal window, a file browser, and a system monitor.

- Terminal:** Shows the command `baobab` being run in a terminal window titled `[archjaim3@ArchJaim3 ~]$ baobab`.
- File Browser:** A file tree view titled `ArchJaim3` showing the directory structure of the system. The root directory (`/`) is expanded, showing its subdirectories and their sizes: `user` (4.4 GB), `var` (1.5 GB), `home` (115.5 GB), `etc` (8.2 MB), `lost+found` (0.0%), `srv` (12.3 kB), `root` (0.0%), `opt` (4.1 kB), and `mnt` (4.1 kB). The `home` directory is selected.
- System Monitor:** A circular disk usage chart showing the total disk space of 6.0 GB. The largest segments are `home` (blue, 115.5 GB), `root` (red, 4.1 kB), and `var` (green, 1.5 GB).

```
[archjain3@ArchJain3 ~]$ raync -av /  
sending incremental file list  
[archjain3@ArchJain3 ~]$ raync -av /proc/[root] /proc/[root]/SystemManager/system-connections  
[archjain3@ArchJain3 ~]$ raync -av /etc/audit/plugins.d  
[archjain3@ArchJain3 ~]$ raync -av /etc/audit/plugins.d failed: Permission denied (13)  
[archjain3@ArchJain3 ~]$ raync -av /etc/audit/plugins.d failed: Permission denied (13)  
drwxr-xr-x 4 898 2025 08/21 21:40:47 /etc/audit/  
drwxr-xr-x 2 898 2025 08/21 21:40:47 audit/  
drwxr-xr-x 7 2041 4121 03:56:21 lib -> /usr/lib  
drwxr-xr-x 7 2041 4121 03:56:21 lib64 -> /usr/lib64  
drwxr-xr-x 7 2041 4121 03:56:21sbin -> /usr/bin  
drwxr-xr-x 4 898 1986 1/21 19:00:00 boot  
drwxr-xr-x 110 113 546 2025 08/21 21:44:42 initramfs-linux-fallback.img  
drwxr-xr-x 8 894 546 2025 08/21 21:44:42 initramfs-linux.img  
drwxr-xr-x 4 898 2025 08/21 21:42:52 boot/grub  
drwxr-xr-x 5 498 2025 08/21 21:42:52 boot/grub/grub.cfg  
drwxr-xr-x 1 498 2025 08/21 21:42:52 boot/grub/fonts  
drwxr-xr-x 4 898 2025 08/21 21:42:58 boot/grub/fonts  
drwxr-xr-x 2 413 19 2025 08/21 21:42:58 boot/grub/fonts/unicode.pf2  
drwxr-xr-x 2 413 19 2025 08/21 21:42:58 boot/grub/fonts/unicode.pf2.old  
drwxr-xr-x 10 576 2025 08/21 21:42:48 boot/grub/1386-pc/quiet.mod  
drwxr-xr-x 1 576 2025 08/21 21:42:48 boot/grub/1386-pc/quiet.mod.old  
drwxr-xr-x 5 680 2025 08/21 21:42:44 boot/grub/1386-pc/afifs.mod  
drwxr-xr-x 6 956 2025 08/21 21:42:49 boot/grub/1386-pc/afifs.mod.old  
drwxr-xr-x 1 956 2025 08/21 21:42:49 boot/grub/1386-pc/afifs.mod.old.old  
drwxr-xr-x 15 688 2025 08/21 21:42:48 boot/grub/1386-pc/ahci.mod  
drwxr-xr-x 1 688 2025 08/21 21:42:48 boot/grub/1386-pc/ahci.mod.old  
drwxr-xr-x 956 2025 08/21 21:42:48 boot/grub/1386-pc/ahci.mod.old.old  
drwxr-xr-x 3 924 2025 08/21 21:42:48 boot/grub/1386-pc/architec.mod  
drwxr-xr-x 25 924 2025 08/21 21:42:48 boot/grub/1386-pc/architec.mod.old  
drwxr-xr-x 44 424 2025 08/21 21:42:46 boot/grub/1386-pc/anci_test.mod  
drwxr-xr-x 5 144 2025 08/21 21:42:48 boot/grub/1386-pc/anci_test.mod.old  
drwxr-xr-x 1 576 2025 08/21 21:42:48 boot/grub/1386-pc/keyboard.mod  
drwxr-xr-x 1 576 2025 08/21 21:42:48 boot/grub/1386-pc/keyboard.mod.old  
drwxr-xr-x 1 576 2025 08/21 21:42:48 boot/grub/1386-pc/acutrace.mod  
drwxr-xr-x 5 832 2025 08/21 21:42:48 boot/grub/1386-pc/acutrace.mod.old  
drwxr-xr-x 4 816 2025 08/21 21:42:46 boot/grub/1386-pc/biosdisk.mod  
drwxr-xr-x 2 188 2025 08/21 21:42:47 boot/grub/1386-pc/biosdisk.mod.old  
drwxr-xr-x 3 188 2025 08/21 21:42:47 boot/grub/1386-pc/biosdisk.mod.old.old  
drwxr-xr-x 2 084 2025 08/21 21:42:44 boot/grub/1386-pc/blocklist.mod  
drwxr-xr-x 1 084 2025 08/21 21:42:44 boot/grub/1386-pc/blocklist.mod.old  
drwxr-xr-x 2 628 2025 08/21 21:42:44 boot/grub/1386-pc/host.mod  
drwxr-xr-x 1 268 2025 08/21 21:42:48 boot/grub/1386-pc/hostlist.mod  
drwxr-xr-x 3 136 2025 08/21 21:42:48 boot/grub/1386-pc/hostlist.mod.old
```

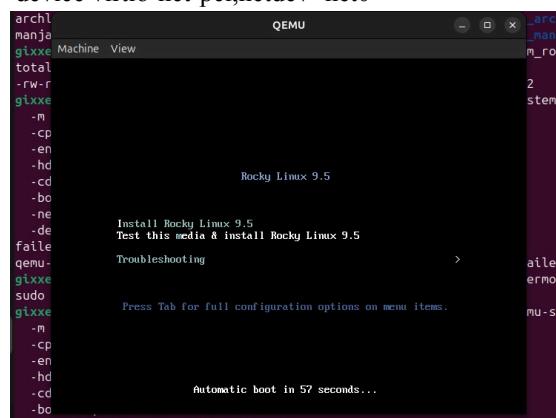
Aquí ejecutamos baobab que es un analizador de disco performativo gráfico a diferencia de NCDU

-ROCKY LINUX: En este paso consiste en la instalación de Rocky Linux

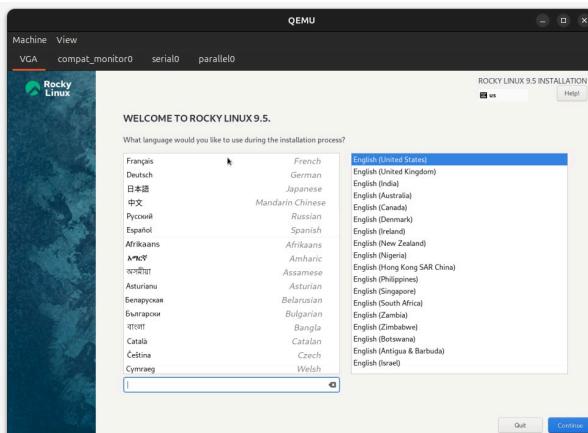
```
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTO2$ sudo qemu-system-x86_64 \
-m 4096 -smp 2 \
-cpu host \
-enable-kvm \
-hda ./VM_rocky/rocky.qcow2 \
-cdrom ./Rocky-9.5-x86_64-dvd.iso \
-boot d \
-netdev bridge,br=virbr1,id=net0 \
-device e1000,netdev=net0
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTO2$
```

Iniciamos QEMU desde la terminal para la instalación con las siguientes specs

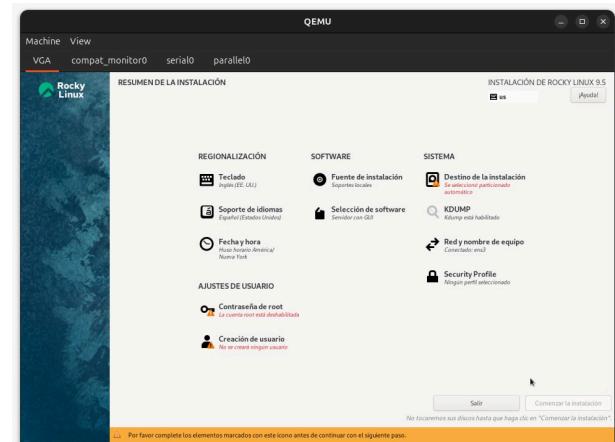
```
sudo qemu-system-x86_64 \
-m 4096 -smp 2 \
-cpu host \
-enable-kvm \
-hda ./VM_rocky/rocky.qcow2 '# <-- ¡VERIFICA ESTA RUTA/NOMBRE! \
-boot c \
-netdev bridge,br=br0,id=net0 '# <-- ¿USAR br0 o virbr1? \
-device virtio-net-pci,netdev=net0
```



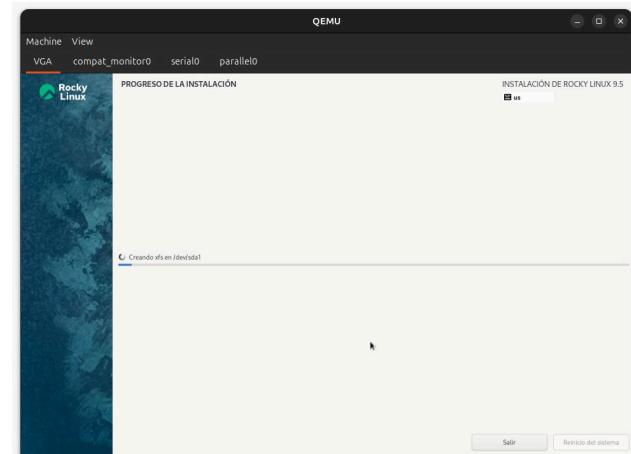
Aquí vemos que la instalación inicio correctamente, seleccionamos Install Rocky.



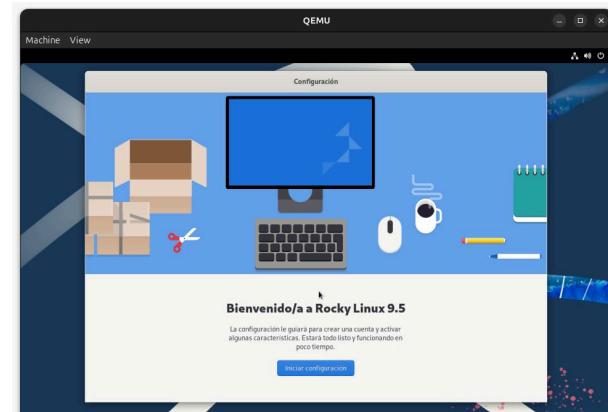
Seguimos con la instalación la cual es intuitiva y fácil. Aquí seleccionamos el idioma.



Esta es la pantalla "Resumen de la Instalación" ("Installation Summary"), que es el centro neurálgico desde donde configuras todas las opciones antes de que comience la instalación de Rocky Linux en el disco duro virtual. Dejamos todo por defecto ya que no se considera hacer gran configuración.



De ser así empezara a cargar.



Ya después de un par de minutos la instalación finalizara con éxito.

```
gixerx155abs@gixerx155abs-Latitude-E6220:~/Corte_2/PROYECTO2$ sudo qemu-system-x86_64 \
-m 4096 \
-smp 2 \
-cpu host \
-enable-kvm \
-hda /home/gixerx155abs/Corte_2/PROYECTO2/vm_rocky/rocky.qcow2 \
-netdev bridge,br=br0,id=net0 \
-device virtio-net-pci,netdev=net0,mac=52:54:00:12:34:5C
[sudo] contraseña para gixerx155abs:
```

Ya después iniciamos nuevamente Rocky cambiando el directorio en vez de cd-rom donde estaba la ISO, ahora iniciamos en la partición virtual que creamos.

Aqui configuramos con la IP 192.168.1.97 estatica IPV4

```
[jaimeandresmendez@localhost ~]$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
        linklayer  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
        linklayer  
2: ens3: <NOFORWDIGIT,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 52:54:00:12:34:5c brd ff:ff:ff:ff:ff:ff  
    altname enp0s3  
    inet 192.168.100.11/24 brd 192.168.100.255 scope global noprefixroute ens3  
        valid_lft forever preferred_lft forever  
        linklayer  
    inet6 fe80::3a30:3118%ens3/128 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
        linklayer  
[jaimeandresmendez@localhost ~]$
```

Pero luego usamos IP 192.168.100.11

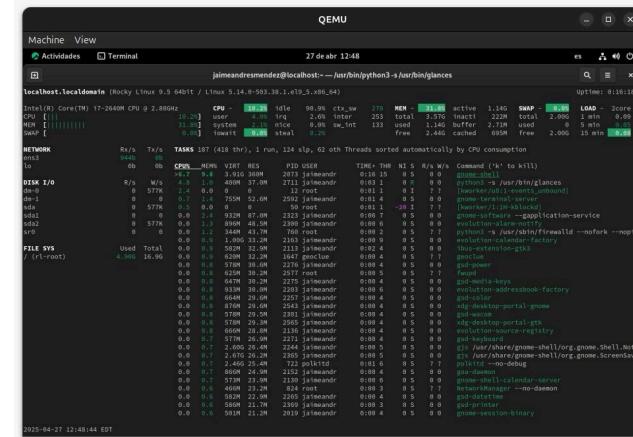
```
QEMU
Machine View
Actividades Terminal 27 de abr 12:34
jaimeandresmedez@localhost:~ [1]
[jaimeandresmedez@localhost ~]$ ping -c 10 192.168.100.1
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=5 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=6 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=7 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=8 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=9 ttl=64 time=5.05 ms
64 bytes from 192.168.100.1: icmp_seq=10 ttl=64 time=5.05 ms
[jaimeandresmedez@localhost ~]$ ping -c 10 192.168.100.1
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
5 packets transmitted, 5 received, 0% packet loss, time 4008ms
rtt min/avg/max/mdev = 1.31/2.183/5.069/1.444 ms
[jaimeandresmedez@localhost ~]$ ping -c 10 192.168.100.12
PING 192.168.100.12 (192.168.100.12) 56(84) bytes of data.
64 bytes from 192.168.100.12: icmp_seq=1 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=2 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=3 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=4 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=5 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=6 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=7 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=8 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=9 ttl=64 time=1.05 ms
64 bytes from 192.168.100.12: icmp_seq=10 ttl=64 time=1.05 ms
[jaimeandresmedez@localhost ~]$ ping -c 10 192.168.100.12
PING 192.168.100.12 (192.168.100.12) 56(84) bytes of data.
5 packets transmitted, 5 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 1.447/2.082/3.031/0.618 ms
[jaimeandresmedez@localhost ~]$
```

Aquí comprobamos conexión haciendo ping con las otras máquinas VM

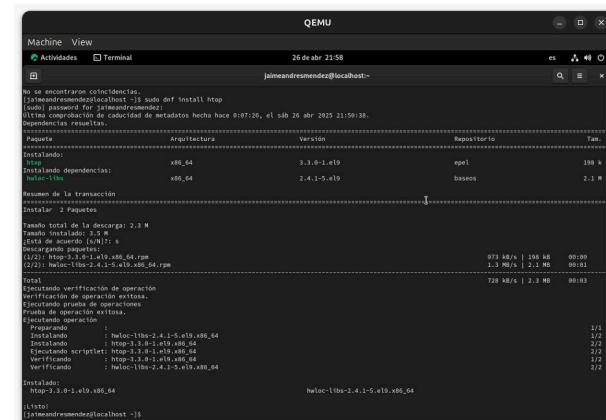
Desarrollar análisis de servicios y procesos con comandos “glances”, “bpytop” y “systemd/journalctl.”



Aquí instalamos GLANCES con éxito con
sudo dnf install glances



Aquí vemos la ejecución de GLANCES



Empezamos instalando BPYTOP, pero como no estuvo disponible, instalamos HTOP en su remplazo, cumple la misma función solo que con diferencia de interfaz

Aquí vemos la ejecución de HTOP con los procesos en funcionamiento

```
QEMU
Machine View
Actividades Terminal 27 de abr 12:54 es
Subject: A start job for unit system.slice has finished successfully
Support: https://wiki.debian.org/ready/support

[lines 1-14/165]
journalctl -b -1 --no-pager | grep -i typeservice
journalctl -b -1 --no-pager | grep -i typeservice
[1]: [  1.000]  systemd[1]: Starting LSB: Start/Stop system services
[1]: [  1.000]  /lib/systemd/system/dbus.service: main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: control process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running Deferred execution scheduler
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running Asynchronous message processor
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running Message, Install and Generate Color Profiles
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running Command Scheduler
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running D-Bus System Message Bus
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/dbus.service: loaded active running Firewall: dynamic firewall daemon
[1]: [  1.000]  /lib/systemd/system/dbus.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running GDM3 Display Manager
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running Triglance daemon
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Create List of Static Device Nodes
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Monitoring of 162 mirrors, snapshots etc. using devd or progress polling
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited ModemManager: ModemManager daemon
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running Network Manager
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running PackageKit Daemon
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Plymouth: Write Out Runtime Data
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running Authorization Manager
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running System Logging Service
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running OpenSSH server daemon
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running SDDM Xorg Cache Manager
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Automatic Root Loader update
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active running Journal Service
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Load Kernel Modules
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Load/Save OS Random Seed
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Load/Save SELinux File Systems
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
[1]: [  1.000]  /lib/systemd/system/gdm.service: loaded active exited Apply Kernel Variables
[1]: [  1.000]  /lib/systemd/system/gdm.service: Main process exited, code=exited, status=0/n/a
```

Aquí el funcionamiento de systemd/journalctl.

Aquí cogimos un servicio aleatorio para usar el comando anterior y analizar sus logs el cual fue nginx.service

-MANJARO: Esta será la etapa de la instalación de Manjaro, empezando en la terminal de Ubuntu por medio de la OEMU

Usas sudo (correcto para permisos).

Estableces memoria (-m 4096), CPUs (-smp 2), usas KVM (-enable-kvm).

Especificas el disco duro con ruta absoluta: -hda /home/gixxer155abs/Corte_2/PROYECTO2/VM_manjaro/manjaro.qcow2. ¡Asegúrate que el archivo se llame manjaro.qcow2 dentro de VM_manjaro!).

Especifica el CDROM con la ruta y nombre correctos del ISO: -cdrom

/home/gixxer155abs/Corte_2/PROYECTO2/manjaro-xfce-25.0.0-250414-linux612.iso.

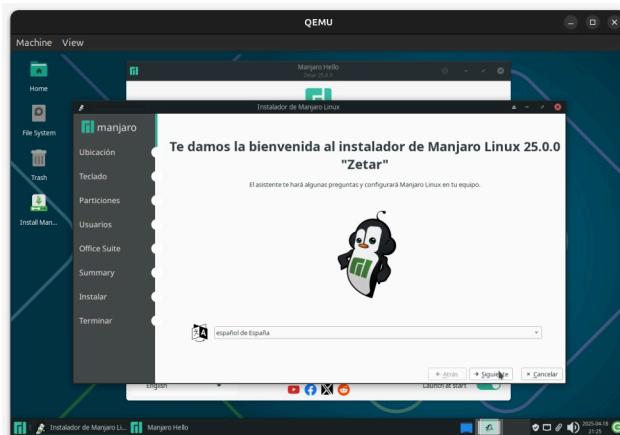
Arrancas desde CD: -boot d

Configuras video y pantalla: -vga virtio -display sdl.

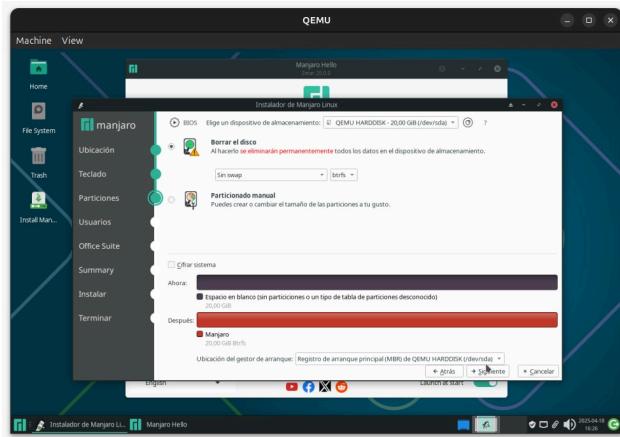
Configuración de Red: Aquí usas -netdev user,id=net0 -device e1000.netdev=net0. Esto configura User Mode Networking.

The screenshot shows a Manjaro Linux boot screen. The desktop environment is QEMU, indicated by the title bar. The desktop background features the Manjaro logo. On the left, there is a terminal window showing command-line output related to network interfaces (xxer155abs, id=net0, ,netdev=ne). The main window displays the GRUB boot menu. The highlighted option is "Boot with open source drivers". Other visible options include "Boot with proprietary drivers", "Edit boot options", "Help", and "Memory Test". At the bottom of the menu, it says "8s remaining." and provides keys for "E = Edit Boot Options" and "C = GRUB Commandline".

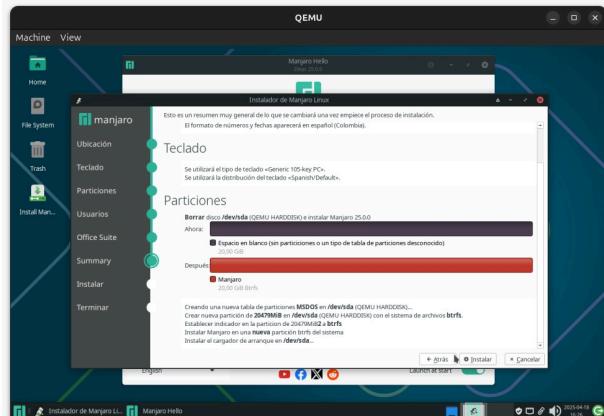
Aquí vemos que el instalador de Manjaro inicio de forma correcta, seleccionamos la opción Boot.



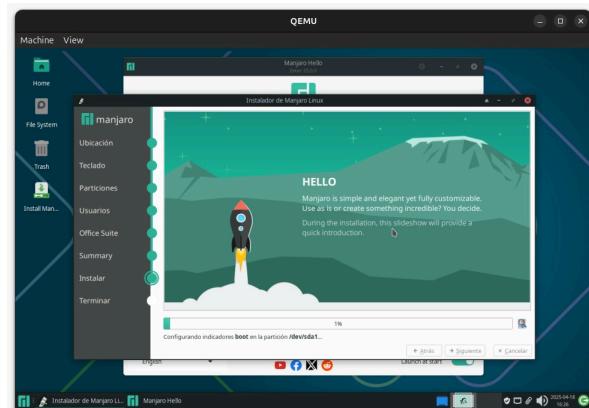
Damos siguiente



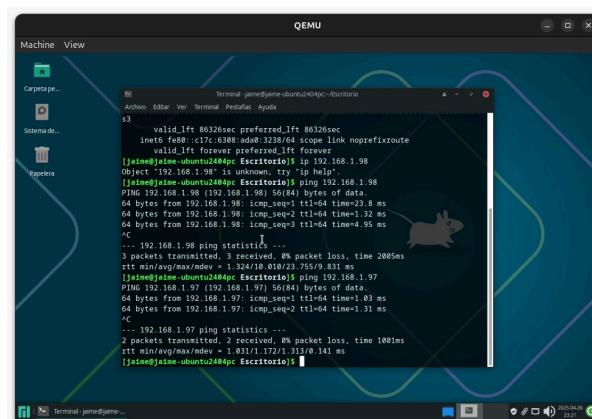
Siguiente, esta parte la dejamos por defecto



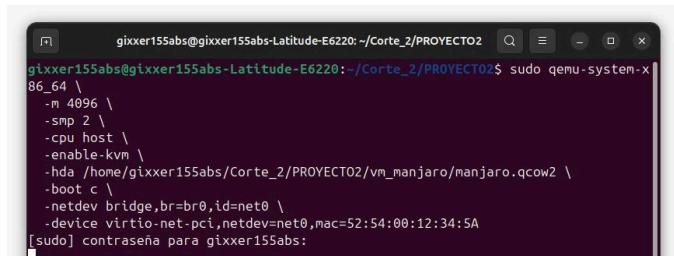
Igual forma dejamos por default



Y aqui vemos que el instalador empieza a correr de forma correcta



Aqui vemos que inicio correctamente Manjaro.



Ya despues iniciamos nuevamente Manjaro cambiando el directorio en vez de cd-rom donde estaba la ISO, ahora inicia en la partición virtual que creamos.

```
[jaime@jaime-ubuntu2404pc Escritorio]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 fe80::1%lo brd ff:ff:ff:ff:ff:ff scope link
                valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    link/ether 52:54:00:12:34:5a brd ff:ff:ff:ff:ff:ff
        altname enp0s3
        altname enx52540012345a
        inet 192.168.1.99/24 brd 192.168.1.255 scope global dynamic noprefixroute en3
            valid_lft 60326sec preferred_lft 86326sec
            inet6 fe80::c17c:6308%enp0s3 brd ff:ff:ff:ff:ff:ff scope link
                valid_lft forever preferred_lft forever
[jaime@jaime-ubuntu2404pc Escritorio]$
```

Luego le asignamos la IP de 192.168.1.99 pero luego la cambiamos por una como 192.168.100.10

```
[jaime@jaime-ubuntu2404pc Escritorio]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:12:34:5a brd ff:ff:ff:ff:ff:ff
        altname enp0s3
        altname enx52540012345a
        inet 192.168.100.10/24 brd 192.168.100.255 scope global noprefixroute ens3
            valid_lft forever preferred_lft forever
        inet6 fe80::cc2:bdd0%ens3 brd ff:ff:ff:ff:ff:ff scope link noprefixroute
            valid_lft forever preferred_lft forever
[jaime@jaime-ubuntu2404pc Escritorio]$
```

```
[jaime@jaime-ubuntu2404pc Escritorio]$ ping 192.168.1.98
PING 192.168.1.98 (192.168.1.98) 56(84) bytes of data.
64 bytes from 192.168.1.98: icmp_seq=1 ttl=64 time=23.8 ms
64 bytes from 192.168.1.98: icmp_seq=2 ttl=64 time=1.32 ms
64 bytes from 192.168.1.98: icmp_seq=3 ttl=64 time=4.95 ms
...
--- 192.168.1.98 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 1.324/10.819/23.775/9.831 ms
[jaime@jaime-ubuntu2404pc Escritorio]$ ping 192.168.1.97
PING 192.168.1.97 (192.168.1.97) 56(84) bytes of data.
64 bytes from 192.168.1.97: icmp_seq=1 ttl=64 time=1.83 ms
64 bytes from 192.168.1.97: icmp_seq=2 ttl=64 time=1.31 ms
...
--- 192.168.1.97 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.831/1.172/1.313/0.141 ms
[jaime@jaime-ubuntu2404pc Escritorio]$
```

Aquí comprobamos conexión haciendo ping con las otras máquinas VM. Con pequeño cambio de IP's luego de evaluar el gateway

```
[jaime@jaime-ubuntu2404pc Escritorio]$ ping 192.168.100.10
PING 192.168.100.10 (192.168.100.10) 56(84) bytes of data.
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 1.844/1.863/1.879/0.846 ms
[jaime@jaime-ubuntu2404pc Escritorio]$ ping 192.168.100.11
PING 192.168.100.11 (192.168.100.11) 56(84) bytes of data.
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 1.844/1.863/1.879/0.846 ms
[jaime@jaime-ubuntu2404pc Escritorio]$ ping 192.168.100.12
PING 192.168.100.12 (192.168.100.12) 56(84) bytes of data.
44 bytes from 192.168.100.12: icmp_seq=1 ttl=64 time=0.485 ms
44 bytes from 192.168.100.12: icmp_seq=2 ttl=64 time=0.485 ms
...
--- 192.168.100.12 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.485/0.485/0.485/0.000 ms
[jaime@jaime-ubuntu2404pc Escritorio]$
```

La ip que quedó fue la 192.168.100.10

Desarrollar análisis de red con comandos “wireshark”, “iftop” y “nethogs”.

```
[jaime@jaime-ubuntu2404pc Escritorio]$ pacman -Ss wireshark
extra/wireshark 2.4.0-2
    Terminal UI for tshark, inspired by Wireshark
extra/wireshark-cll 4.4.5-1
    Network traffic and protocol analyzer/sniffer - CLI tools and file files
extra/wireshark-qt 4.4.5-1
    Network traffic and protocol analyzer/sniffer - Qt GUI
[jaime@jaime-ubuntu2404pc Escritorio]$ sudo pacman -S wireshark
[jaime@jaime-ubuntu2404pc Escritorio]$ sudo pacman -S wireshark-qt
resolviendo dependencias...
:: Existen 2 proveedores disponibles para qt6-multimedia-backend:
:: Repositorio extra
  1) qt6-multimedia-ffmpeg 2) qt6-multimedia-gstreamer

Introduzca un número (por omisión=1): 1
buscando conflictos entre paquetes...
advertencia: se ha detectado un bucle de dependencias;
advertencia: qt6-multimedia-ffmpeg se instalará antes de su dependencia qt6-multimedia

Paquetes (12) bcg729-1.1.2-c-ares-1.34.4-1 libmaxminddb-1.2.2-lua53-5.3.6-3 minizip-1;1.3.1.2 qt6-5compat-6.9.0-0 qt6-multimedia-6.9.0-0.1 qt6-multimedia-ffmpeg-6.9.0-0.1 qt6-shadertools-6.9.0-1 wireshark-qt-4.4.5-1 zlib-ng-2.2.4-1 wireshark-qt-4.4.5-1

Tamaño total de la descarga: 27,99 MiB
Tamaño total de la instalación: 128,80 MiB

:: Continuar con la instalación? [S/n] s
:: Obteniendo los paquetes...
qt6-multimedia-ffmpeg-6.9.0-0.1-x86_64 1134,9 KiB 634 KiB/s 00:02: [=====] 100%
qt6-multimedia-ffmen-6.9.0-0-x86_64 576,0 KiB 1879 KiB/s 00:00: [=====] 100%
[1/1] The Wireshark Network... Terminal: jaime@jaime...
```

Aquí instalamos el Wireshark

```
[jaime@jaime-ubuntu2404pc Escritorio]$ pacman -Ss iftop
(4/5) updating icon theme Caches...
(5/5) updating the desktop file MIME type cache...
[jaime@jaime-ubuntu2404pc Escritorio]$ sudo pacman -S iftop
resolviendo dependencias...
buscando conflictos entre paquetes...

Paquetes (1) iftop-1.0px4-6

Tamaño total de la descarga: 0,03 MiB
Tamaño total de la instalación: 0,06 MiB

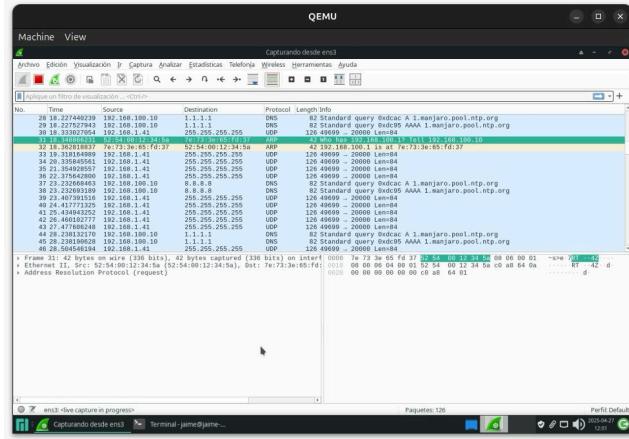
:: Continuar con la instalación? [S/n] s
:: Obteniendo los paquetes...
iftop-1.0px4-6-x86_64 29,4 KiB 32,8 KiB/s 00:01: [=====] 100%
(1/1) comprobando las claves del depósito
(1/1) integrando los cambios de los paquetes
(1/1) cargando los símbolos de los paquetes
(1/1) comprobando conflictos entre archivos
(1/1) comprobando el espacio disponible en el disco
(1/1) ejecutando las acciones de los paquetes...
(1/1) instalando iftop
:: Ejecutando los "hooks de postinstalación"...
(1/1) comprobando las claves del depósito
[jaime@jaime-ubuntu2404pc Escritorio]$ sudo pacman -S nethogs
resolviendo dependencias...
buscando conflictos entre paquetes...

Paquetes (1) nethogs-0.8-0.2

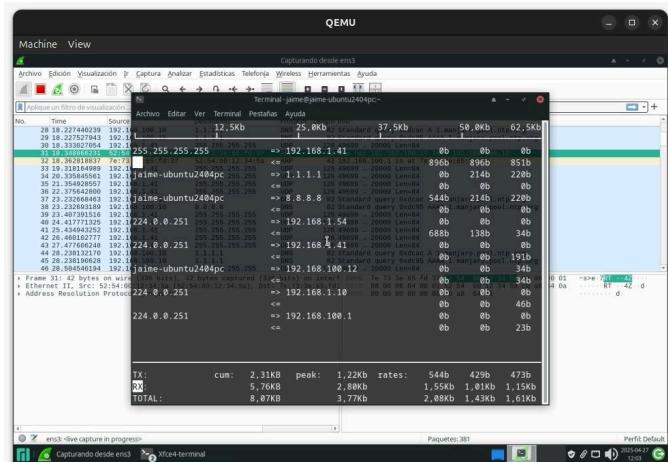
Tamaño total de la descarga: 0,05 MiB
Tamaño total de la instalación: 0,15 MiB

:: Continuar con la instalación? [S/n] s
:: Obteniendo los paquetes...
nethogs-0.8-0.2-x86_64 54,4 KiB 46,4 KiB/s 00:01: [=====] 100%
(1/1) comprobando las claves del depósito
(1/1) verificando la integridad del depósito
[1/1] The Wireshark Network... Terminal: jaime@jaime...
```

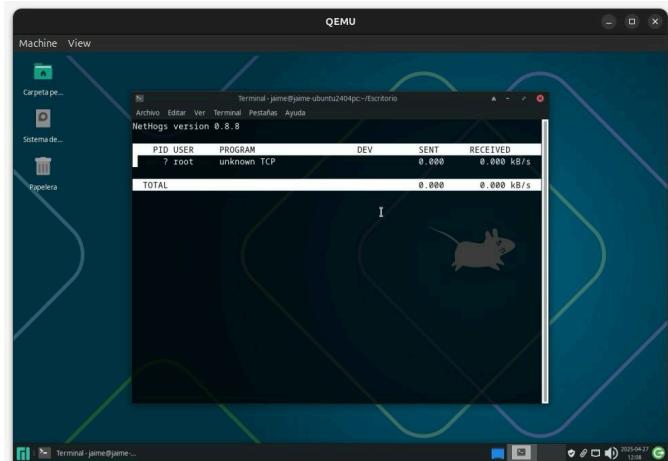
Aquí instalamos iftop y nethogs



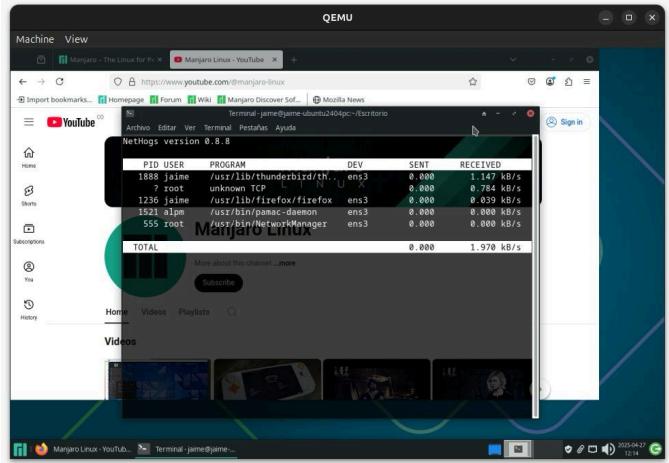
Iniciamos Wireshark y tomamos un paquete de protocolo ARP el cual se encarga de hacer la tabla de MAC y IP's de las otras pc's en la red, en este caso las otras VM's



Y aquí estamos usando iftop una herramienta de monitoreo de ancho de banda, el cual es insignificante en este caso por el poco tráfico que hay en este



De igual forma con Nethogs es nula.



B. Configuración del Entorno de Contenedores Docker y activades (Actividad 2)

Creación de red para contenedores:

```
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO1$ docker network create --subnet=172.28.0.0/24 red-monitoreo
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Head "http://127.0.0.1:2375/v1/networks/create"
f18232174bc9: dial unix /var/run/docker.sock: connect: permission denied
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO1$ sudo docker network create --subnet=172.28.0.0/24 red-monitoreo
d664048a982a1c02c1e02b6992345154b4ba01331d5bcb4d55a947ee927b19
```

La imagen muestra el proceso correcto para crear la red Docker personalizada (red-monitoreo) que se necesita para las Actividades 2 y 5 del proyecto. El primer intento falló por falta de permisos, y el segundo (con sudo) tuvo éxito, confirmando la creación de la red.

Ahora tenemos la red lista para conectar tus contenedores (Garuda, Alpine, Debian, y más tarde los de monitoreo) a ella.

ALPINE:

```
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO1$ docker pull alpine
Using default tag: latest
Digest: sha256:a856936eb8210634ff7d9f1f9ef7ffad463e380b75e2e74aff451df3ef88c
Status: Downloaded newer image for alpine:latest
f18232174bc9: Pulling from library/alpine
Digest: sha256:a856936eb8210634ff7d9f1f9ef7ffad463e380b75e2e74aff451df3ef88c
Status: Downloaded newer image for alpine:latest
dr54014fec98eef1f9291924fb18132d5f16d08d224be73d5f66de
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO1$ docker run -dit --name contenedor-alpine --network red-monitoreo alpine
sh: /etc/ld.so.cache: No such file or directory
dr54014fec98eef1f9291924fb18132d5f16d08d224be73d5f66de
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO1$
```

latest: Pulling from library/alpine: Docker procede a descargar automáticamente la imagen desde Docker Hub (el registro oficial).

f18232174bc9: Already exists: Una capa de la imagen ya existía localmente.

Digest: sha256:....: Identificador único de la imagen descargada.

Status: Downloaded newer image for alpine:latest: Confirma que la imagen se descargó correctamente.

df5430141fec...: Este es el ID largo del contenedor que se acaba de crear y ejecutar.

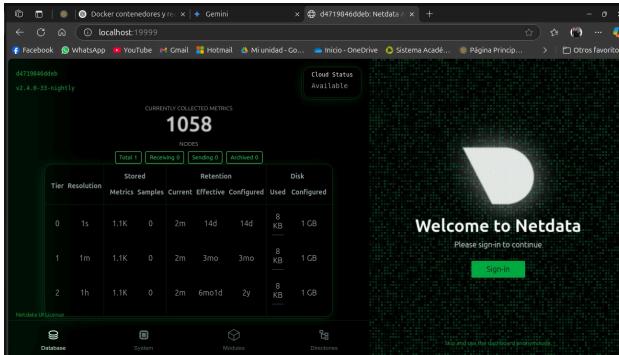
```
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
65b35277cd1        alpine              "tail -f /dev/null"   3 minutes ago    Exited (137) About a minute ago   contenedor-alpine
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker rm contenedor-alpine
contenedor-alpine
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker run -dit \
--network bridge \
-p 19999:19999 \
alpine sh
0@2280:be619xa9e9dd0b31b6e058f0526a75ad15b9d894c94f7aed2b08f9
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker exec -it contenedor-alpine netdata sh
# apk update
fetch https://dlcdn.alpinelinux.org/alpine/v3.21/main/x86_64/APKINDEX.tar.gz
fetch https://dlcdn.alpinelinux.org/alpine/v3.21/community/x86_64/APKINDEX.tar.gz
V: 25396 packages available (0 orphaned)
OK: 25396 distinct packages available
# apk upgrade
OK: 7 MB in 15 packages
# pinfo 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56 data bytes
64 bytes from 1.1.1.1: seq=8 ttl=54 time=67.882 ms
```

aquí actualizamos librerías y instalamos paquetes necesarios, dejando los puertos abiertos para visualizar NetData

```
Join our community and connect with us on:
- GitHub: https://github.com/netdata/netdata/discussions
- Discord: https://discord.gg/584f8R6
- Our community forums: https://community.netdata.cloud/
[root]# rm -rf /tmp/netdata-kickstart-XXXXP0Amp
OK

/ #
/ #
/ #
/ #
/ #
/ #
/ #
/ #
/ #
/ ps aux | grep netdata
539 root      0:00 [netdata]
504 netdata   0:00 /opt/netdata/bin/srv/netdata
542 netdata   0:00 /opt/netdata/plugins
828 root      0:00 [NETWORK-VIEWER] /opt/netdata/usr/libexec/netdata/plugins.d/network-viewer.plugin 1
841 root      0:00 /opt/netdata/usr/libexec/netdata/plugins.d/fact.plugin 1
868 root      0:00 /opt/netdata/usr/libexec/netdata/plugins.d/apps.plugin 1
870 netdata   0:00 /opt/netdata/usr/libexec/netdata/plugins.d/go.plugin 1
875 root      0:00 /opt/netdata/usr/libexec/netdata/plugins.d/debugfs.plugin 1
1122 root     0:00 grep netdata
/ # ss -tln | grep 19999
sh: ssi: not found
/ # netstat -tun | grep 19999
tcp        0      0 0.0.0.0:19999          0.0.0.0:*                LISTEN
tcp        0      0 ::1:19999             ::*                    LISTEN
/ #
```

Entramos en el localhost exactamente a la dirección 1999 y podremos observar NetData



Como podemos ver no hay muchos datos en transcurso ya que es una máquina sin programa alguno y tráfico es absoluto.

DEBIAN:

```
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker pull debian
Using default tag: latest
latest: Pulling from library/debian
23b7d26ef1d2: Pull complete
Digest: sha256:00cd4b40c4d9ff9f0c24548bbde0533ca3791edccad0de36d6b9fb3260d89e2
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$
```

```
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ sudo docker run -dit --name contenedor-debian --network red-monito
root@debian bash
[sudo] contraseña para gixer155abs:
Unable to find image 'debian:latest' locally
latest: Pulling from library/debian
23b7d26ef1d2: Pull complete
Digest: sha256:00cd4b40c4d9ff9f0c24548bbde0533ca3791edccad0de36d6b9fb3260d89e2
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$
```

La ejecución del comando sudo docker run en la terminal del sistema anfitrión (host) con el objetivo de crear e iniciar uno de los contenedores Docker requeridos por el proyecto, específicamente, el contenedor basado en la imagen alpine.

sudo docker run: Inicia la creación y ejecución de un nuevo contenedor Docker con privilegios de superusuario, necesarios para interactuar con el demonio Docker.

-it: Ejecuta el contenedor en modo interactivo (-i) asignando una pseudo-terminal (-t), permitiendo la interacción directa con el shell del contenedor.

--name contenedor-alpine: Asigna un nombre legible (contenedor-alpine) al contenedor para facilitar su gestión posterior.

--network red-monitoreo: Conecta el nuevo contenedor a la red Docker personalizada red-monitoreo previamente creada. Esta acción es fundamental para asegurar que el contenedor opere dentro de la subred designada para los servicios contenerizados y pueda comunicarse con otros contenedores en la misma red, como los servicios de monitoreo.

alpine: Especifica la imagen base que se utilizará para crear el contenedor. En este caso, se utiliza la imagen oficial alpine:latest, conocida por su pequeño tamaño y enfoque minimalista.

sh: Indica el comando que se ejecutará una vez que el contenedor se inicie, en este caso, el shell sh, que es el shell estándar en Alpine Linux.

Unable to find image 'alpine:latest' locally: El sistema informa que la imagen alpine:latest no se encuentra descargada en la máquina host.

latest: Pulling from library/alpine: Como consecuencia, Docker procede a descargar (hacer "pull") la imagen desde el registro oficial Docker Hub (library/alpine).

f18... Already exists, Digest:..., Status: Downloaded newer image...: Mensajes estándar que indican el progreso y la finalización exitosa de la descarga de las capas de la imagen.

df543...: Muestra el ID largo y completo del contenedor recién creado y ejecutado.

```
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker pull debian
Using default tag: latest
latest: Pulling from library/debian
23b7d26ef1d2: Already exists
Digest: sha256:00cd4b40c4d9ff9f0c24548bbde0533ca3791edccad0de36d6b9fb3260d89e2
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
gixer155abs@gixer155abs-Latitude-E6220:~/Corte_2/PROYECTO$ docker run -dit --name contenedor-debian debian bash
root@36d6b9fb3260d89e2:/# curl https://deb.debian.org/debian/bookworm/InRelease
root@36d6b9fb3260d89e2:/# curl https://deb.debian.org/debian/bookworm-updates/InRelease
root@36d6b9fb3260d89e2:/# curl https://deb.debian.org/debian-security/bookworm/InRelease
root@36d6b9fb3260d89e2:/# curl https://deb.debian.org/debian-security/bookworm-updates/InRelease
root@36d6b9fb3260d89e2:/# apt update
Get:1 https://deb.debian.org/debian/binary-amd64/InRelease [131 kB]
Get:2 https://deb.debian.org/debian/bookworm/InRelease [155.4 kB]
Get:3 https://deb.debian.org/debian/bookworm-updates/InRelease [48.0 kB]
Get:4 https://deb.debian.org/debian/bookworm/main/amd64 Packages [8792 kB]
Get:5 https://deb.debian.org/debian/bookworm-updates/main/amd64 Packages [512 B]
Get:6 https://deb.debian.org/debian-security/bookworm-security/InRelease [104 kB]
Get:7 https://deb.debian.org/debian-security/bookworm-security/main/amd64 Packages [260 kB]
Fetched 9307 kB in 1s (6297 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading information for installed packages... Done
1 package can be upgraded, run 'apt list --upgradable' to see it.
root@36d6b9fb3260d89e2:/# apt upgrade
Reading package lists... Done
Reading dependency information... Done
Reading status information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  perl-base
1 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 1699 kB of archives.
```



```

docker [o/library/archlinux:latest
gixxer155abs@lxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ docker run -it --name garuda-like archlinux
:: Synchronizing package databases...
core downloading...
extra downloading...
:: starting full system upgrade...
resolving dependencies...
looking for conflicting packages...
Package (4) Old Version New Version Net Change Download Size
core/glibc 2.41-r9+ge990dbaf7ff0-1 2.41-r47+g0d6b33800c3e-1 0.02 MiB 9.96 MiB
core/libcpg-error 1.53-1 1.54-1 0.00 MiB 0.26 MiB
core/libtpmtp3 1.8.0-1 1.9.0-1 0.00 MiB 0.07 MiB
core/libxml2 2.13.7-1 2.13.8-1 0.00 MiB 0.80 MiB
Total Download Size: 11.89 MiB
Total Installed Size: 52.91 MiB
Net Upgraded Size: 0.02 MiB
:: Proceed with installation? [Y/n] y
:: Retrieving packages...
glIBC-2.41-r47+g0d6b33800c3e-1-x86_64 downloading...
libxml2-2.13.8-1-x86_64 downloading...
libcpg-error-1.54-1-x86_64 downloading...
libtpmtp3-1.9.0-1-x86_64 downloading...
checking keyring...
checking package integrity...
loading package files...
checking for file conflicts...

```

En estas dos imágenes, se aprecia cuando descargamos una imagen de arch y luego construimos la imagen como si fuera Garuda, luego usamos el comando pacman -Syu para actualizar librerías

```

[root@63c74c854084/]# pacman -S neofetch zsh base-devel git nano
:: resolving dependencies...
looking for conflicting packages...
Package (34) New Version Net Change Download Size
core/mesonconf 2.77-1 2.77 MiB 0.64 MiB
core/autonake 1.17-1 1.62 MiB 0.61 MiB
core/binutils 2.44-1 41.78 MiB 7.71 MiB
core/blison 3.8.2-8 2.52 MiB 0.75 MiB
core/dbus5 5.3.28-5 6.45 MiB 1.18 MiB
core/designtool 5.1.1-1 0.10 MiB 0.00 MiB
core/failfutils 3.19-1 1.33 MiB 0.34 MiB
core/fakeroot 1.37.1.1-1 0.14 MiB 0.08 MiB
core/flex 2.6.4-5 0.93 MiB 0.30 MiB
core/gc 8.2.8-2 0.76 MiB 0.23 MiB
core/gc 14.2.1-r753+g1cd7446628f-1 202.42 MiB 51.88 MiB
core/goff 1.1.0-1 0.44 MiB 2.25 MiB
core/guile 3.0.10-1 54.73 MiB 8.31 MiB
core/jansson 2.14.1-1 0.27 MiB 0.09 MiB
core/liblsl 0.27-1 5.75 MiB 0.87 MiB
core/libmpc 1.3.1-2 0.18 MiB 0.00 MiB
core/libtboot 2.27.1-1 0.47 MiB 0.41 MiB
core/m4 1.4.19-3 0.47 MiB 0.24 MiB
core/make 4.4.1-2 1.67 MiB 0.51 MiB
core/patch 2.8-1 0.17 MiB 0.08 MiB
core/perl 5.40.2-1 71.35 MiB 19.98 MiB
extra/cmake 0.9.4.1-1 0.00 MiB 0.00 MiB
extra/perl-maillist 2.22-1 0.10 MiB 0.06 MiB
extra/perl-timedate 2.33-7 0.08 MiB 0.03 MiB
core/pkgconf 2.4.3-1 0.17 MiB 0.06 MiB

```

Seguimos instalando mas librerías necesarias para la práctica

```

[59/5] Checking for old perl modules...
[root@63c74c854084/]# pacman -S lshw inxi util-linux systemd --noconfirm
warning: util-linux-2.41-4 is up to date -- reinstalling
warning: systemd-257.5-2 is up to date -- reinstalling
:: resolving dependencies...
looking for conflicting packages...
Package (4) Old Version New Version Net Change Download Size
extra/inxi 3.3.38.1-1 1.34 MiB 0.34 MiB
extra/lshw B.02.20-1 8.08 MiB 1.60 MiB
core/systemd 257.5-2 257.5-2 0.00 MiB 8.88 MiB
core/util-linux 2.41-4 2.41-4 0.00 MiB 5.06 MiB
Total Download Size: 15.81 MiB
Total Installed Size: 68.32 MiB
Net Upgraded Size: 9.42 MiB
:: Proceed with installation? [Y/n] y
:: Retrieving packages...
systemd-257.5-2-x86_64 downloading...
util-linux-2.41-4-x86_64 downloading...
lshw-B.02.20-x86_64 downloading...
inxi-3.3.38.1-x86_64 downloading...
checking keyring...
checking package integrity...
loading package files...
checking for file conflicts...
:: Processing package changes...
installing lshw...

```

Paquetes Involucrados:

tnx1 (3.3.38.1-1)

Tamaño: 1.34 MiB (descarga), 0.34 MiB de cambio neto

Parece ser un paquete de los repositorios "extra"

lshw (8.02.20-1)

Tamaño: 8.08 MiB (descarga), 1.60 MiB de cambio neto

Herramienta para listar hardware (Hardware Lister)

systemd (257.5-2)

Versión actual y nueva: 257.5-2 (reinstalación)

Sistema de init y administración de servicios

Tamaño: 8.80 MiB (descarga), sin cambio neto
util-linux (2.41-4)

Versión actual y nueva: 2.41-4 (reinstalación)

Colección de utilidades básicas

Tamaño: 5.06 MiB (descarga), sin cambio neto

```

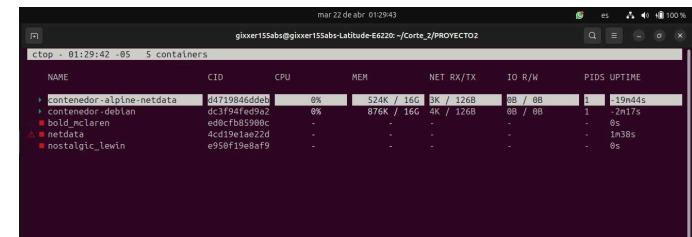
[root@63c74c854084/]# lshw | less
bash: less: command not found
[root@63c74c854084/]# pacman -S less --noconfirm
resolving dependencies...
looking for conflicting packages...
Package (1) New Version Net Change Download Size
core/less 1:668-1 0.29 MiB 0.14 MiB
Total Download Size: 0.14 MiB
Total Installed Size: 0.29 MiB
:: Proceed with installation? [Y/n]
:: Processing dependencies...
less-1:668-1-x86_64 downloading...
checking keyring...
checking package integrity...
loading package files...
checking for file conflicts...
:: Processing package changes...
installing less...
:: Running post-transaction hooks...
(1/1) Arming ConditionNeedsUpdate...
[root@63c74c854084/]# lshw | less
[root@63c74c854084/]#

```

```

gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ docker stop contenedor-debian
contenedor-debian
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ ^C
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ docker start contenedor-debian
Contenedor arrancado
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ docker exec -it contenedor-debian bash
root@dc3f04fed9a2:/# exit
exit
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ docker start contenedor-alpine-netdata
contenedor-alpine-netdata
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$ curl -s https://raw.githubusercontent.com/bciken/ctop/master/insta
ll.sh | bash
curl: (29) Could not connect to https://raw.githubusercontent.com/bciken/ctop/master/insta...: Failed to connect to raw.githubusercontent.com port 443: Connection refused
* Connection timed out after 1000ms
* Closing connection 0
stop-install fetching latest release info
stop-install fetching release checksums
stop-install fetching latest ctop
stop-install installing ctop-0.9.1-1_amd64
stop-install installing to /usr/local/bin
[sudo] contraseña para gixxer155abs:
stop-install done!
gixxer155abs@gixxer155abs-Latitude-E6220:~/Corte_2/PROYECTOS$
```

Aquí instalamos CTOP



Usamos CTOP y vemos los detalles de los contenedores, como consumo de CPU, memoria y de red, falta Arch linux por el hecho de que aun no estaba descargada cuando se ejecuto CTOP

```
gioxer155abs@giixer155abs-Latitude-E8220:~/Corte_2/PROYECTO2
```

Logs - Estadísticas - Variables de entorno - Configuración - Top

```
Proyecto PROYECTO2
```

```
Contenedores
```

Nombre	Imagen	Estado	CPU
contenedor-alpine-netdata	0.0	running	0.0
contenedor-debian	0.0	exited (0)	bold_mclaren
netdata	0.0	exited (0)	netdata
nostalgic_lewin	0.0	exited (0)	nostalgic_lewin

```
Imagenes
```

Imagen	Tamaño
alpine	latest 7.83MB
debian	latest 116.57MB
hello-world	latest 10.07KB
netdata/netdata	latest 780.20MB

```
Volumenes
```

```
Redes
```

Nombre	Driver	Host	Mode
bridge	bridge	host	host
host	host	host	none
null	null	host	none
bridge	red-monitree	host	bridge

Felpa/Puñal scroll, b/ i ver comandos masivos, q: salir, x: menu, -- !: navegar

Donar 0.24.1

Aquí vemos el funcionamiento de Lazy docker, aunque no se ve mucho que digamos.

Aquí vemos el Dockerfile con el cual se construyó el contenedor central

```
[]
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network connect bridge Fedora_Central
Error response from daemon: could not find a network matching network node red-monitorreto: network red-monitorreto not found
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network create \
--driver=macvlan \
--subnet=192.168.100.0/24 \
--gateway=192.168.100.1 \
--ip-range=192.168.100.128/25 \
--aux-address 'host:192.168.100.1' \
--parent=br0
red-monitorreto
Error response from daemon: failed to allocate secondary ip address (host:192.168.100.1): Address already in use
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network create \
--driver=macvlan \
--subnet=192.168.100.0/24 \
--gateway=192.168.100.1 \
--ip-range=192.168.100.128/25 \
--parent=br0
red-monitorreto
66aa2ab5b564627f3aa181432e4354756e815f77dfabc209d3c886ca9c5752f7a
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
159661544c2     bridge    bridge      local
7242b7145b      host      host       local
ac4fc916e6d2    none     macvlan   local
66aa2ab5b5646  red-monitorreto  macvlan  local
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network connect red-monitorreto Fedora_Central
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network connect red-monitorreto contenedor-alpine-netdata
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker network connect red-monitorreto garrigue-like
gixxer15Sabs@giixer15Sabs-Latitude-E6220:~/Corte_2/PROYECTO$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
845443333333        garrigue           "tail -f /dev/null"   2 minutes ago      Up 2 minutes

```

configuración de los contendores en la red

```
On Jun 27 06:01 17:53:12  
gixxer155abs@gixxer155abs-Latitude-E6220: ~/Corte_2/P  
{"Ingress": false,  
 "ConfigFrom": {  
     "Network": ""  
 },  
 "ConfigOnly": false,  
 "Containers": [  
     "63c74c540847524c32386e4565cfed398d4817d2f87c89f0c14955b1874": {  
         "Name": "contenedor-debian",  
         "EndpointID": "c9cb0930ba5a75e25dc8a700529f1ae2f7d8b96e1b2d4c1c182047c48fe542",  
         "MacAddress": "42:82:b9:5d:3c:e8",  
         "IPv4Address": "192.168.108.136/24",  
         "IPv6Address": ""  
     },  
     "fb980c8124c7269b7b710787478d677e208d71e1672d239e19e968c3b4a": {  
         "Name": "contenedor-debian",  
         "EndpointID": "e9b05815c3258a4c3c445fb0b0362c82e3195a3acfda52a9fb9e693d0a0ac435",  
         "MacAddress": "5a:b9:45:e2:b8:59",  
         "IPv4Address": "192.168.108.131/24",  
         "IPv6Address": ""  
     },  
     "d4719846deb3ea8eacf2e95fd076857b536ed1d428a29daaf768e0a67d2c2": {  
         "Name": "contenedor-alpine-netdata",  
         "EndpointID": "be32c2d4369a3a5fc046c5286cf27745ad24effea4da28e3a5fb6bb7e67",  
         "MacAddress": "56:88:d4:01:00:85",  
         "IPv4Address": "192.168.108.129/24",  
         "IPv6Address": ""  
     },  
     "dd7d37c9015db98ab0fd249946de991c3d6aab084857e9ff8634fe47fcbb7": {  
         "Name": "Fedorax_Central",  
         "EndpointID": "1027924ed48f4e0e0c21dcb34fe3317cf8e4ec4c68fb913a9467d217e1aa98a6",  
         "MacAddress": "00:0c:88:d4:3f:f1",  
         "IPv4Address": "192.168.108.128/24",  
         "IPv6Address": ""  
     }  
 },  
 "Options": {  
     "parent": "br0"  
 },  
 "Labels": {}  
 }  
 ]  
 }  
 gixxer155abs@gixxer155abs-Latitude-E6220: ~/Corte_2/PROYECTO/S
```

C. Configuración e Instalación contenedor central Fedora y actividades (Actividad 3)

Instalación de Fedora y construcción de esta como contenedor central

Aquí comprobamos que los 4 contenedores están conectados a la red-monitoreo de los contenedores para analizar y monitorizar desde Fedora su tráfico y demás

```
Abrir ▾ dockerfile
~/Corte_2/PROYECTO2

# Usa Fedora como imagen base
FROM fedora:latest

# Actualiza los repositorios e instala las herramientas necesarias
RUN dnf -y update && \
    dnf -y install \
        htop \
        net-tools \
        iputils \
        python3 \
        git && \
    dnf clean all

# Establece el directorio de trabajo
WORKDIR /root

# Comando predeterminado para cuando el contenedor arranque
CMD ["bash"]
```

Tambien realizamos un ping entre ellas

```
[]
gixer155abs@gixer155abs-Latitude-E6220:/Corte_2/PROYECTO]$ docker exec -it Fedora_Central bash
[root@dd7d37c90125d ~]# which nmap
/usr/sbin/nmap
[root@dd7d37c90125d ~]# which http
/usr/sbin/http
[root@dd7d37c90125d ~]# nmap -sn 192.168.100.6/24
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 22:35 UTC
Nmap scan report for gixer155abs-Latitude-E6220 (192.168.100.1)
Host is up (0.1s latency).
MAC Address: 8C:70:5A:00:53:48 (Intel Corporate)
Nmap scan report for 192.168.100.10
Host is up (0.00049s latency).
MAC Address: 52:54:00:12:34:5A (QEMU virtual NIC)
Nmap scan report for 192.168.100.11
Host is up (0.00027s latency).
MAC Address: 52:54:00:12:34:5C (QEMU virtual NIC)
Nmap scan report for 192.168.100.12
Host is up (0.00066s latency).
MAC Address: 52:54:00:12:34:5B (QEMU virtual NIC)
Nmap scan report for contenedor-alpine-netdata.red-monitoreo (192.168.100.129)
Host is up (0.000030s latency).
MAC Address: CE:52:47:67:8B:85 (Unknown)
Nmap scan report for garuda-like.red-monitoreo (192.168.100.130)
Host is up (0.00065s latency).
MAC Address: 42:02:89:5D:3C:EE (Unknown)
Nmap scan report for contenedor-debian.red-monitoreo (192.168.100.131)
Host is up (0.000028s latency).
MAC Address: MA:89:45:E2:B3:58 (Unknown)
Nmap scan report for dd7d37c90125d (192.168.100.128)
Host is up.
Nmap done: 256 IP addresses (8 hosts up) scanned in 2.20 seconds
[root@dd7d37c90125d ~]#
```

Usamos NMap y vemos reflejado los 3 contenedores y las 3 VM's

```
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.58 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.12
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:19 UTC
Nmap scan report for 192.168.100.12
Host is up (0.00063s latency).
All 1000 scanned ports on 192.168.100.12 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 52:54:00:12:34:5B (QEMU virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.66 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.11
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:19 UTC
Nmap scan report for 192.168.100.11
Host is up (0.0013s latency).
Not shown: 988 filtered tcp ports (no-response), 10 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh   OpenSSH 8.7 (protocol 2.0)
9090/tcp  closed  zeus-admin

MAC Address: 52:54:00:12:34:5C (QEMU virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 5.66 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.10
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:19 UTC
Nmap scan report for 192.168.100.10
Host is up (0.00056s latency).
All 1000 scanned ports on 192.168.100.10 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 52:54:00:12:34:5A (QEMU virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.71 seconds
[root@dd7d37c90125d ~]#
```

Nmap a detalles de cada IP de las maquinas virtuales

```
nmap done: 1 IP address (0 hosts up) scanned in 1.91 seconds
[root@dd7d37c90125d ~]# nmap -p_ 192.168.100.128
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:14 UTC
Nmap scan report for dd7d37c90125d (192.168.100.128)
Host is up (0.000012s latency).
All 65535 scanned ports on dd7d37c90125d (192.168.100.128) are in ignored states.
Not shown: 65535 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 1.63 seconds
[root@dd7d37c90125d ~]# nmap -p_ 192.168.100.129
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:14 UTC
Nmap scan report for contenedor-alpine-netdata.red-monitoreo (192.168.100.129)
Host is up (0.000014s latency).
All 65535 scanned ports on contenedor-alpine-netdata.red-monitoreo (192.168.100.129) are in ignored states.
Not shown: 65535 closed tcp ports (reset)
MAC Address: CE:52:47:67:8B:85 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.74 seconds
[root@dd7d37c90125d ~]# nmap -p_ 192.168.100.130
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:15 UTC
Nmap scan report for garuda-like.red-monitoreo (192.168.100.130)
Host is up (0.000014s latency).
All 65535 scanned ports on garuda-like.red-monitoreo (192.168.100.130) are in ignored states.
Not shown: 65535 closed tcp ports (reset)
MAC Address: 42:02:89:5D:3C:EE (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.71 seconds
[root@dd7d37c90125d ~]# nmap -p_ 192.168.100.131
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:15 UTC
Nmap scan report for contenedor-debian.red-monitoreo (192.168.100.131)
Host is up (0.000014s latency).
All 65535 scanned ports on contenedor-debian.red-monitoreo (192.168.100.131) are in ignored states.
Not shown: 65535 closed tcp ports (reset)
MAC Address: 5A:89:45:E2:B3:58 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.71 seconds
[root@dd7d37c90125d ~]#
```

Nmap a detalles de cada IP de los contenedores

```
[]
@dd7d37c90125d:~#
Nmap done: 1 IP address (1 host up) scanned in 1.71 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.131
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:17 UTC
Nmap scan report for garuda-like.red-monitoreo (192.168.100.131)
Host is up (0.00015s latency).
All 1000 scanned ports on garuda-like.red-monitoreo (192.168.100.131) are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 42:02:89:5D:3C:EE (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.63 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.130
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:18 UTC
Nmap scan report for garuda-like.red-monitoreo (192.168.100.130)
Host is up (0.000014s latency).
All 1000 scanned ports on garuda-like.red-monitoreo (192.168.100.130) are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 42:02:89:5D:3C:EE (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.61 seconds
[root@dd7d37c90125d ~]# nmap -sv 192.168.100.129
Starting Nmap 7.92 ( https://nmap.org ) at 2025-04-27 23:18 UTC
Nmap scan report for contenedor-alpine-netdata.red-monitoreo (192.168.100.129)
Host is up (0.000014s latency).
All 1000 scanned ports on contenedor-alpine-netdata.red-monitoreo (192.168.100.129) are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: CE:52:47:67:8B:85 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.58 seconds
[root@dd7d37c90125d ~]#
```

D. (Actividad Número 4) Desarrollo de análisis de red y procesos por medio del contenedor central.

Primera actividad: Escanear los puertos abiertos tanto en máquinas virtuales,

```
[root@dd7d37c90125d ~]# yum install -y procps-ng
Updating and loading repositories:
Repositories loaded.
Package           Arch      Version
Installing:
procps-ng          x86_64   4.0.4-6.fc42
Transaction Summary:
Installing:       1 package
Total size of inbound packages is 365 Kib. Need to download 365 KiB.
After this operation, 1 MiB extra will be used (install 1 MiB, remove 0 B).
[1/1] procps-ng-0:4.0.4-6.fc42.x86_64
-----
[1/1] Total
Running transaction
[1/3] Verify package files
[2/3] Prepare transaction
[3/3] Installing procps-ng-0:4.0.4-6.fc42.x86_64
Complete!
[root@dd7d37c90125d ~]# ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0  4788 3964 pts/0    S+  21:32  0:00 bash
root        20  0.0  0.0  4788 3906 pts/1    S+  22:05  0:00 bash
root        44  0.0  0.0  4788 4044 pts/2    S+  22:14  0:00 bash
root       162  0.0  0.0  4788 3944 pts/3    S+  22:22  0:00 bash
root       187  0.0  0.0  4788 3932 pts/4    S+  22:35  0:00 bash
root       245  0.0  0.0  6896 3888 pts/4    R+  22:51  0:00 ps aux
[root@dd7d37c90125d ~]#
```

Aqui se uso e instaló el ps aux viendo los puertos abiertos en el sistema

```
[root@dd7d37c90125d ~]# traceroute 192.168.100.131
traceroute to 192.168.100.131 (192.168.100.131), 30 hops max, 60 byte packets
  1 contenedor-debian.red-monitoreo (192.168.100.131)  0.045 ms  0.924 ms  0.880 ms
traceroute to 192.168.100.130 (192.168.100.130), 30 hops max, 60 byte packets
  1 garuda-like.red-monitoreo (192.168.100.130)  0.712 ms  0.573 ms  0.515 ms
[root@dd7d37c90125d ~]# traceroute 192.168.100.129
traceroute to 192.168.100.129 (192.168.100.129), 30 hops max, 60 byte packets
  1 contenedor-alpine-netdata.red-monitoreo (192.168.100.129)  0.978 ms  0.824 ms  0.720 ms
[root@dd7d37c90125d ~]# traceroute 192.168.100.128
traceroute to 192.168.100.128 (192.168.100.128), 30 hops max, 60 byte packets
  1 dd7d37c90125d (192.168.100.128)  0.088 ms  0.032 ms  0.023 ms
[root@dd7d37c90125d ~]# traceroute 192.168.100.127
traceroute to 192.168.100.127 (192.168.100.127), 30 hops max, 60 byte packets
  1 192.168.100.10 (192.168.100.10)  1.566 ms  1.600 ms  1.595 ms
[root@dd7d37c90125d ~]# traceroute 192.168.100.11
traceroute to 192.168.100.11 (192.168.100.11), 30 hops max, 60 byte packets
  1 192.168.100.11 (192.168.100.11)  0.847 ms  IX  0.776 ms  IX  0.735 ms  IX
[root@dd7d37c90125d ~]# traceroute 192.168.100.11
traceroute to 192.168.100.11 (192.168.100.11), 30 hops max, 60 byte packets
  1 192.168.100.12 (192.168.100.12)  1.409 ms  1.368 ms  1.324 ms
[root@dd7d37c90125d ~]#
```

En este se uso traceroute y como todos estos sistemas estan dentro del host comparten las mismas rutas

Tercera actividad: Usar netstat -tulpn en las máquinas virtuales y

contenedores y comparar resultados de lo visualizado.

```
[jaimeandremendez@localhost ~]$ sudo netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State      PID/Program name
tcp     0      0    127.0.0.1:631          0.0.0.0:*          LISTEN    849/cupsd
tcp     0      0    0.0.0.0:22            0.0.0.0:*          LISTEN    850/sshd: /usr/sbin
tcp6    0      0    ::1:631               ::*:*              LISTEN    851/sshd: /usr/sbin
tcp6    0      0    ::1:22                ::*:*              LISTEN    851:sshd: /usr/sbin
udp    0      0    0.0.0.0:36503        0.0.0.0:*          LISTEN    714/avahi-daemon: r
udp    0      0    127.0.0.1:323          0.0.0.0:*          LISTEN    741/chrony
udp    0      0    0.0.0.0:5353          0.0.0.0:*          LISTEN    714/avahi-daemon: r
udp6   0      0    ::1:323              ::*:*              LISTEN    741/chrony
udp6   0      0    ::0:55876            ::*:*              LISTEN    714/avahi-daemon: r
udp6   0      0    ::0:5353             ::*:*              LISTEN    714/avahi-daemon: r

[jaimeandremendez@localhost ~]$
```

Uso de sudo netstat-tulpn en RockyLinux

```
valid_lft forever preferred_lft forever
[archaim3@Archaim3 ~]$ sudo netstat -tulpn
[sudo] contraseña para archaim3:
Active Internet connections (only servers)
Proto Recv-Q Local Address           Foreign Address         State      PID/Program name
tcp     0    0.0.0.0.54..53          0.0.0.0.*              LISTEN    284/systemd-resolve
tcp     0    0.0.0.0.5355          0.0.0.0.*              LISTEN    284/systemd-resolve
tcp     0    0.0.0.0.53..53          0.0.0.0.*              LISTEN    284/systemd-resolve
tcp6    0    :::::5355             :::::*                  LISTEN    284/systemd-resolve
udp     0    0.0.0.0.5353          0.0.0.0.*              LISTEN    284/systemd-resolve
udp     0    0.0.0.0.5355          0.0.0.0.*              LISTEN    284/systemd-resolve
udp     0    0.0.0.0.38415         0.0.0.0.*              LISTEN    1914/python3
udp     0    0.192.168.1.198:3702   0.0.0.0.*              LISTEN    1914/python3
udp     0    0.293.265.255.250:3702 0.0.0.0.*              LISTEN    1914/python3
udp     0    0.0.0.0.54..53          0.0.0.0.*              LISTEN    284/systemd-resolve
udp     0    0.127.0.0.53..53        0.0.0.0.*              LISTEN    284/systemd-resolve
udp6    0    :::::5353             :::::*                  LISTEN    284/systemd-resolve
udp6    0    :::::5355             :::::*                  LISTEN    284/systemd-resolve
udp6    0    fe80::b1e1:3df7:d4:3702 :::::*                  LISTEN    1914/python3
udp6    0    ff02::1:c:3702         :::::*                  LISTEN    1914/python3
udp6    0    0.0.0.0.54904         0.0.0.0.*              LISTEN    1914/python3
[archaim3@Archaim3 ~]$
```

Uso de sudo netstat-tulpn en ArchLinux

```
[jaime@jaime-ubuntu-2404pc Escritorio]$ sudo netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp     0      0    127.0.0.1:631           0.0.0.0:*          LISTEN    645/cupsd
tcp6    0      0    ::1:631                ::*:*              LISTEN    645/cupsd
udp     0      0    0.0.0.0:5353           0.0.0.0:*          LISTEN    556/avahi-daemon: r
udp     0      0    0.0.0.0:55366          0.0.0.0:*          LISTEN    556/avahi-daemon: r
udp6    0      0    ::0:48086              ::*:*              LISTEN    556/avahi-daemon: r
udp6    0      0    ::0:5353               ::*:*              LISTEN    556/avahi-daemon: r

[jaime@jaime-ubuntu-2404pc Escritorio]$
```

Uso de sudo netstat-tulpn en Manjaro

```
g1xker155abs@lxker155abs-Latitude-E6220:~/Corte_2/PROYECTO2$ docker exec -it Fedora_Central bash
[root@d7d37c90125 ~]# sudo netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp     0      0    127.0.0.1:33889      0.0.0.0:*           LISTEN    
tcp     0      0    127.0.0.1:54868      0.0.0.0:*           LISTEN    
[root@d7d37c90125 ~]#
```

Uso de sudo netstat-tulpn en Fedora

```
gkxwef455abebfux155s-latitude-E6228:/~Corte_Z/PROYECTO$ docker exec -it contenedor-alpine-alpine-netdata bash  
d4719846deb# netstat -tulpn  
Active Internet connections (only servers)  
Proto Recv-Q Send-Q Local Address          Foreign Address      State       PID/Program name  
tcp        0      0    127.0.0.1:42893        0.0.0.0:*           LISTEN      -  
udp       0      0    127.0.0.1:41689        0.0.0.0:*           -  
d4719846deb#
```

Uso de sudo netstat-tulpn en Alpine

```
glixer155b@glixer155b:~$ docker exec -it garuda-like bash
[root@63c74c854084 ~]# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp     0      0    127.0.0.11:46623      0.0.0.0.*      LISTEN      -
udp     0      0    127.0.0.11:48716      0.0.0.0.*      LISTEN      -
[root@63c74c854084 ~]# sudo netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp     0      0    127.0.0.11:46623      0.0.0.0.*      LISTEN      -
udp     0      0    127.0.0.11:48716      0.0.0.0.*      LISTEN      -
[root@63c74c854084 ~]#
```

Uso de sudo netstat-tulpn en Garuda

```
Looted file /etc/hosts.equiv sequence  
gixxer155abs@gixxer155abs-Latitude-E6220:~/corute_2/PROVECTOZ$ docker exec -it contenedor-debian bash  
root@fb9890c0124c:/# netstat -tulpn  
Active Internet connections (only servers)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State      PID/Program name  
tcp        0      0 127.0.0.1:37789          0.0.0.0.*      LISTEN     -  
udp        0      0 127.0.0.1:41890          0.0.0.0.*      -  
root@fb9890c0124c:/#
```

Uso de sudo netstat-tulpn en Debian

Cuarta actividad: Usar “ip a” en las máquinas virtuales y “docker inspect”

<nombre-c> | grep IPAddress” contenedores y comparar resultados de lo

visualizado

```
gixxer155abs@gixxer155abs-Latitude-E620:~/Corte_2/PROYECTOS docker inspect Fedora_Central | grep IPAddress
  "SecondaryIPAddresses": null,
  "IPAddress": "172.17.0.4",
  "IPAddress": "172.17.10.4",
  "IPAddress": "192.168.100.128",
gixxer155abs@gixxer155abs-Latitude-E620:~/Corte_2/PROYECTOS docker inspect contenedor-debian | grep IPAddress
  "SecondaryIPAddresses": null,
  "IPAddress": "172.17.0.5",
  "IPAddress": "172.17.0.5",
  "IPAddress": "192.168.100.131",
gixxer155abs@gixxer155abs-Latitude-E620:~/Corte_2/PROYECTOS docker inspect garuda-like | grep IPAddress
  "SecondaryIPAddresses": null,
  "IPAddress": "172.17.0.2",
  "IPAddress": "172.17.0.2",
  "IPAddress": "192.168.100.130",
gixxer155abs@gixxer155abs-Latitude-E620:~/Corte_2/PROYECTOS docker inspect contenedor-alpine-netdata | grep IPAddress
  "SecondaryIPAddresses": null,
  "IPAddress": "172.17.0.3",
  "IPAddress": "172.17.0.3",
  "IPAddress": "192.168.100.129",
gixxer155abs@gixxer155abs-Latitude-E620:~/Corte_2/PROYECTOS
```

Contenedores y sus Direcciones IP:

Fedora_Central

IP en red bridge de Docker: 172.17.0.4

Otra IP: 192.168.100.128

contenedor-debian

IP en red bridge de Docker: 172.17.0.5

Otra IP: 192.168.100.131

garuda-like

IP en red bridge de Docker: 172.17.0.2

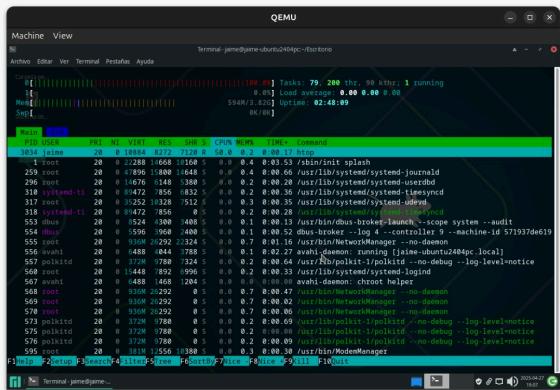
Otra IP: 192.168.100.130

contenedor-alpine-netdata

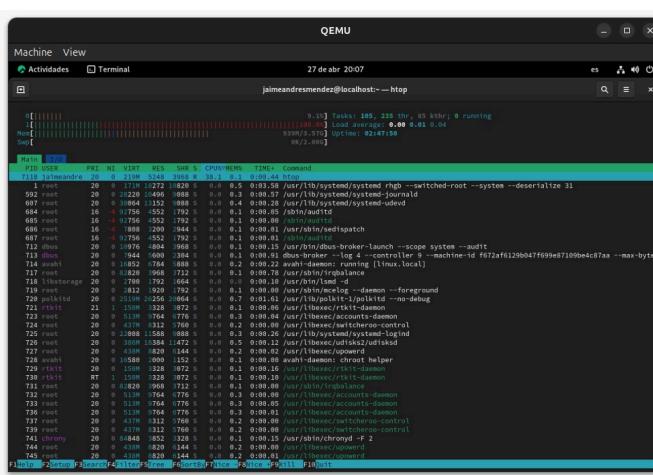
IP en red bridge de Docker: 172.17.0.3

Otra IP: 192.168.100.129

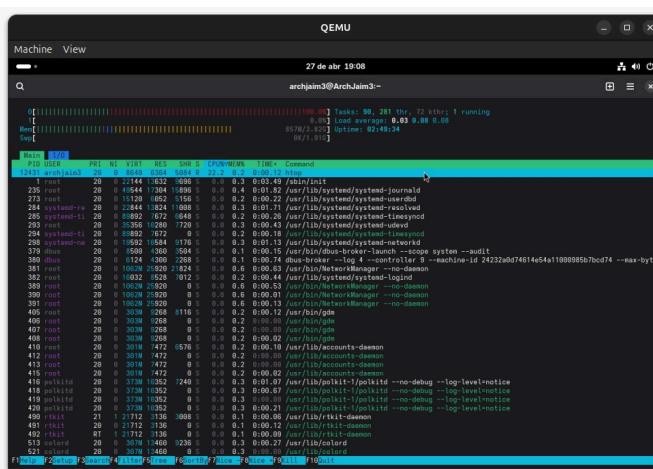
Quinta actividad: Ejecutar htop en máquinas virtuales y contenedores.



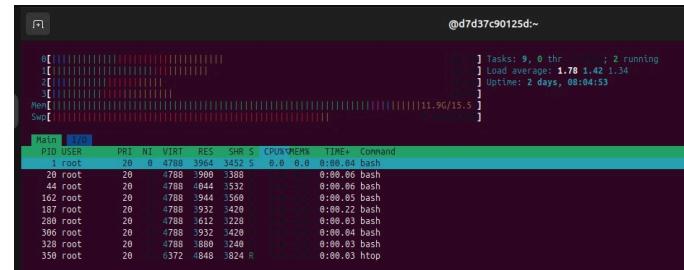
HTOP: en Manjaro



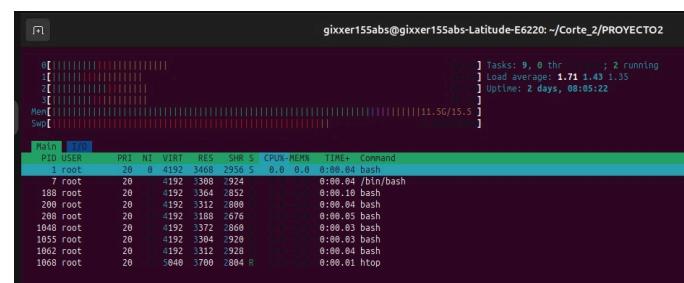
HTOP en RockyLinux



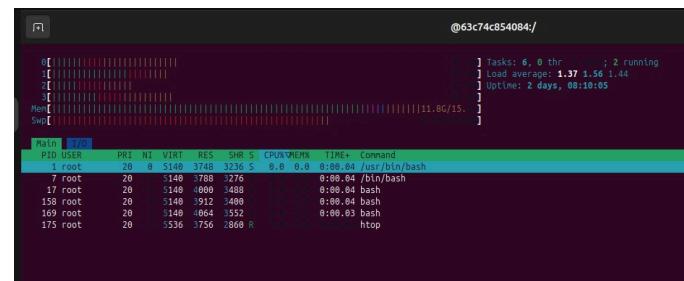
HTOP en Archlinux



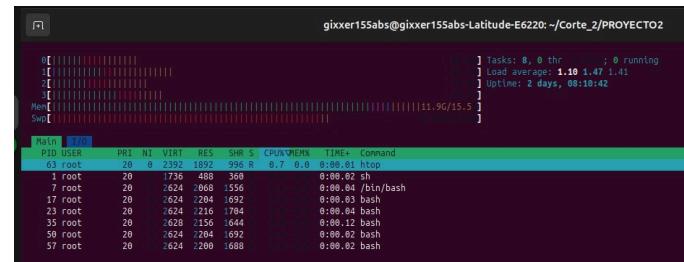
HTOP en Fedora



HTOP en DEBIAN



HTOP en GARUBA



HTOP en Alpine

Sexta actividad: Ejecutar tcpdump en máquinas virtuales, contenedores y contenedor central para analizar el tráfico generado.

Primeramente instalar tcpdump y luego se ejecuta en Alpine

tcpdump en Manjaro

Primeramente instalar tcpdump y luego se ejecuta en Garuda

tcpdump en RockyLinux

```
Unpacking tcpdump (4.99.3-1) ...
Setting up tcpdump (4.99.3-1) ...
root@bf9890c8124c:/# tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), snapshot length 262144 bytes
00:39:19.932873 IP gixer155abs-Latitude-E6220.52949 > 239.255.255.250.1900: UDP, length 173
00:39:28.29.934090 IP gixer155abs-Latitude-E6220.52949 > 239.255.255.250.1900: UDP, length 173
00:39:21.934492 IP gixer155abs-Latitude-E6220.52949 > 239.255.255.250.1900: UDP, length 173
00:39:22.935363 IP gixer155abs-Latitude-E6220.52949 > 239.255.255.250.1900: UDP, length 173
00:41:19.933883 IP gixer155abs-Latitude-E6220.54280 > 239.255.255.250.1900: UDP, length 173
00:41:20.935481 IP gixer155abs-Latitude-E6220.54280 > 239.255.255.250.1900: UDP, length 173
00:41:21.936122 IP gixer155abs-Latitude-E6220.54280 > 239.255.255.250.1900: UDP, length 173
00:41:22.937029 IP gixer155abs-Latitude-E6220.54280 > 239.255.255.250.1900: UDP, length 173
^C
8 packets captured
8 packets received by filter
0 packets dropped by kernel
root@bf9890c8124c:/#
```

Primeramente instalar tcpdump y luego se ejecuta en Debian

tcpdump en ArchLinux

```
0 packets dropped by kernel
[root@63c74c854084/]# docker exec -it Fedora_Central bash
bash: docker: command not found
[root@63c74c854084/]# tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
00:41:19.933927 IP _gateway.54280 > 239.255.255.250.ssdp: UDP, length 173
00:41:20.935601 IP _gateway.54280 > 239.255.255.250.ssdp: UDP, length 173
00:41:21.936189 IP _gateway.54280 > 239.255.255.250.ssdp: UDP, length 173
00:41:22.937077 IP _gateway.54280 > 239.255.255.250.ssdp: UDP, length 173
^C
4 packets captured
4 packets received by filter
0 packets dropped by kernel
[root@63c74c854084/]#
```

Primeramente instalar tcpdump y luego se ejecuta en Fedora

Actividad Número 5

Desarrollar Arquitectura Propuesta-Análisis con Zabbix, Grafana y Prometheus

```
[n] gixxer155abs@gixxer155abs-Latitude-E6220: ~ $ sudo apt-get install -y apt-transport-https software-properties-common wget  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias... Hecho  
Leyendo la información de estado... Hecho  
apt-transport-https ya está en su versión más reciente (2.7.14+build2).  
software-properties-common ya está en su versión más reciente (0.99.49.2).  
wget ya está en su versión más reciente (1.21~1ubuntu4.1).  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
libssl2.0-pref greencard squashfs-tools tree uidmap wl-clipboard xclip  
Utilice "sudo apt autoremove" para eliminarlos.  
actualizados. 0 nuevo paquete se instalarán, 0 paquetes se actualizan.  
gixxer155abs@gixxer155abs-Latitude-E6220: ~ $ wget q -O https://packages.grafana.com/gpg.key | sudo apt-key add -  
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).  
OK  
gixxer155abs@gixxer155abs-Latitude-E6220: ~ $ sudo add-apt-repository deb https://packages.grafana.com/oss/deb stable main"  
Repositorio: deb https://packages.grafana.com/oss/deb stable main  
Añadiendo repositorio:  
  Archive for codename: stable components: main  
  Más información: https://packages.grafana.com/oss/deb  
Añadiendo repositorio:  
  Oprimir [INTRO] para continuar o Ctrl+P para cancelar.  
  Añadiendo repositorio: https://archive_url=https://packages.grafana.com/oss/deb/noble.list  
  Añadiendo repositorio: https://archive_url=https://packages.grafana.com/oss/deb/noble.list  
  Añadiendo repositorio: https://archive_url=https://packages.grafana.com/oss/deb/noble.list  
  Añadiendo repositorio: https://archive_url=https://packages.grafana.com/oss/deb/noble.list  
  Añadiendo repositorio: https://archive_url=https://packages.grafana.com/oss/deb/noble.list  
Des1: https://packages.grafana.com/oss/deb stable InRelease [7,661 B]  
Des2: https://archive/ubuntu.com/ubuntu noble InRelease  
Des3: https://archive.ubuntu.com/ubuntu noble InRelease  
Des4: https://download.docker.com/linux/ubuntu noble InRelease  
Des5: https://archive.ubuntu.com/ubuntu noble-security InRelease [126 kB]  
Des6: https://archive.ubuntu.com/ubuntu/dists/bionic-updates InRelease [126 kB]  
Des7: https://repo.zabbix.com/zabbix/6.4/ubuntu noble InRelease  
Des8: https://packages.grafana.com/oss/deb/main amd64 Packages [386 kB]
```

Este registro muestra los pasos realizados para configurar el repositorio de Grafana

Proceso Realizado:

Instalación de Paquetes Requeridos:

bash

```
sudo apt-get install -y apt-transport-https  
software-properties-common wget
```

Todos los paquetes ya estaban instalados en sus versiones más recientes

Se sugiere ejecutar sudo apt autoremove para limpiar paquetes innecesarios

Adición de Clave GPG de Grafana:

hash

```
bash
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
```

Se recibió una advertencia sobre que apt-key está obsoleto

La clave se añadió correctamente ("OK")

Configuración del Repositorio de Grafana:

```
bash                               add-apt-repository      "deb
sudo                               https://packages.grafana.com/oss/deb stable main"
https://packages.grafana.com/oss/deb stable main"
Se crearon dos entradas en /etc/apt/sources.list.d/:
```

Una para binarios (habilitada)

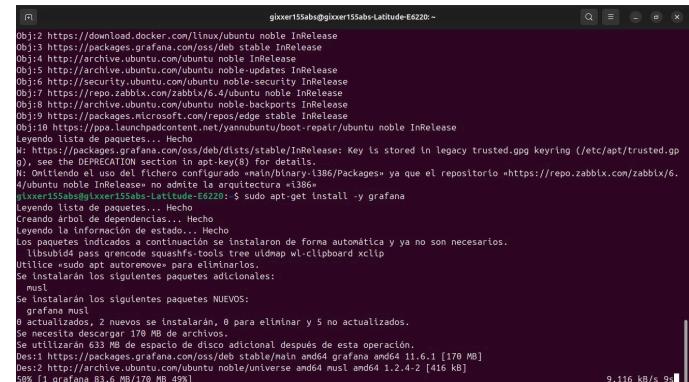
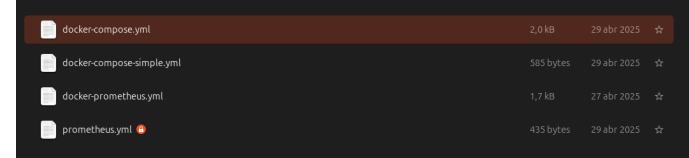
Otra para código fuente (deshabilitada)

Actualización de Paquetes (implícita):

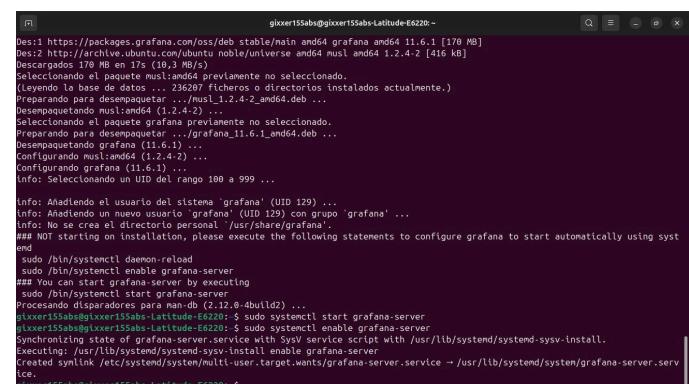
El sistema comenzó a actualizar la lista de paquetes, mostrando:

Repositorio de Grafana (386 kB de metadatos descargados)

Repositorios adicionales configurados (Google Chrome, Docker, Zabbix)



Instalamos Grafana



Damos inicio al servidor de Grafana

Aquí determinamos los puertos y objetivo del Node Exporter, en el archivo prometheus.yml

Configuramos Prometheus

determinamos usuario en prometheus.service

```
glixer155abs@glixer155abs:~$ sudo chmod 755 prometheus /var/lib/prometheus
glixer155abs@glixer155abs:~$ sudo nano /etc/prometheus/prometheus.yml
glixer155abs@glixer155abs:~$ sudo nano /etc/prometheus/prometheus.yml
glixer155abs@glixer155abs:~$ sudo nano /etc/systemd/system/prometheus.service
glixer155abs@glixer155abs:~$ sudo systemctl daemon-reload
glixer155abs@glixer155abs:~$ sudo systemctl start prometheus
glixer155abs@glixer155abs:~$ sudo systemctl enable prometheus
Created symlink /etc/systemd/system/multi-user.target.wants/prometheus.service → /etc/systemd/system/prometheus.service.
glixer155abs@glixer155abs:~$
```

Movemos los archivos al directorio correspondiente

```
[root@lxvm1 ~]# curl -v http://192.168.1.11:9100/metrics | grep node_exporter
node_exporter-1.3.1.linux-amd64/
node_exporter-1.3.1.linux-amd64/LICENSE
node_exporter-1.3.1.linux-amd64/NOTICE
node_exporter-1.3.1.linux-amd64/node_exporter
gixler155abs@lxvm155abs-Latitude-E6202: $ sudo cp node_exporter-1.3.1.linux-amd64/node_exporter /usr/local/bin/
gixler155abs@lxvm155abs-Latitude-E6202: $ sudo nano /etc/systemd/system/node_exporter.service
gixler155abs@lxvm155abs-Latitude-E6202: $ sudo systemctl daemon-reload
gixler155abs@lxvm155abs-Latitude-E6202: $ sudo systemctl start node_exporter
gixler155abs@lxvm155abs-Latitude-E6202: $ sudo systemctl enable node_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/node_exporter.service → /etc/systemd/system/node_exporter.service.
gixler155abs@lxvm155abs-Latitude-E6202: $
```

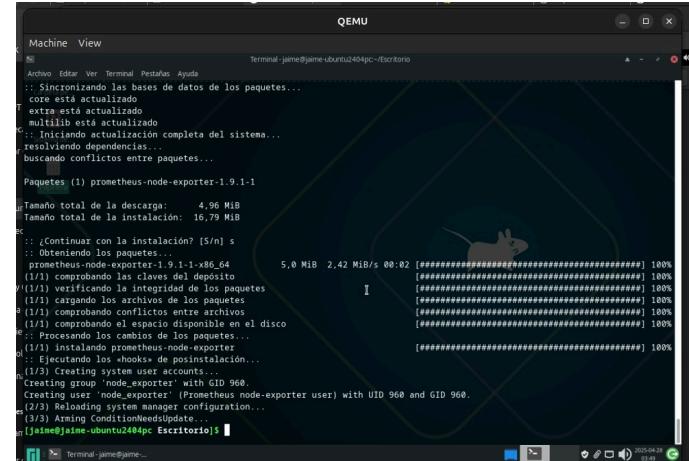
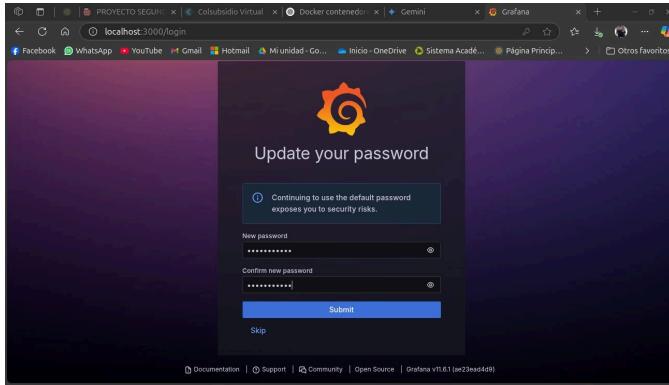
Más Configuraciones de Prometheus

Descargamos y habilitamos el servicio Node_exporter

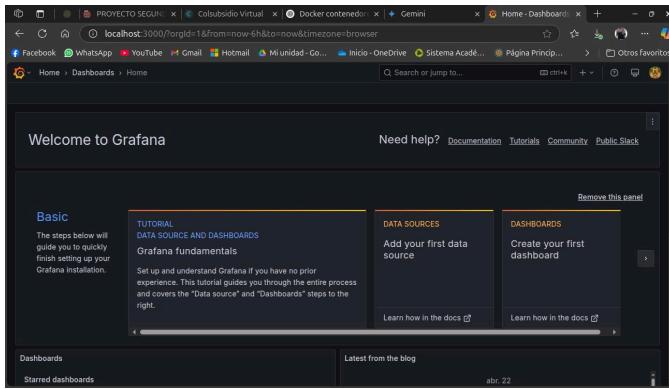
```
gioxer15sabs@gioxer15sabs-Latitude-E6220 ~
GNU nano 7.2
global:
  scrape_interval: 15s # Establece el intervalo de recolección global a 15 segundos.

scrape_configs:
  # Job para recopilar métricas del propio Prometheus.
  - job_name: 'prometheus'
    static_configs:
      - targets: [ 'localhost:9090' ] # Define el objetivo (target) como la instancia local de Prometheus en el puerto 9090.

  # Job para recopilar métricas de un Node Exporter (asumiendo que está instalado y corriendo).
  - job_name: 'node_exporter'
    static_configs:
      - targets: [ 'localhost:9100' ] # Define el objetivo como el Node Exporter local en el puerto 9100.
```



Configuramos una nueva contraseña



Y ya estamos adentro de Grafana

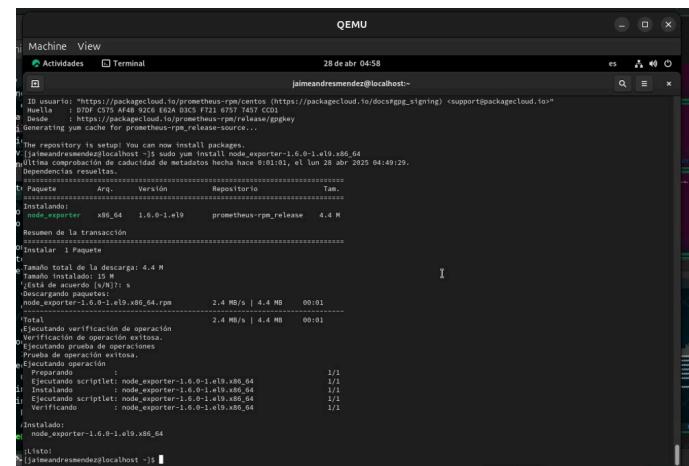
```
gixer155abs@giixer155abs-Latitude-E6220:~/Corte_2/PROYECTO2/monitoring$ nano docker-compose.yml
gixer155abs@giixer155abs-Latitude-E6220:~/Corte_2/PROYECTO2/monitoring$ docker compose up -d
No se ha encontrado la orden `docker compose`, pero se puede instalar con:
  sudo snap install docker
  o
  curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
  Consulte https://docs.docker.com/compose/install/ para ver más versiones.
gixer155abs@giixer155abs-Latitude-E6220:~/Corte_2/PROYECTO2/monitoring$ docker compose up -d
WARN [0800] /home/gixer155abs/Corte_2/PROYECTO2/monitoring/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored. Please remove it to avoid potential confusion
  (i) Running 8/7
    zabbix-server [  ] 30.19MB / 31.23MB Pulling
    zabbix-web [  ] Pulling
    zabbix-db [  ] Pulling
    prometheus [  ] Pulling
    grafana [  ] Pulling
5.2s
5.2s
5.2s
5.2s
5.2s
```

Co Docker Compose y debidamente configurado, construimos las imágenes donde se aloja configurado, Grafana, Zabbix y Prometheus

Aquí ya están iniciadas.



Y el Zabbix explorer y lo activamos



Rocky Linux

```
  Activities Terminal 28 abr 04:59
-jaimeandresmendez@local-host:-
Dependencias resueltas
Paquetes:
Instalando
  libcurl4-openssl4.x86_64 1:6.39-1.el9      epel          293 k
Instalando dependencias:
  libcurl4.x86_64           1:6.39-1.el9      epel          739 k
    -cabecera-selinux         1:6.39-1.el9      epel          26 k
Resumen de la transacción
Instalar 3 Paquetes

Tamaño total de la descarga: 1.0 M
Tamaño total en el sistema: 293 k
[Está de acuerdo (y/N)?] y
(3/3): zabbix-selinux-6.8-39-1.el9.noarch.rpm
(3/3): zabbix-agent-1.6.39-1.el9.x86_64.rpm
(3/3): zabbix-cron-1.6.39-1.el9.x86_64.rpm
Total
660 kB/s | 1.0 MB 00:01
Total
  Verificando verificación de operación
  Verificación de operación exitosa
  Instalando paquetes
  Prueba de operación exitosa
  Ejecutando operación
  Ejecutando scriptlet zabbix-setenforce-1.6.39-2.el9.noarch
  Ejecutando scriptlet zabbix-setenforce-1.6.39-1.el9.noarch
  Ejecutando scriptlet zabbix-setenforce-1.6.39-1.el9.noarch
  Ejecutando scriptlet zabbix-setenforce-1.6.39-1.el9.noarch
  Usando modprobe para establecer el módulo zabbix module at lower priority 100 with module at priority 200.
  Instalando
    zabbix-agent-1.6.39-1.el9.x86_64
  Ejecutando scriptlet zabbix-agent-1.6.39-1.el9.x86_64
  Ejecutando scriptlet zabbix-cron-1.6.39-1.el9.x86_64
  Ejecutando scriptlet zabbix-setenforce-1.6.39-1.el9.noarch
  Ejecutando scriptlet zabbix-setenforce-1.6.39-1.el9.noarch
  Verificado
    zabbix-agent-1.6.39-1.el9.x86_64
  Verificado
    zabbix-cron-1.6.39-1.el9.x86_64
  Resumen
zabbix-1.6.39-1.el9.x86_64          zabbix-agent-1.6.39-1.el9.x86_64          zabbix-setenforce-1.6.39-1.el9.noarch
[lista]
[jaimeandresmendez@local-host: ~]$ sudo dnf install projectheus-node-reporter*
[lista]
[jaimeandresmendez@local-host: ~]$ curl -s https://packagedata.net/install-repositories/projectheus-rpm/release/script.rpm.sh | sudo bash
```

```

# Multiple Zabbix servers/clusters and Zabbix proxies can be specified, separated by semicolon.
# Note that proxy must be specified, not port, as it is not specified from each Zabbix server/cluster.
# If Zabbix proxy is specified than Zabbix server/cluster for that proxy should not be specified.
# Multiple comma-delimited addresses can be provided to use several independent Zabbix servers in parallel. Separated by semicolon.
# If port is not specified, default port is used.
# IPv6 addresses must be enclosed in square brackets if port for that host is specified.
# If port is not specified, square brackets for IPv6 addresses are optional.
# If this parameter is not specified, active checks are disabled.
# Example for Zabbix proxy:
#   ServerActive=127.0.0.1:10851
# Example for multiple servers:
#   ServerActive=[127.0.0.1:20051;zabbix.domain,[::1]:30051,::1,[12fc::1]]
# Example for high availability:
#   ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
# Example for high availability with two clusters and one server:
#   ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster2.node1;zabbix.cluster2.n

# Mandatory: no
# Default:
# ServerActive=

ServerActive=192.168.100.1

### Option: Hostname
# List of comma delimited unique, case sensitive hostnames.
# Required for active checks and must match hostnames as configured on the server.
# Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=Debian-Contenededor

### Option: HostnameItem
# Item used for generating Hostname if it is undefined. Ignored if Hostname is defined.
# Does not support UserParameters or aliases.
#




```

Zabbix en Rocky Linux

Aqui estamos instalando Prometheus en Debian

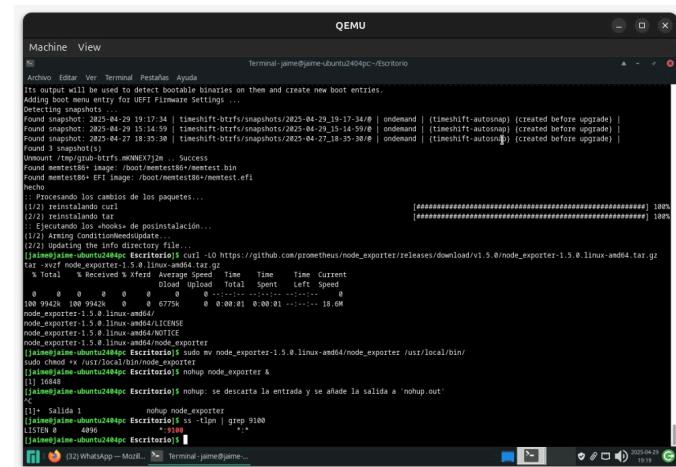
```
root@gbf9890c8124c:/# apt install -y zabbix-agent
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libmodbus5 libusb-1.0-0 pciutils sensible-utils ucf usbutils
Suggested packages:
  bzip2 wget l curl lynx logrotate
The following NEW packages will be installed:
  libmodbus5 libusb-1.0-0 pciutils sensible-utils ucf usbutils zabbix-agent
  0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 1021 kB of archives.
After this operation, 2482 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 libmodbus5 amd64 3.1.6-2.1 [31.3 kB]
Get:2 http://deb.debian.org/debian bookworm/main amd64 sensible-utils all 0.0.17+nmu1 [19.0 kB]
Get:3 http://deb.debian.org/debian bookworm/main amd64 ucf all 3.0.043+nmu1+deb12u1 [50.3 kB]
Get:4 http://deb.debian.org/debian bookworm/main amd64 pciutils amd64 1:3.9.0-4 [104 kB]
Get:5 http://deb.debian.org/debian bookworm/main amd64 zabbix-agent amd64 1:6.0.14+dfsg-1+b1 [67 kB]
Get:6 http://deb.debian.org/debian bookworm/main amd64 libusb-1.0-0 amd64 2:1.0.26-1 [62.6 kB]
Get:7 http://deb.debian.org/debian bookworm/main amd64 usbutils amd64 1:014-1+deb12u1 [73.5 kB]
Fetched 1021 kB in 0s (3665 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package libmodbus5:amd64.
(Reading database ... 15892 files and directories currently installed.)
Preparing to unpack .../0-libmodbus5_3.1.6-2.1_amd64.deb ...
Unpacking libmodbus5:amd64 (3.1.6-2.1) ...
Selecting previously unselected package sensible-utils.
Preparing to unpack .../1-sensible-utils_0.0.17+nmu1_all.deb ...
Unpacking sensible-utils (0.0.17+nmu1) ...
Selecting previously unselected package ucf.
```

Ajustamos el archivo Zabbix

```
[root@node01 ~]# apt update & apt install -y grafana
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://deb.debian.org/debian-security bookworm-security InRelease
Get:3 https://packages.grafana.com/oss/deb stable InRelease [768 B]
Get:4 https://packages.grafana.com/oss/deb stable/main arm64 Packages [4492 kB]
Get:5 https://packages.grafana.com/oss/deb stable/main arm64 Packages [388 kB]
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
Building dependency tree... Done
Reading state information... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  nodejs
The following NEW packages will be installed:
  grafana 11.6.1-1+deb12u1_arm64
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 4492 kB of archives.
After this operation, 631 kB of additional disk space will be used.
Get:1 https://deb.debian.org/debian/bookworm/main arm64 nodejs 12.3.1 (468 kB)
Get:2 https://deb.debian.org/debian-security/bookworm-security/main arm64 grafana 11.6.1-1 [376 kB]
Get:3 https://packages.grafana.com/oss/deb/stable/main arm64 grafana 11.6.1-1 [376 kB]
Patched 170 files in 122.142 kB/s
dpkg: warning: maintainer script /var/lib/dpkg/info/grafana.postinst failed to start up
  /usr/bin/python3 /var/lib/dpkg/info/grafana.postinst
dpkg: error processing package grafana (--install):
  subprocess /usr/bin/python3 /var/lib/dpkg/info/grafana.postinst returned an error exit status 1
(Reading database... 16946 files and directories currently installed.)
Preparing to unpack .../grafana_11.6.1-1_arm64.deb...
Unpacking grafana (11.6.1)...
Selecting previously unselected package grafana.
Preparing to unpack .../grafana_11.6.1-1_arm64.deb...
Unpacking grafana (11.6.1)...
Setting up grafana (11.6.1)...
Setting up grafana (11.6.1)...
adding system user 'grafana' (UID 1000)...
adding user 'grafana' to group 'grafana'...
Not creating home directory '/usr/share/grafana'.
## NOT starting on installation, please execute the following statements to configure grafana to start automatically using system
## services
  sudo systemctl enable grafana-server
```

y luego Zabbix

Aqui estamos instalando el agente Zabbix en Debiam

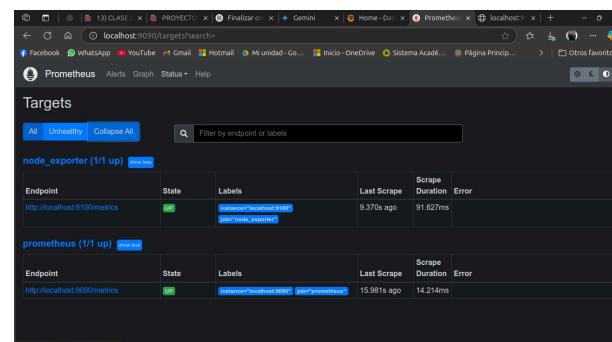


Node Exporter en Manjaro

```
gitxer15Sabs@giixer15Sabs-Latitude-E6220:/Corte_2/PROYECTO2/monitoring
```

```
Otro modo: 7-3  
-----  
$ ./monit -c /etc/monitrc  
monit: Interval: 15s # Frecuencia de recolección  
  
scraper_configs:  
# Para monitorear el servidor de pruebas  
- job_name: 'scraperweb'  
  static_configs:  
    - targets: [ 'localhost:8080' ]  
  
  # 000 para monitorear las Máquinas Virtuales (VMs)  
- job_name: 'scrapervm'  
  static_configs:  
    - targets:  
        - '192.168.100.1:8080' # IP Apache VM (Virt/Fisica/Ahosta si es arreglado)  
        - '192.168.100.1:8081' # IP MySQL VM (Virt/Fisica/Ahosta si es arreglado)  
        - '192.168.100.1:8082' # IP Arch VM (Virt/Fisica/Ahosta si es arreglado)  
  
# 000 para monitorear los Contenedores Docker  
- job_name: 'scrapercontenedores'  
  static_configs:  
    - targets:  
        - '# IP de los contenedores en red-monitores + puerto Node Exporter (9080)  
          - 192.168.100.179:9080 # IP del contenedor_docker-nodeexporter  
        - 192.168.100.179:9081 # IP del contenedor_docker-metrics  
        - 192.168.100.179:9082 # IP del contenedor_docker
```

Modificando el archivo prometheus.yml del contenedor



Node exporter en ArchLinux

Aqui estamos en el puerto en el que se asigno Prometheus desde el Host y no vemos reflejado las maquinas ni contenedores

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
7a51e13bba3a	grafana/grafana	"/run.sh"	39 hours ago	Exited (255)	22 hours ago
ac638907eb5	prom/prometheus	"./bin/prometheus -c -"	39 hours ago	Exited (255)	22 hours ago
Ffd3eb0fcf13	e7b144b5ddf4	"echo 'Hola Docker H_'"	3 days ago	Exited (0)	3 days ago
d7d37c90125d	central-analytics	"bash"	3 days ago	Exited (255)	22 hours ago
b7f898c8124c	debian	"bash"	3 days ago	Exited (255)	22 hours ago
:8880->88/tcp	contenedor-debian	"netdata/netdata	9 days ago	Exited (128)	9 days ago
63cf4cb54084	archlinux	"/usr/bin/bash"	8 days ago	Exited (255)	22 hours ago
d4719846deb	alpine	"sh"	9 days ago	Exited (255)	22 hours ago
[::]:19999->19999/tcp	contenedor-alpine-netdata	"netdata/netdata	9 days ago	0.0.0.0:8880->80/tcp, [::]:	
4cd19e1aae22d	netdata	"netdata"	9 days ago	Exited (0)	9 days ago
ed0cfb8590c	hello-world	"hello"	9 days ago	Exited (0)	9 days ago
e950f19e8af9	hello-world_mclaren	"hello"	9 days ago	Exited (0)	9 days ago
	nostalgic_lewin				

Aquí observamos los puertos abiertos de los contendores tratando de entender porque no funciona

```
# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS
7a51e13bba3a        grafana/grafana      "/run.sh"          39 hours ago       Exited (255)        22 hours ago
ac638907eb5         prom/prometheus   "./bin/prometheus -c -"  39 hours ago       Exited (255)        22 hours ago
Ffd3eb0fcf13        e7b144b5ddf4      "echo 'Hola Docker H_'"  3 days ago        Exited (0)          3 days ago
d7d37c90125d        central-analytics   "bash"            3 days ago        Exited (255)        22 hours ago
b7f898c8124c        debian             "/bin/bash"        3 days ago        Exited (255)        22 hours ago
:8880->88/tcp       contenedor-debian  "netdata/netdata"  9 days ago        Exited (128)        9 days ago
63cf4cb54084        archlinux          "/usr/bin/bash"    8 days ago        Exited (255)        22 hours ago
d4719846deb        alpine             "sh"              9 days ago        Exited (255)        22 hours ago
[::]:19999->19999/tcp  contenedor-alpine-netdata  "netdata/netdata"  9 days ago        0.0.0.0:8880->80/tcp, [::]:
```

Puerto desde el host de Zabbix y no sabes si es normal esto...

Y A PARTIR DE AQUI TODO EMPEZO A FALLAR,
DAMOS CONCLUIDO EL INTENTO DEL PROYECTO 😞

IV.CONCLUSIONES

El desarrollo de este proyecto permitió la implementación de un entorno de pruebas híbrido orientado al monitoreo y análisis de sistemas operativos utilizando contendores Docker, redes virtuales personalizadas (macvlan), y herramientas de diagnóstico como glances, btop, iotop, wireshark, ncdt y rsync. Se logró la integración de contendores Linux livianos (Arch, Alpine, Debian) junto con máquinas virtuales, conectadas mediante una red interna (red-monitoreo) con capacidades de comunicación y aislamiento definidas.

Durante la ejecución se identificaron y resolvieron desafíos comunes del ecosistema Docker, como conflictos de red, problemas de caché de red y errores de conexión entre contendores. Estos obstáculos permitieron profundizar en el manejo avanzado de Docker, incluyendo creación de redes macvlan, asignación de IPs fijas, y gestión de contendores personalizados.

El entorno resultante facilita pruebas sistemáticas, monitoreo de recursos, y análisis de tráfico de red entre sistemas, permitiendo simular escenarios reales de producción o laboratorios de ciberseguridad. Además, la automatización mediante docker-compose y la modularidad de los servicios garantizan escalabilidad y reutilización del proyecto en entornos similares.

En resumen, este proyecto no solo consolida conocimientos en virtualización ligera, redes y monitoreo, sino que establece una base robusta para el despliegue de laboratorios técnicos avanzados con bajo consumo de recursos.

RECONOCIMIENTOS

No hay reconocimiento...

REFERENCIAS

- [9] Docker Inc. (2024). Docker Documentation. Recuperado de: <https://docs.docker.com/>
- [10]
- [11] Arch Linux. (2024). ArchWiki - Docker. Recuperado de: <https://wiki.archlinux.org/title/Docker>
- [12]
- [13] Alpine Linux. (2024). Official Alpine Linux Docker Image. Docker Hub. Recuperado de: https://hub.docker.com/_alpine
- [14]
- [15] Debian Project. (2024). Debian Docker Image. Docker Hub. Recuperado de: https://hub.docker.com/_debian
- [16]
- [17] Red Hat. (2023). An Introduction to macvlan Network Drivers in Docker. Recuperado de: <https://www.redhat.com/sysadmin/macvlan-docker>
- [18]
- [19] Prometheus Authors. (2024). Prometheus Documentation. Recuperado de: <https://prometheus.io/docs/introduction/overview/>
- [20]
- [21] Grafana Labs. (2024). Grafana Documentation. Recuperado de: <https://grafana.com/docs/>
- [22]
- [23] Zabbix LLC. (2024). Zabbix Docker Install Guide. Recuperado de: <https://www.zabbix.com/documentation/current/en/manual/installation/containers>
- [24]
- [25] Mitchell, B. (2022). Understanding Computer Networking for System Monitoring. TechBooks Publishing.
- [26]
- [27] Gite, V. (2023). Linux Performance and Monitoring Tools. nixCraft. Recuperado de: <https://www.cyberciti.biz/tips/linux-performance-monitoring-tools.htm>