Implementación de entorno de infraestructura virtual con PNetLab sobre Proxmox VE 8.x



Requisitos para la implementación

Proxmox para entornos de producción, es necesario altos recursos en hardware como RAM, CPU, espacio en discos y de calidad, esto lo obtendremos en equipos desarrollados por varios fabricantes de hardware específicos, sean estos, HP, Dell, Super Micro, IBM, entre otros, pero esta la otra opción no recomendada por los componentes electrónicos que conlleva un equipo como es el que llamos “Equipo Clon”. Lo mas importante es, dimensionar según sus necesidades.

Para entornos para evaluar:

- CPU de 64bits (Intel-64 o AMD-64).
- Soporte de virtualización completa Intel VT/ADM-V.
- RAM de 1 Gb.
- Disco Duro.
- Una tarjeta de red (NIC).

Entorno recomendado:

- CPU de 64bits (Intel-64 o AMD-64) que cuente con múltiples core.
- Un mínimo de 2Gb de RAM, con soporte para Ceph y ZFS, es necesario adicionar más RAM por cada TB de espacio.
- Discos Duros de gran velocidad, para esto los mejores resultados es utilizar SSDs.
- Almacenamiento para el O.S., se implementa una controladora de hardware RAID con cache de escritura protegida por batería, no para sistemas distribuido Ceph y con ZFS (esto para almacenamiento de VM).
- Múltiples interfaces de red, de alta velocidad de transferencia de datos para entornos donde entra el juego de segmentación de red y demás(vlan).
- Para temas de PCI(e) passthrough, el CPU necesita que soporte la tecnología VT-d/AMD-d.

Para más información:

https://pve.proxmox.com/pve-docs/pve-admin-guide.html#_system_requirements
https://www.proxmox.com/en/products/proxmox-virtual-environment/requirements
https://pve.proxmox.com/wiki/System_Requirements

Para PNetLab según el fabricante:

- CPU con soporte de para virtualización Intel VT-x/EPT y de múltiple core.
- Memoria de 8Gb a más.
- Espacio en disco de 40Gb a más.
- Interfaz de red físico o Virtual NIC Bridged.

Para más información:

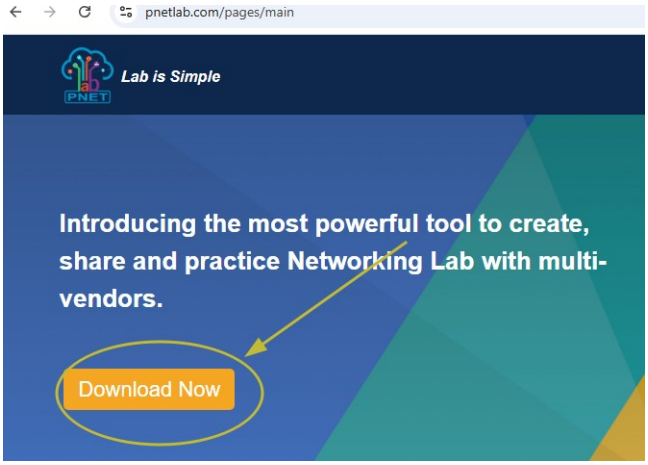
https://pnetlab.com/pages/documentation?slug=hardware-requirements

Descarga de PNetLab

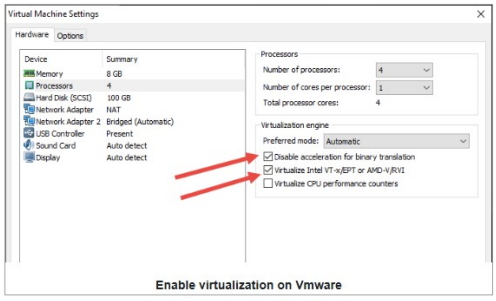
Para la implementación de un entorno virtual que ayudará en las prácticas y preparación para rendir a las certificaciones tales como: Cisco, Huawei, Mikrotik, Fortinet, entre otras. Contamos con un entorno que nos llega a la mano llamado “PnetLab”, más información ir a la siguiente url https://pnetlab.com/pages/main.

Este presente documento pretende ser una guía para ser implementado en un entorno de producción haciendo uso de un HyperVisor llamado “Proxmox VE” en su versión 8.3.0, para más información ir al url https://www.proxmox.com/en/.

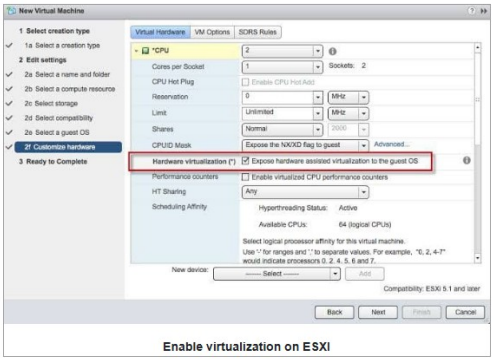
Empezamos con descargar la versión de PnetLab desde su sitio web, en este caso es la versión 4.2.10:



Descargamos en uno de los enlaces que nos ofrece:



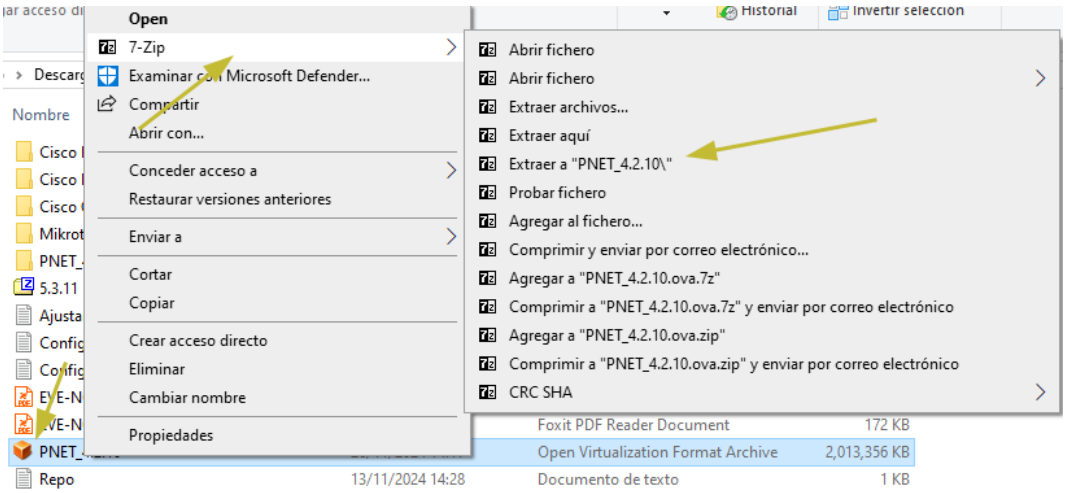
Enable virtualization on Vmware



Enable virtualization on ESXI

Link Download	MD5 Checksum	Size
Link Download PNET 4.2.10 Google Drive		2G
Link Download PNET 4.2.10 Google Drive (Backup 1)		2G
Link Download PNET 4.2.10 Mega (Backup 2)		2G

Una vez descargado el fichero en su formato OVA, lo descomprimimos:



Verificamos que tenemos 3 ficheros, estos ficheros están en un carpeta que se llama "***PNET_4.2.10***", por tanto, copiaremos esta carpeta hacia nuestro entorno de virtualización "***Proxmox VE***":

Listamos el directorio donde esta ubicado el directorio a copiar:

```
 Directorio de C:\Users\PC\Downloads\Para PNetLab

11/12/2024  14:09    <DIR>          .
11/12/2024  14:09    <DIR>          ..
14/11/2024  12:46       132,142,153 5.3.11.zip
11/12/2024  17:11             1,794 Ajustar en proxm
01/11/2024  14:22    <DIR>          Cisco IMG
01/11/2024  14:22    <DIR>          Cisco IOU
01/11/2024  14:22    <DIR>          Cisco QEMU
11/12/2024  14:09             1,163 Configuracion de
01/11/2024  14:18               822 Configuracion Mi
01/11/2024  14:03           155,789 EVE-NG and GNS3
01/11/2024  14:03           175,445 EVE-NG Qemu Imag
13/11/2024  12:09    <DIR>          Mikrotik IMG
06/12/2024  13:35    <DIR>          PNET_4.2.10
26/11/2024  14:17     2,061,676,032 PNET_4.2.10.ova
13/11/2024  14:28               479 Repo.txt
               8 archivos  2,194,153,677 bytes
               7 dirs  388,122,173,440 bytes libres

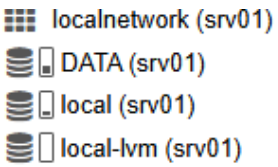
C:\Users\PC\Downloads\Para PNetLab>
```

Importación de disco de PNetLab a Proxmox VE 8.x

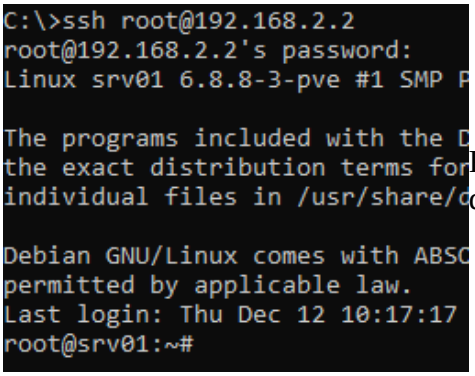
Copiamos el directorio hacia el servidor "***Proxmox VE***":

```
C:\Users\PC\Downloads\Para PNetLab>scp -r PNET_4.2.10 root@192.168.2.2:/opt
root@192.168.2.2's password:
PNET_4.2.10-disk1.vmdk                                      5%  108MB 107.4MB/s   00:17 ETA
```

Ingresamos al HyperVisor "***Proxmox VE***" vía WEB para ver los diferentes almacenamientos disponibles para exportar nuestro entorno de infraestructura de red "***PnetLab***":

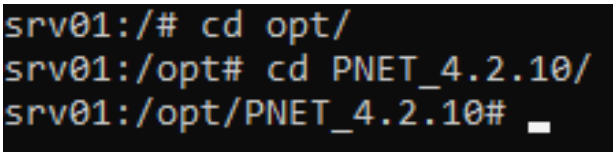


Podremos observar los diferentes almacenamientos y en mi caso seleccionaré "***DATA***", usted puede elegir en que almacenaimento importará "***PnetLab***". Ingresamos vía SSH al Server y nos ubicamos en el directorio donde copiamos el directorio que tiene los ficheros de "***PnetLab***":



Recuerde que es posible que su servidor tenga una dirección IP diferente al que se muestra.

En la copia desde nuestro equipo desktop hacia el server, el directorio destino fue "***/opt***", ingresamos hasta estar dentro del directorio "***PNET_4.2.10***":



Estando dentro del directorio "***PNET_4.2.10***", importamos el fichero con formato OFV:

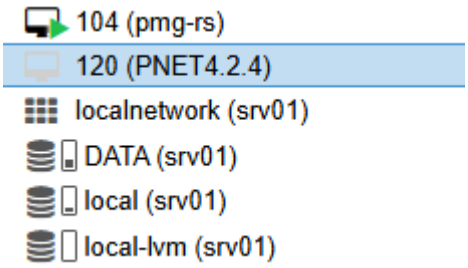
Sintaxis: *qm importovf VM_ID DISK.ovf STORAGE*

qm importovf 120 PNET_4.2.10.ovf DATA

En el proceso de importación no debe finalizar con errores:

```
root@srv01:/opt/PNET_4.2.10# qm importovf 120 PNET_4.2.10.ovf DATA
Formatting '/mnt/pve/DATA/images/120/vm-120-disk-0.raw', fmt=raw size=107374182400 preallocation=off
transferred 0.0 B of 100.0 GiB (0.00%)
transferred 1.0 GiB of 100.0 GiB (1.00%)
transferred 2.0 GiB of 100.0 GiB (2.00%)
transferred 3.0 GiB of 100.0 GiB (3.00%)
transferred 4.0 GiB of 100.0 GiB (4.00%)
transferred 5.0 GiB of 100.0 GiB (5.00%)
transferred 6.0 GiB of 100.0 GiB (6.00%)
transferred 7.0 GiB of 100.0 GiB (7.01%)
transferred 8.0 GiB of 100.0 GiB (8.01%)
transferred 9.0 GiB of 100.0 GiB (9.01%)
transferred 10.0 GiB of 100.0 GiB (10.01%)
transferred 11.0 GiB of 100.0 GiB (11.01%)
transferred 12.0 GiB of 100.0 GiB (12.01%)
```

Una vez finalizado la importación, ingresamos una vez más a "***Proxmox VE***" vía WEB, para verificar que ya fue creado la VM para "***PNetLab***":

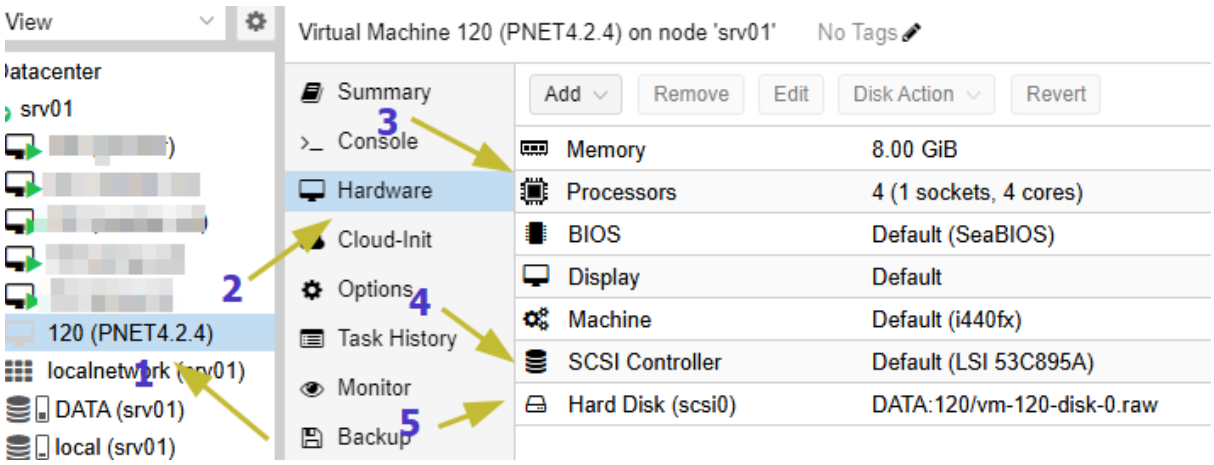


Al verificar que fue creado, eliminamos el directorio que copiamos anteriormente en "***/opt***":

```
:/opt/PNET_4.2.10# cd ..
:/opt# rm -rf PNET_4.2.10/
```

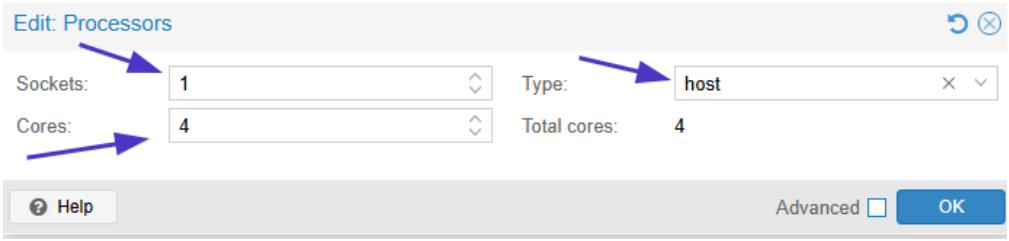
Ajuste de Hardware Virtual de PNetLab

Ahora nos toca proceder a ajustar la VM de "***PnetLab***", esto en "***Proxmox VE***" vía WEB:

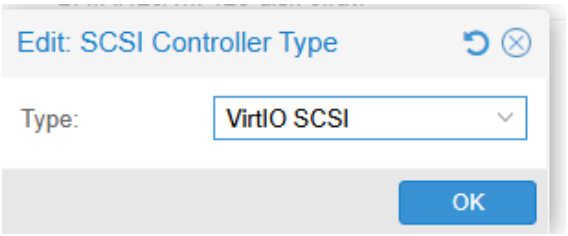


Los puntos a ajustar son:

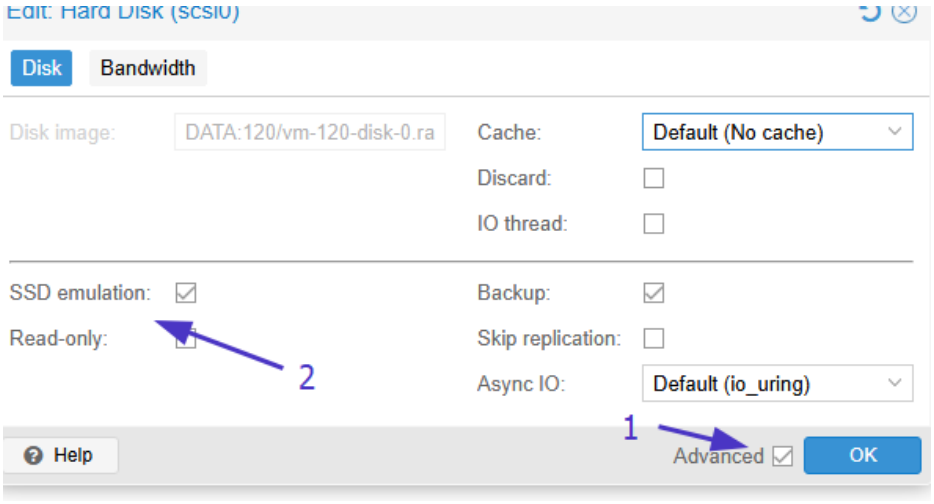
3 – Procesador, ahí seleccionamos el tipo de CPU como "***host***".



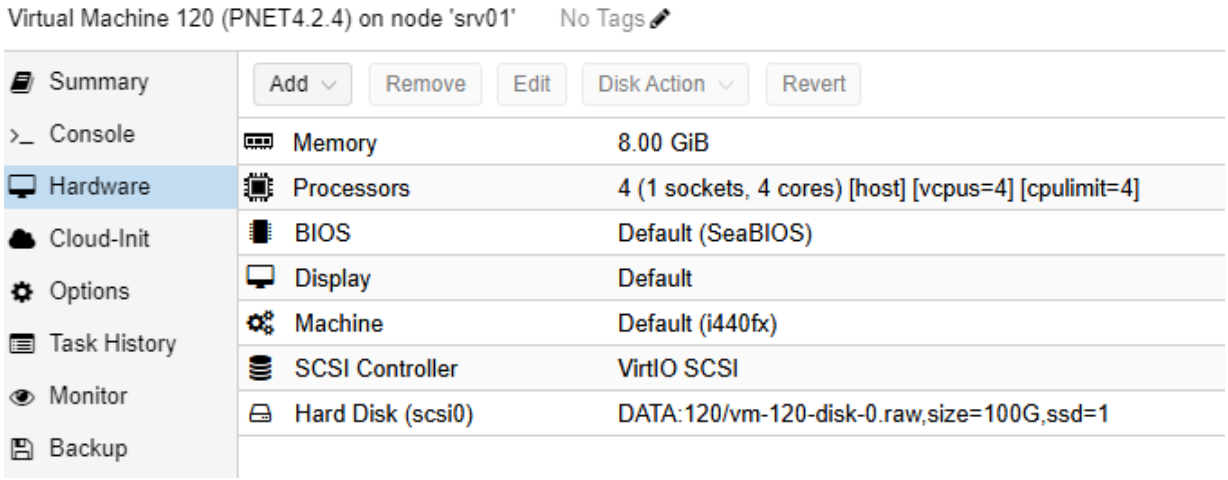
4 – Controlador SCSI, seleccionamos "***VirtIO SCSI***":



5 – En disco duro, seleccionamos la opción de "***SSD emulation***":

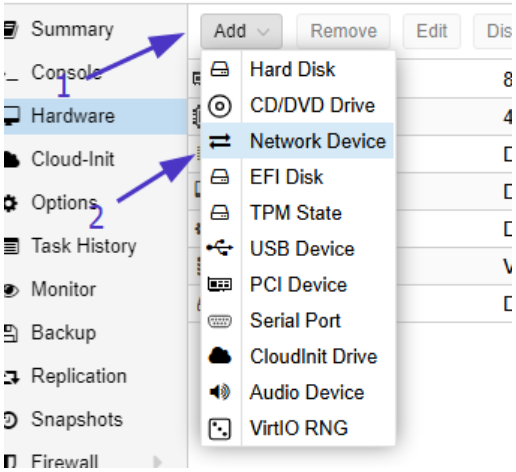


Así quedaría nuestra VM con recursos ajustados:

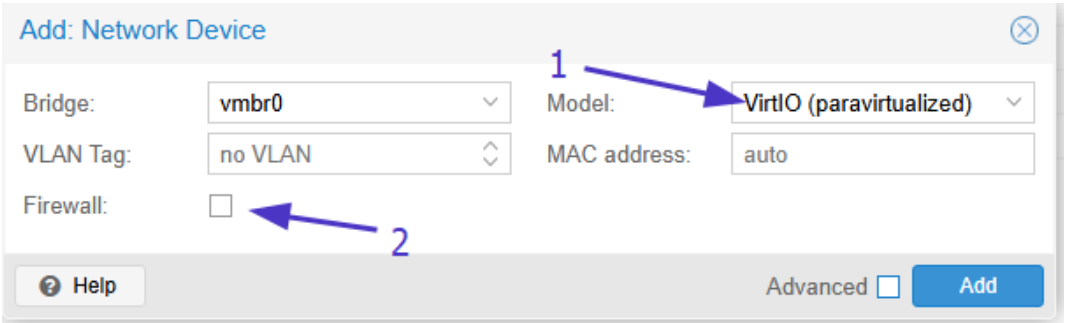


Sin embargo podemos ajustar RAM y Core, para el caso de disco duro y llegase a agotar espacio en nuestra VM, favor no aumentar el tamaño del disco, se recomienda que se agregue otro disco duro virtual, puesto que, automáticamente "***PNetLab***" lo dimensionará automáticamente.

Nos falta agregar una tarjeta de red o tarjetas de red según sea la necesidad que tenga:



Cuando agregue la tarjeta de red, seleccione el modelo "***VirtIO***" y demarque la casilla "***Firewall***":



Este es la configuración final, recuerde que es libre de ajustar la VM según los recursos que tenga en su HyperVisor o Proxmox VE:

Memory	8.00 GiB
Processors	4 (1 sockets, 4 cores) [host] [vcpus=4] [cpulimit=4]
BIOS	Default (SeaBIOS)
Display	Default
Machine	Default (i440fx)
SCSI Controller	VirtIO SCSI
Hard Disk (scsi0)	DATA:120/vm-120-disk-0.raw,size=100G,ssd=1
Network Device (net0)	virtio=BC:24:11:84:29:D5,bridge=vmbr0

Configuración de virtualización anidada "nested" y arranque de PNetLab

Al ser un entorno donde podremos disponer de máquinas con Docker, imágenes de Router, Switch de diferentes fabricantes, Proxmox(8.x) no viene por defecto activado la opción de "***nested***" o virtualización anidada, debemos verificar que tenga y activarlo cada vez que se inicie/reinicie el HyperVisor, esto lo realizamos vía SHELL:

cat /sys/module/kvm_intel/parameters/nested → Para el caso que sea INTEL

o

cat /sys/module/kvm_amd/parameters/nested → Para el caso que sea AMD

Si la salida a este comando es "N", debemos ejecutar lo siguiente:

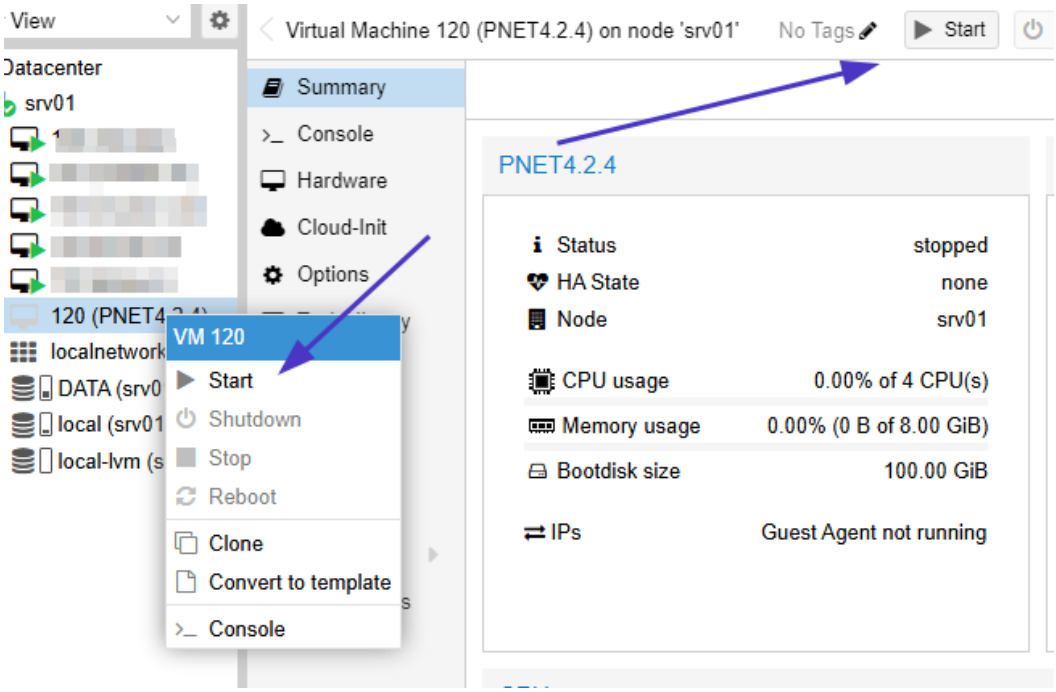
echo "options kvm-intel nested=Y" > /etc/modprobe.d/kvm-intel.conf → Para INTEL

o

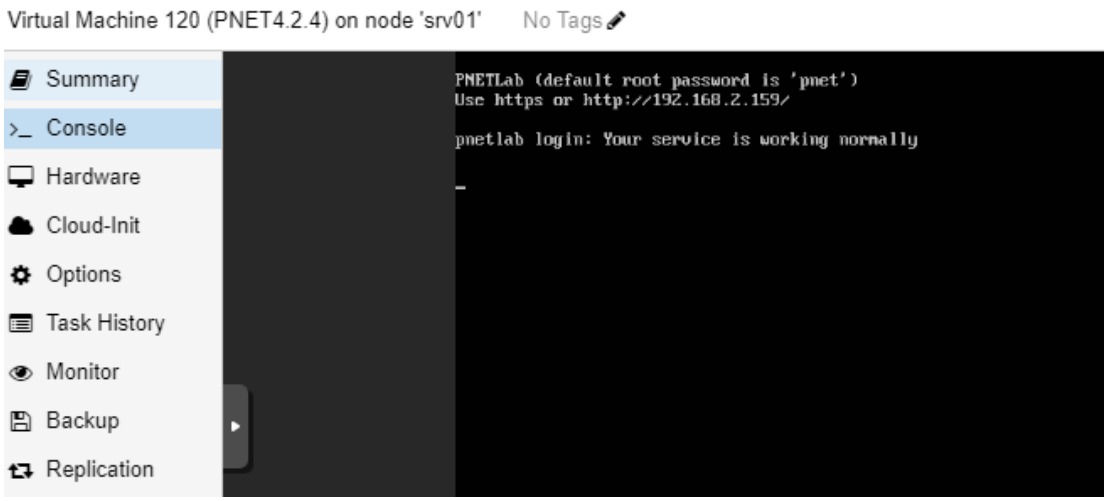
echo "options kvm-amd nested=Y" > /etc/modprobe.d/kvm-amd.conf → Para AMD

Tras esto, reiniciamos el HyperVisor "***Proxmox VE***".

Procedemos a iniciar nuestra VM "***PNetLab***", para esto hacemos clic en el botón superior derecha "***Start***" o haciendo clic sobre la VM y aparecerá la opción de "***Start***":



Esperamos a que finalice la carga del O.S. de "***PNetLab***":



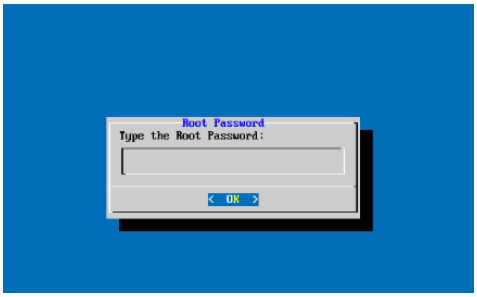
Ingresamos las credenciales para dar inicio al asistente de instalación:

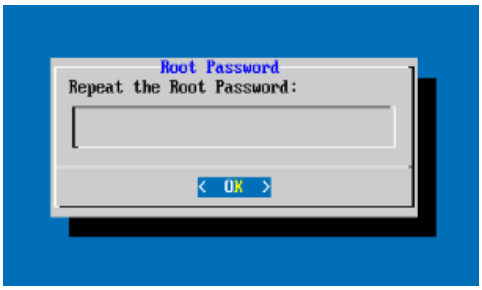
Por defecto:

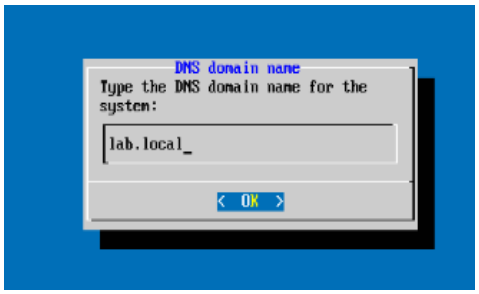
user → ***root***
pass → ***pnet***

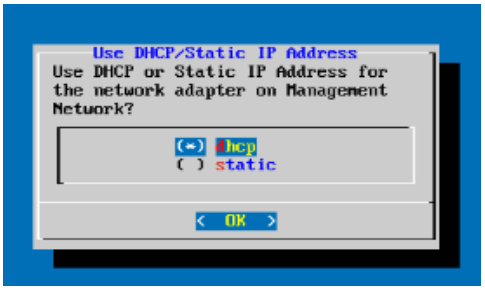
IP → si tiene un DHCP en su red, podrá visualizar la dirección IP asignada.

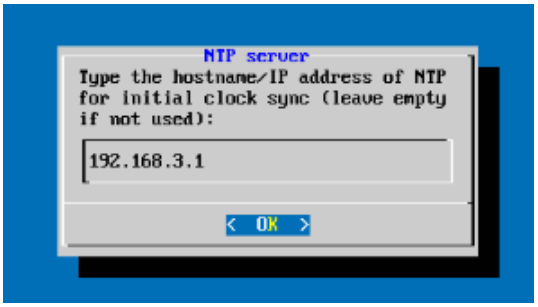
Una vez ingresado las credenciales, el asistente se ejecuta y seguimos los pasos ingresando datos según sea lo necesario, esto en cada escenario puede variar:

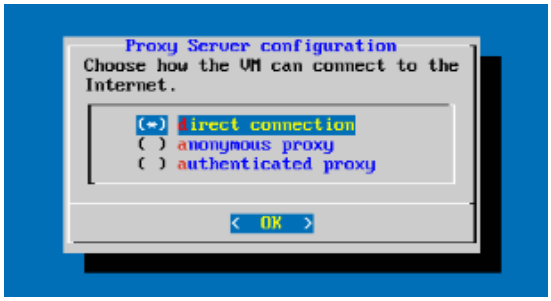




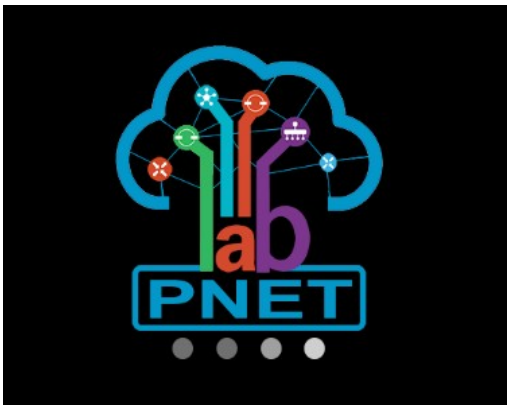




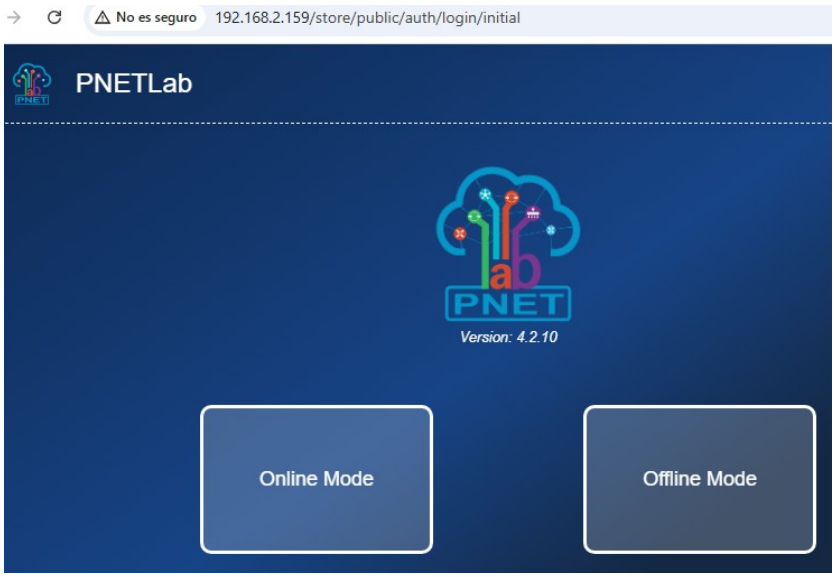




Luego la VM se reiniciará con los datos ingresados:



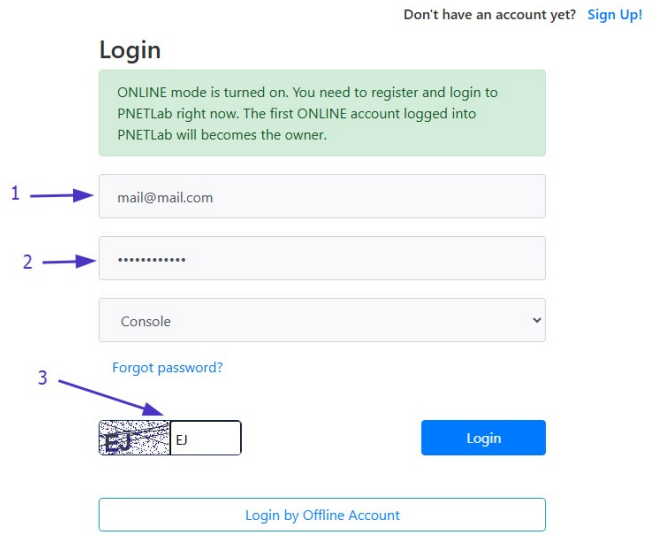
Tras el reinicio, ya podemos ingresar al entorno vía WEB ingresando en un navegador la dirección IP de PNetLab, de las cuales nos ofrece 2 modos como se muestra en la imagen siguiente:



Primer inicio en el entorno de PNetLab – Modos de ingreso

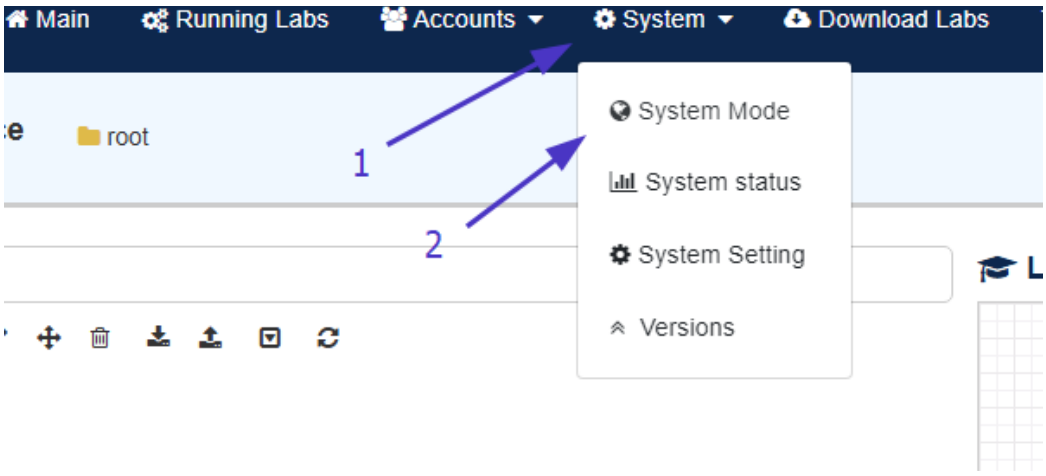
La opción de "***Online Mode***", es utilizado para colaborar en línea con labs que están realizando o resolviendo, ademas de disponer de descargar labs que tengan el permiso publico. Si no cuenta con registro, puede registrarse sin costo alguno.

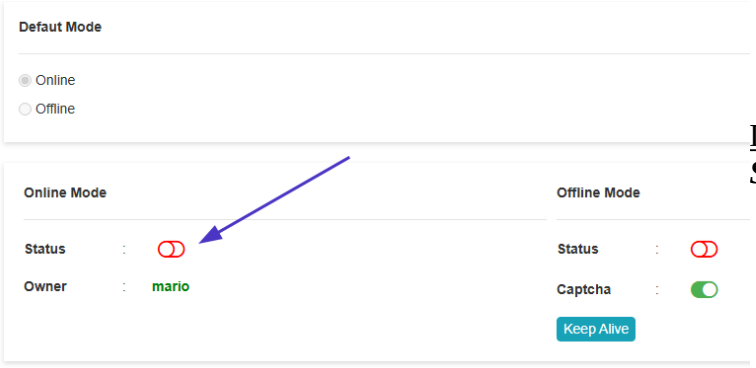




Nota: Es necesario el registro, para que pueda trabajar en "***Mode Offline***" y dejar de depender del acceso con internet para su uso.

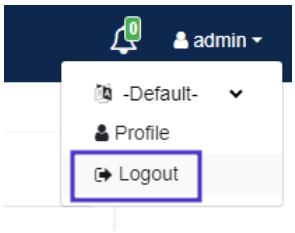
Estando logueado en "***Mode Online***", nos vamos a la opción de "***System/System Mode***":





“ <u>Desactivamos</u> el modo Online en el botón ***Status***”.

Nos salimos del entorno:

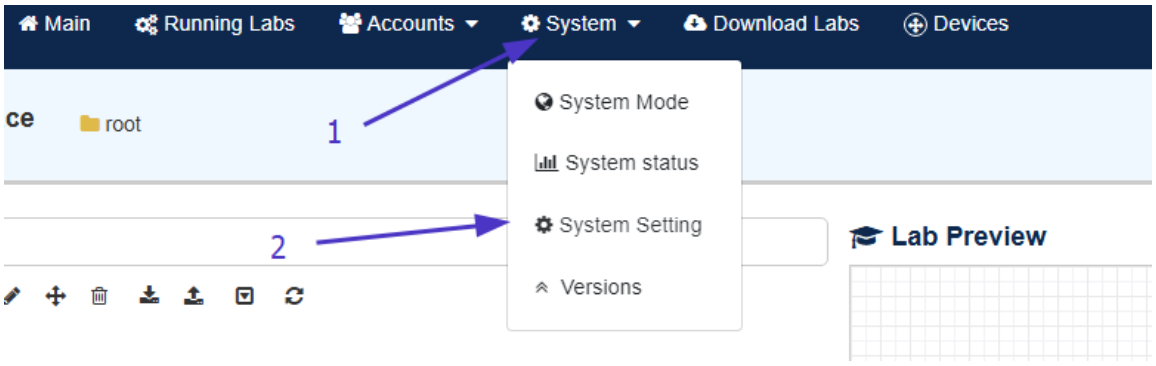


Luego se nos presenta este otro formulario, que tiene similitud al del “***Mode Online***”, pero esta vez nos indica que ingresemos con las credenciales por defecto que es “***admin/pnet***”, esto se puede cambiar en administración de usuarios:

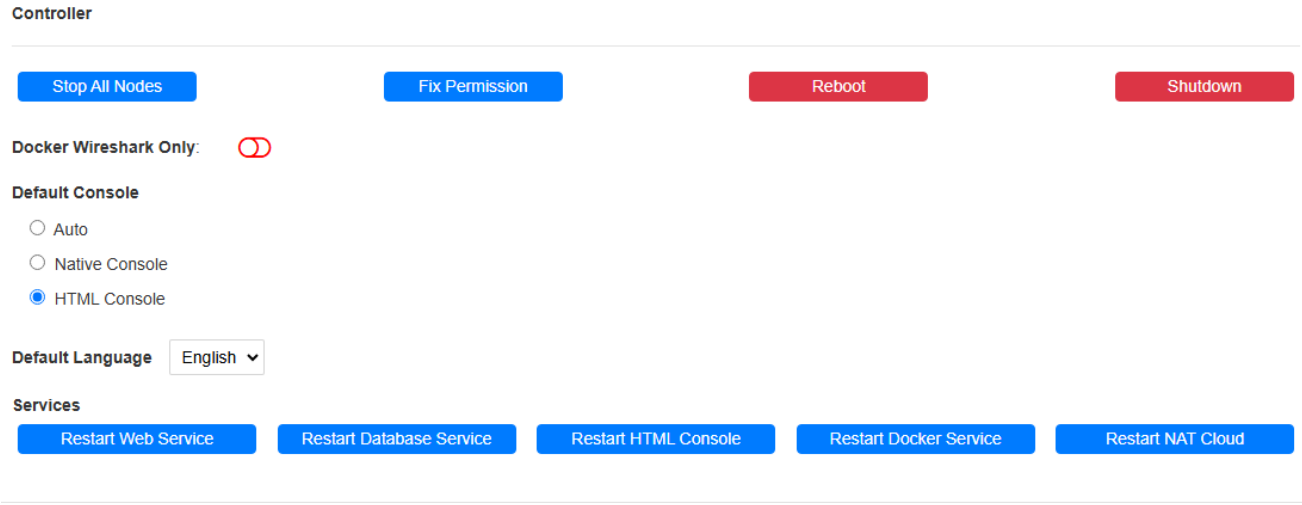


Explorando opciones de PNetLab

Vamos a explorar las opciones que estaremos ajustando, iniciaremos haciendo clic en "***System/System Settings***":

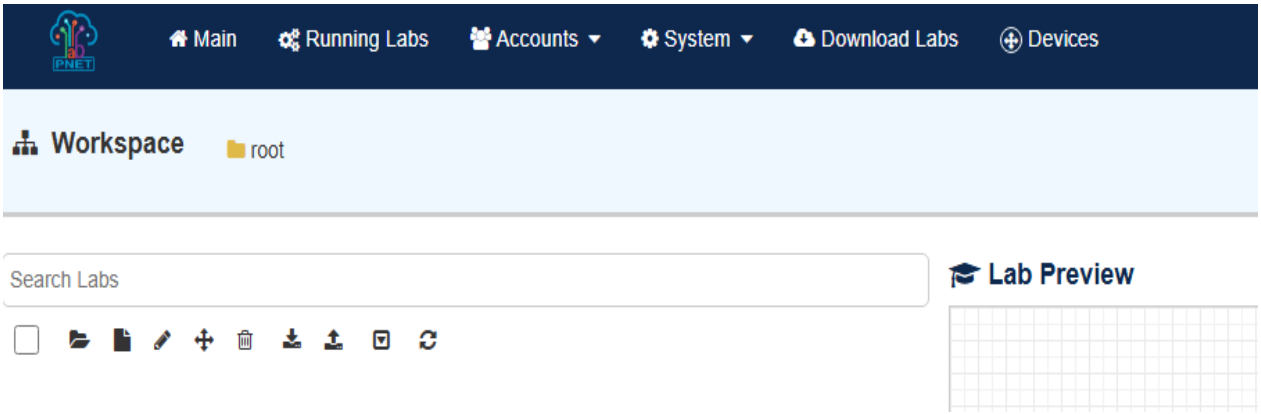


Las opciones típica que usaremos y será muy poco, es el botón de "***Reboot y Shutdown***", luego el botón que utilizaremos a menudo es "***Fix Permision***", éste nos ayudará a restablecer permisos de una imagen personalizada de algún fabricante. Todo esto se puede realizar bajo entorno SHELL.

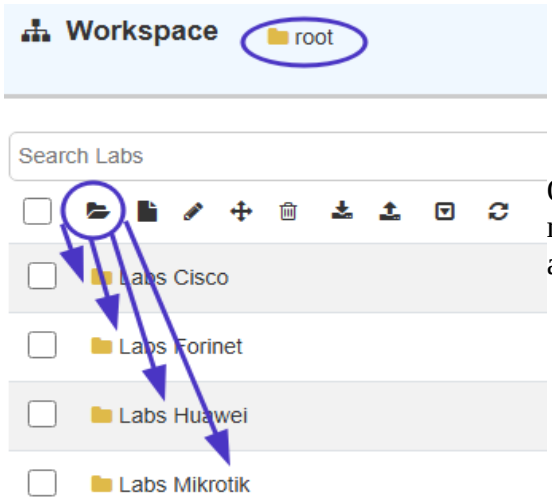


Las otras opciones nos ayudará a realizar reinicios de servicios, sin embargo pero no lo estaremos utilizando a menudo.

De las opciones del entorno de infraestructura virtual, observamos lo siguiente:

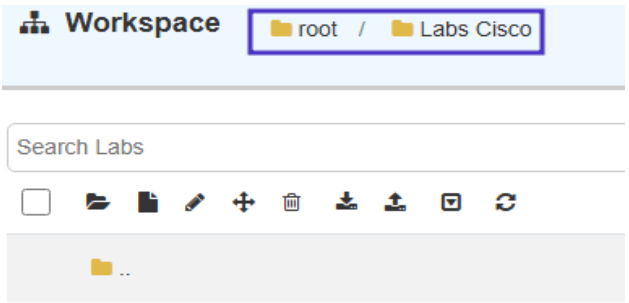


1 – Main: nos lleva a la ventana anterior, donde podremos crear carpetas de laboratorios, exportarlos, importarlos, eliminarlos, entre otros. Disponemos de un espacio donde se previsualizará un laboratorio.

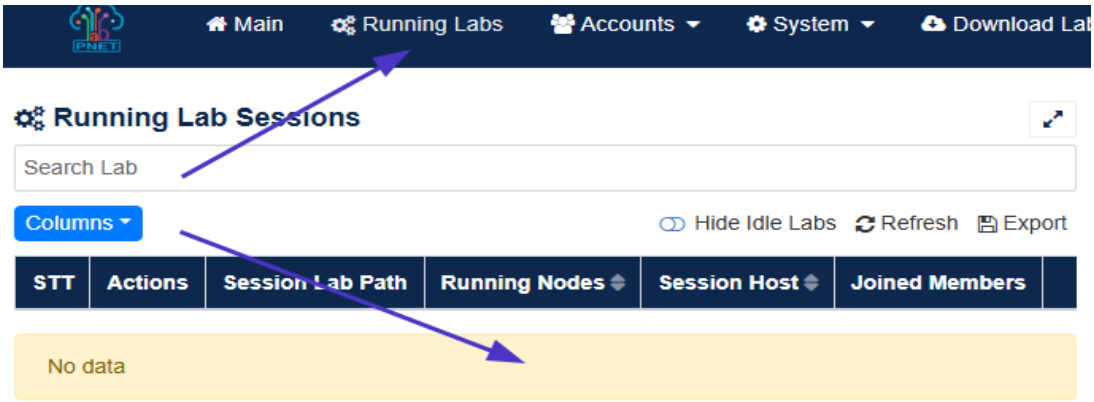


Observemos que podemos crear carpetas de labs o trabajos a recrear, estas carpetas estarán ubicadas en la raíz "***root***", de aquí podemos ver en que nivel de carpetas estamos

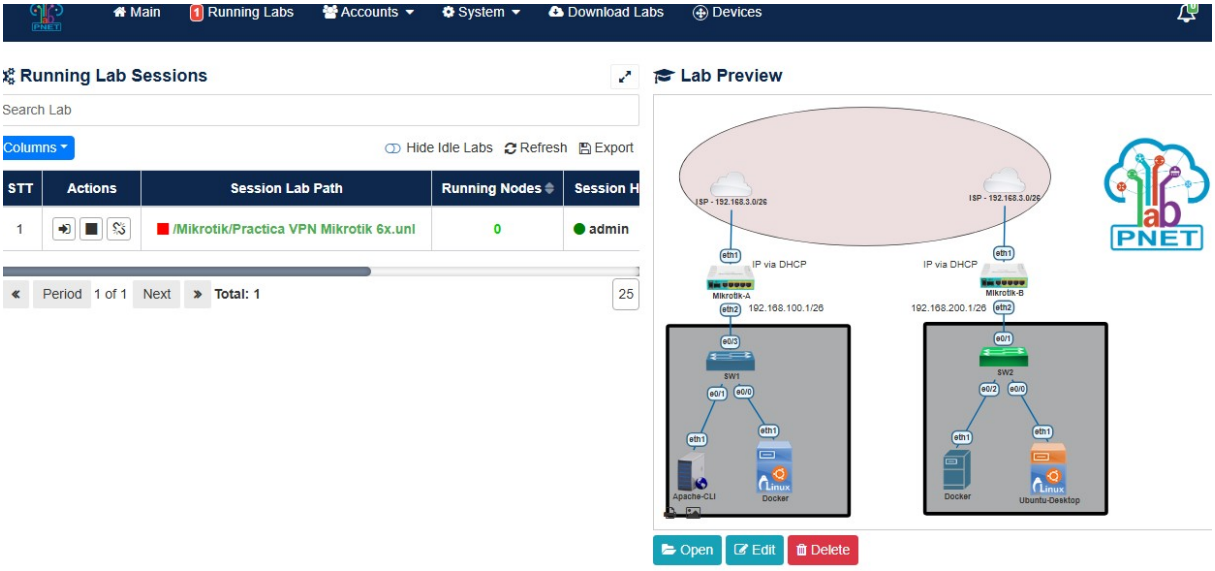
Observemos en el recuadro, la carpeta de trabajo "***Labs Cisco***" cuelga de "***root***", desde aquí podemos ir creando el laboratorio.



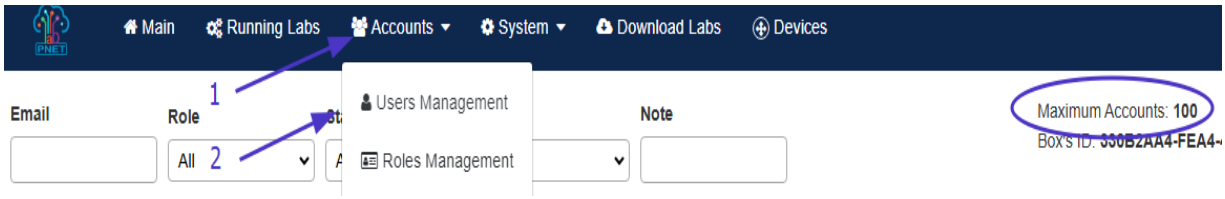
En la opción de "***Running Labs***", aparecerán los laboratorios o no, esto depende del uso que se de en ese momento:



Ejemplo en esta opción de "***Running Labs***":

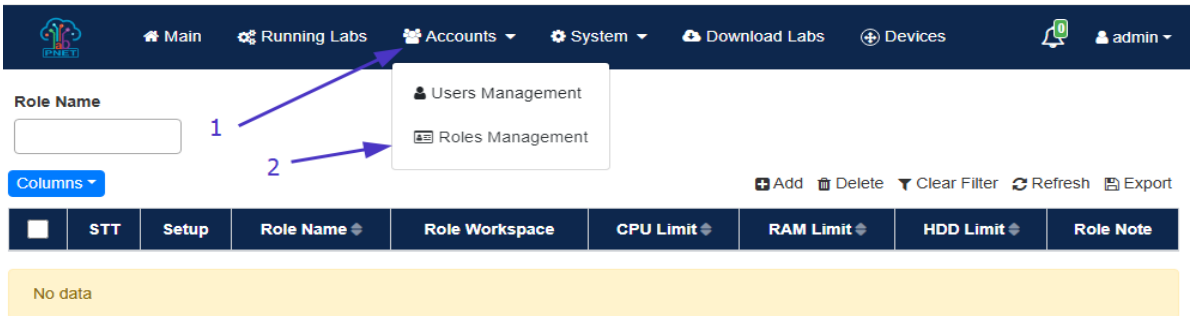


Los desarrolladores de PNetLab, no paran, esto porque a partir de esta versión, contamos con hasta ***100 cuentas*** incluyendo la cuenta de "***admin***" (esto en otros productos hay versiones limitadas y de paga), esto lo podremos ver en la opción de "***Accounts / Users Management***":

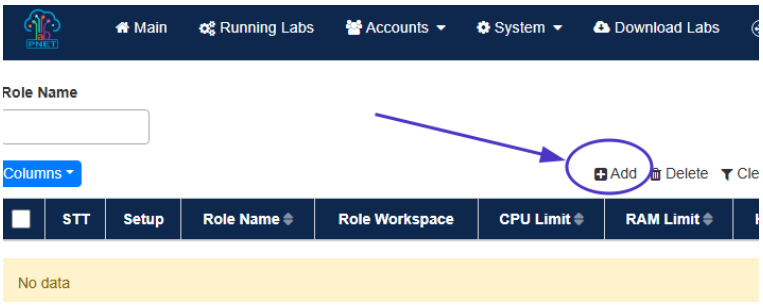


Aquí puede ir creando usuarios para ingresar al entorno, para luego asignarle un rol a cada usuario, los roles podremos crearlos y ajustarlos para limitar que debe hacer cada usuario.

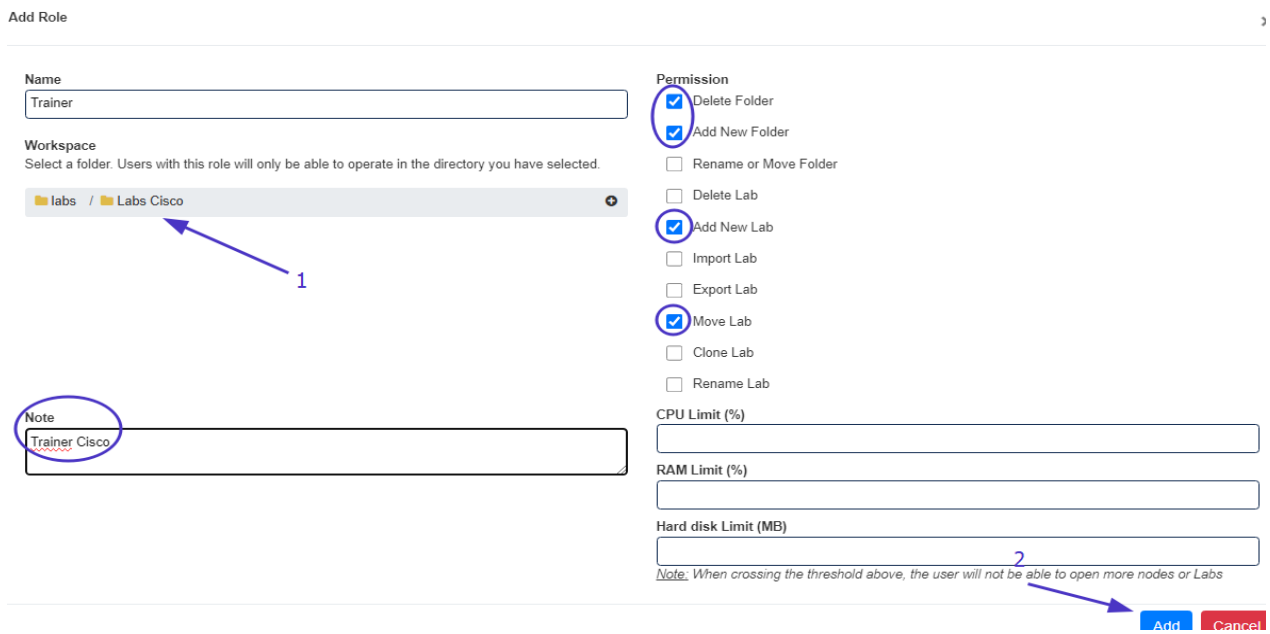
Dentro de esta opción tenemos de "***Accounts/Roles Management***":



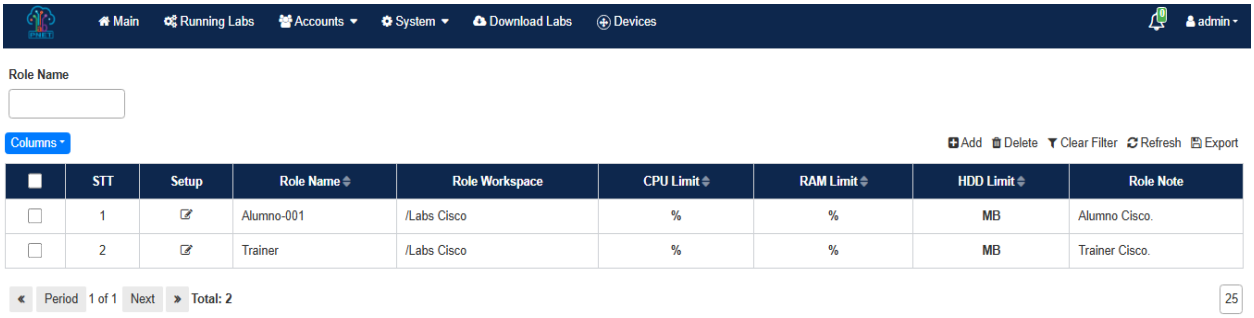
Aquí podemos ir agregando permisos a usuarios que roles podrán tener, limitarlos en recursos, etc:



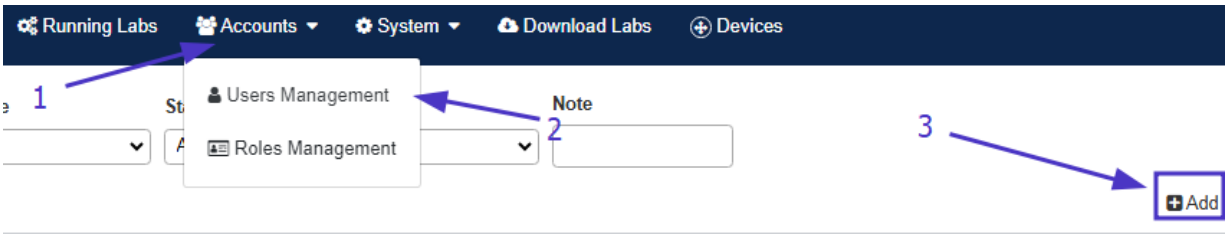
Este es el formulario de roles donde podremos ir ajustando a nuestras necesidades, para luego aplicarselo a un usuario determinado o a varios usuarios:



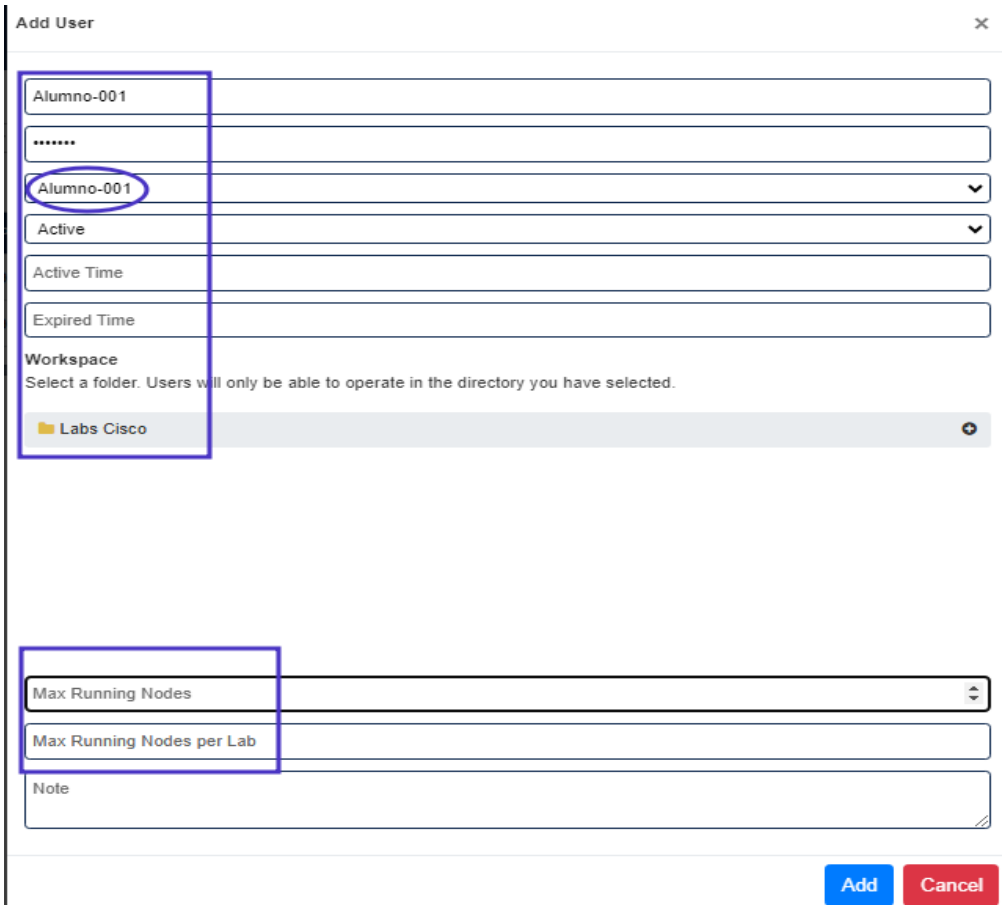
Visualización de 2 roles creados "***Alumno-001 y Training***":



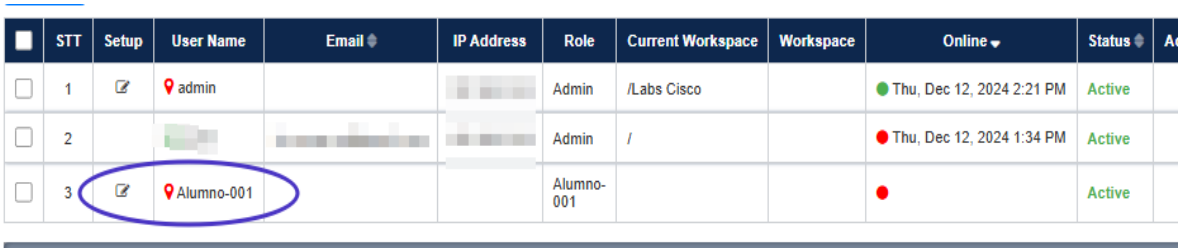
En este punto regresaremos a usuarios y crearemos uno de prueba y visualizaremos los roles que hemos creado y asignárselo a un usuario o varios usuarios:



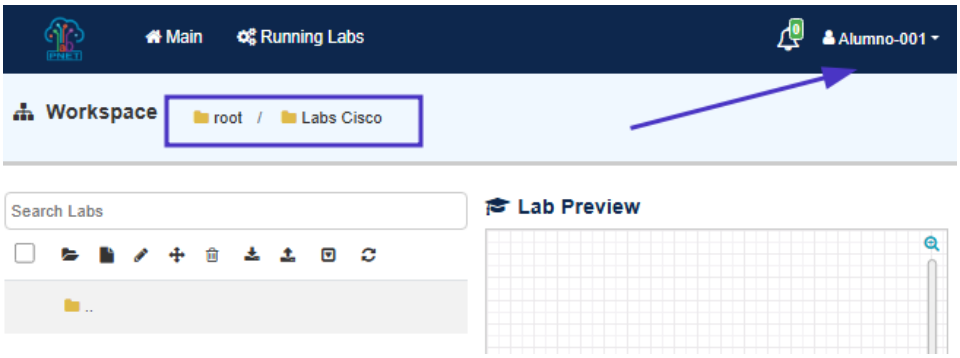
Ingresamos datos en el formulario del usuario y algún rol:



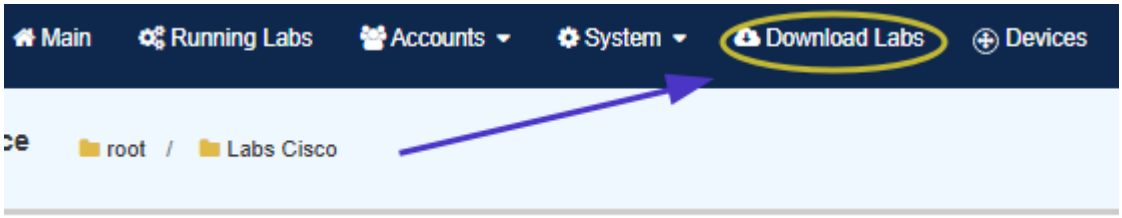
Ya podremos ver a los usuarios con un rol específico:



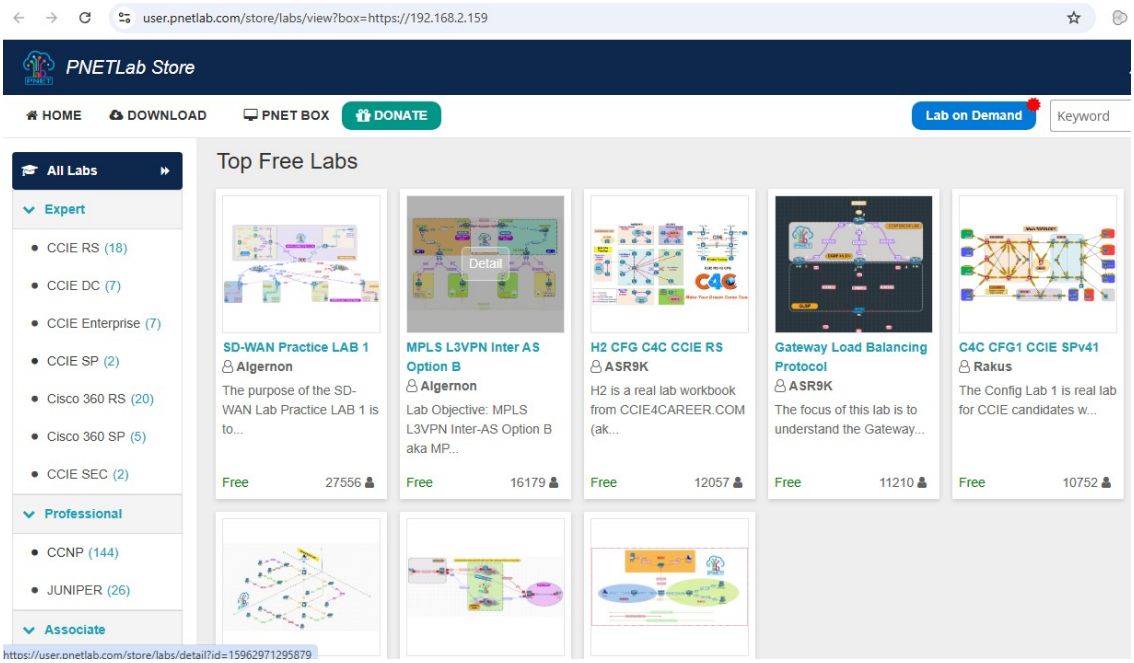
Probaremos a ingresar con el nuevo usuario:



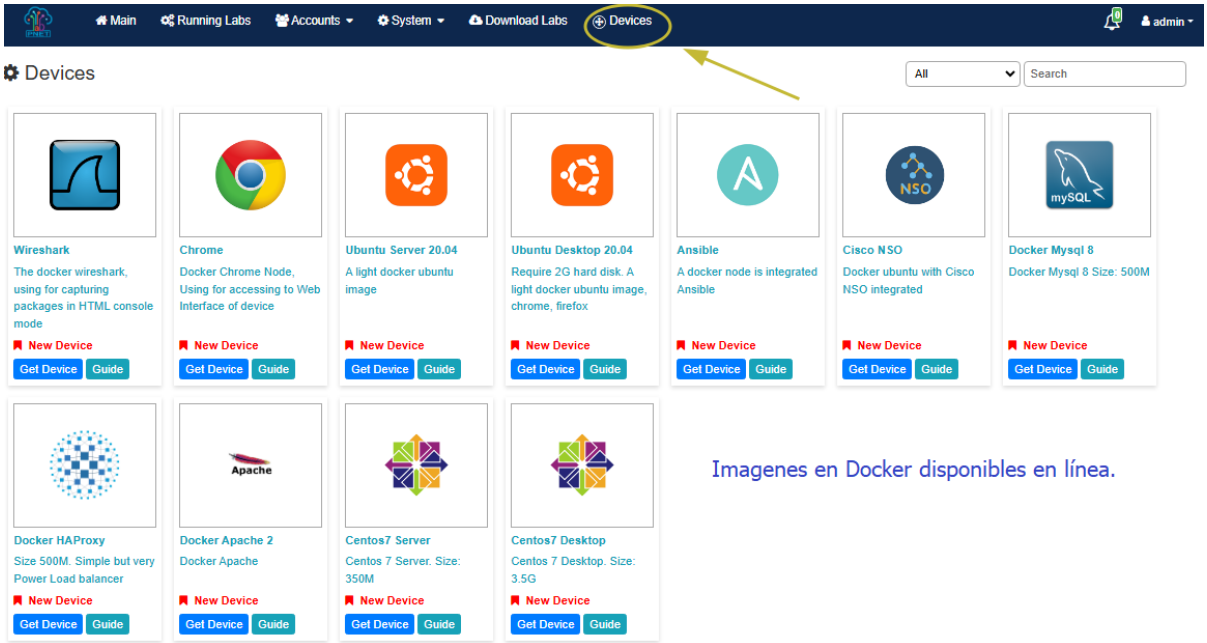
Seguimos con la opción de "***Download Labs***", aquí, el sistema nos dirige al repositorios de laboratorios que usuarios de todo el mundo crean y algunos son compartidos o públicos donde podremos descargarlo a nuestro servidor "***PNetLab***" y tenerlos disponibles para utilizarlos con los usuarios:



Son muchos los laboratorios que muchos usuarios ponen al público y ser descargados para práctica:



En la opción de "***Devices***", teniendo internet en la VM "***PnetLab***", podremos descargarlos a nuestro Server y utilizarlos en el entorno de laboratorio:



Actualización del Server PNetLab

Toca trabajar con los laboratorios, para esto es necesario actualizar el sistema operativo y utilizar una herramienta que nos facilite la descarga de las imágenes de algunos fabricantes, para esto procedemos a ingresar vía SHELL a la VM "***PNetLab***" y editamos la "***source.list***":

```
# nano /etc/apt/sources.list
```

Borramos todo y agregamos estas líneas, para mi caso me es mas cómodo:

deb http://us.archive.ubuntu.com/ubuntu bionic main restricted universe multiverse
deb http://us.archive.ubuntu.com/ubuntu bionic-updates main restricted universe multiverse
deb http://us.archive.ubuntu.com/ubuntu bionic-backports main restricted universe multiverse
deb http://us.archive.ubuntu.com/ubuntu bionic-security main restricted universe multiverse

deb [trusted=yes] https://repo.pnetlab.com ./
deb [trusted=yes] https://pnetlab.labhub.eu.org ./
#deb [trusted=yes] http://i-share.top/repo ./

```
deb http://vn.archive.ubuntu.com/ubuntu bionic main restricted universe multiverse

deb http://vn.archive.ubuntu.com/ubuntu bionic-updates main restricted universe multiverse

deb http://vn.archive.ubuntu.com/ubuntu bionic-backports main restricted universe multiverse

deb http://vn.archive.ubuntu.com/ubuntu bionic-security main restricted universe multiverse

deb [trusted=yes] https://repo.pnetlab.com ./

deb [trusted=yes] https://pnetlab.labhub.eu.org ./

#deb [trusted=yes] http://i-share.top/repo ./
```

Guardamos el fichero y salimos.

Ejecutamos el siguiente comando para actualizar los repositorios:

apt update

```
# apt update
```

```
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:10 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:11 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
Ign:10 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:11 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:10 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:15 https://repo.pnetlab.com ./ Packages
Ign:11 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:10 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:11 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
Ign:15 https://repo.pnetlab.com ./ Packages
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
Get:15 https://repo.pnetlab.com ./ Packages [4,899 B]
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
Ign:17 https://repo.pnetlab.com ./ Translation-en
Ign:18 https://repo.pnetlab.com ./ Translation-en_US
```

Ejecutamos el comando, e iniciamos la descarga de nuevos paquetes para el O.S.:

apt upgrade

```
apt upgrade
```

Podremos observar que necesitará descargar mas de 400M, aceptamos tecleando la letra “Y”:

```
  sudo systemd systemd-sysv tar tcpdump tmux tzdata ubuntu-advantage-too
  ubuntu-standard udev ufw uidmap unzip update-manager-core update-notif
  wireless-regdb xxd xz-utils zlib1g zlib1g-dev
384 upgraded, 13 newly installed, 0 to remove and 1 not upgraded.
Need to get 495 MB of archives.
After this operation, 409 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Esperamos a que finalice, esto tomará algún tiempo dependiendo de la velocidad y calidad del servicio de internet que se tenga.

Una vez descargado, procede a instalar los paquetes nuevos, puede aparecernos este mensaje, para esto tecleamos “***Enter***” de nuestro teclado, el “***Enter***” representa la letra “***N***” para esta pregunta, dejaremos la versión que trae el O.S., esto contiene un contenido en el fichero “***/etc/issue***”:

```
Configuration file '/etc/issue'
 ==> Modified (by you or by a script) since installation.
 ==> Package distributor has shipped an updated version.
   What would you like to do about it ?  Your options are:
    Y or I  : install the package maintainer's version
    N or O  : keep your currently-installed version
      D     : show the differences between the versions
      Z     : start a shell to examine the situation
 The default action is to keep your current version.
*** issue (Y/I/N/O/D/Z) [default=N] ?
Progress: [  0%] [.....................................................
```

Si nos aparece al este error de paquete relacionado con “***docker***”:

```
Errors were encountered while processing:
 /tmp/apt-dpkg-install-ySbZya/056-docker.io_20.10.21-0ubuntu1~18.04.3_amd64.deb
E: Sub-process /usr/bin/dpkg returned an error code (1)
root@pnetlab:/var/lib/apt#
```

Procederemos a corregirlo:

rm -rf /var/lib/docker/aufs && apt remove docker.io

```
rm -rf var/lib/docker/aufs/
apt remove docker.io
```

Luego proceder con el siguiente comando:

apt --fix-broken install

```
apt --fix-broken install
```

Este error, obedece a paquetes obsoletos o no se encuentran o entran en conflicto con el nuevo paquete descargado, para esto se ejecutan los comandos anteriores para eliminar la versión de docker del O.S. y corregirlos con el nuevo paquete.

Una vez corregido el paquete en cuestión, procedemos a limpiar paquetes viejos y/o ficheros de configuración que no se utilizaran mas tras la actualización del O.S., esto lo realizaremos con 2 comandos:

apt autoremove && apt autoclean

Luego tecleamos "***Y***" o "***Enter***" y reniciamos la VM, después de finalizar los comandos anteriores

```
root@pnetlab:/# apt autoremove && apt autoclean
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  libnetplan0 linux-headers-4.15.0-142 linux-headers-4.15
  linux-modules-4.15.0-142-generic linux-modules-extra-4.
0 upgraded, 0 newly installed, 6 to remove and 1 not upgr
After this operation, 350 MB disk space will be freed.
Do you want to continue? [Y/n]
```

Hasta aquí la finalización del S.O., es recomendable aplicar un reinicio una vez finalizado esta tarea de actualización.

Instalación de la herramienta ishare2

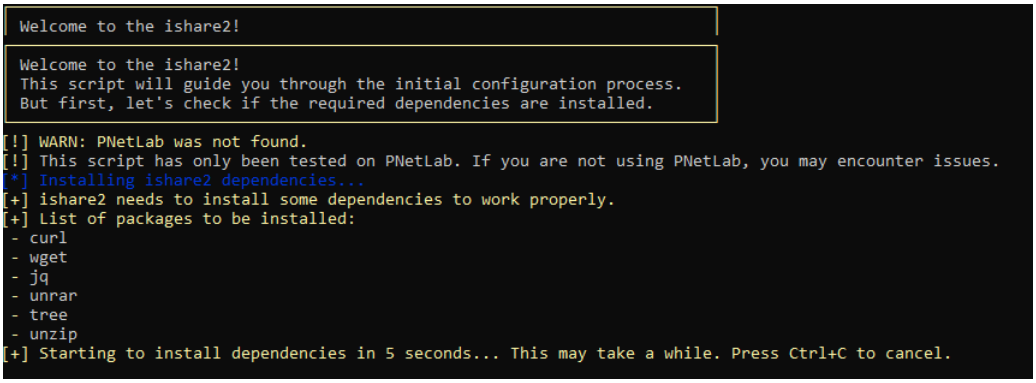
Tras el reinicio, procedemos a instalar la herramienta que nos ayudará a descargar las imágenes de algunos fabricantes, como: “Cisco, Fortinet, Mikrotik, Huawei, etc”. Esto lo realizamos vía SHELL, para esto ingresamos al Server VM “***PNetLab***”. Más información ir al url https://github.com/ishare2-org/ishare2-cli:

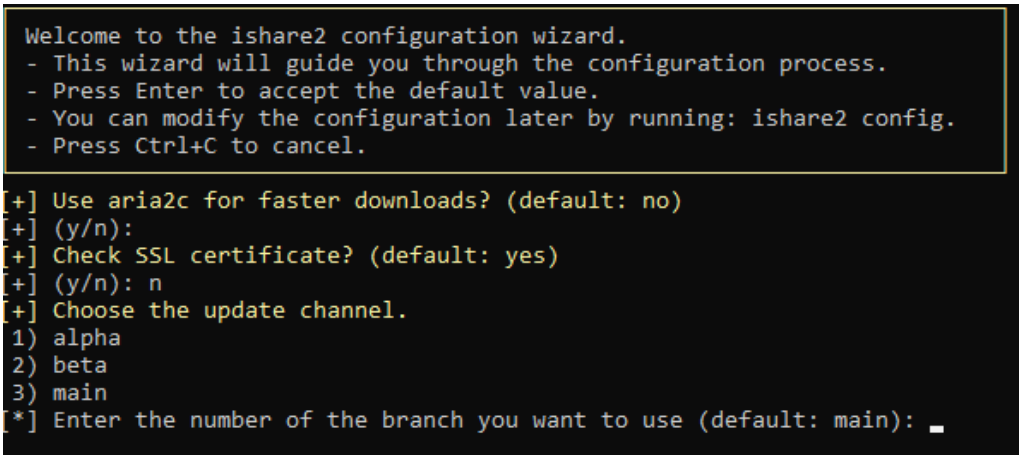
Ejecutamos la siguiente línea de comando:

wget -O /usr/sbin/ishare2 https://raw.githubusercontent.com/ishare2-org/ishare2-cli/main/ishare2 && chmod +x /usr/sbin/ishare2 && ishare2

Nota: Repositorio de imágenes de diferentes fabricantes → https://drive.labhub.eu.org

Tras la ejecución del comando, nos aparece un asistente, seguir los pasos:





Continuamos con la instalación siguiendo los pasos:

```
[+] Use aria2c for faster downloads? (default: no)
[+] (y/n):
[+] Check SSL certificate? (default: yes)
[+] (y/n): n
[+] Choose the update channel.
 1) alpha
 2) beta
 3) main
[*] Enter the number of the branch you want to use (default: main):
[!] Using the default branch.
[+] Choose a mirror. (default: Rotate mirrors)
 1) Rotate mirrors (recommended)
 2) Google Drive mirror
 3) Onedrive mirror
 4) Custom mirror
[*] Enter the number of the mirror you want to use (default: 1): 1
```

Si encuentra algún paquete no instalado en nuestro server "***PNetLab***" y lo necesita, el asistente lo descargará e instalará automáticamente. Si no aparece nada pendiente que descargar e instalar, nos mostrará que ishare2 se ha instalado correctamente:

```
 Telegram: https://t.me/NetLabHub
 Donate: https://buymeacoffee.com/sudoalex
 GitHub: https://github.com/ishare2-org/ishare2-cli

yntax

ishare2 [action] [param1] [param2] [--overwrite]

ction:
   search      : Search for images available on LabHub mirrors.
   pull        : Download an image by specifying the type and id.
   installed   : Shows images installed on the server.
   labs        : Shows available labs and downloads the images for the selected lab.
   mylabs      : Same as labs command but you can use a customized path.
   relicense   : Generates an iourc license for iol images.
   upgrade     : Shows a menu to upgrade ishare2 or PNETLab.
   test        : Test internet connectivity to required sites.
   help        : Shows full help information.

aram1:
   type = all, iol, qemu, dynamips, docker or name.

aram2:
   id = This can be obtained using ishare2 search <type>

-overwrite:
   Used to overwrite an existing image if it already exists on your system.

ry: ishare2 help for more information.
oot@pnetlab:/#
```

Uso de la herramienta ishare2

Utilizando **iS*hare2***:

Estando en la SHELL, procedemos a ejecutar el comando en búsqueda de imágenes como dispositivos L2 y L3 de Cisco, por ejemplo imagenes iOL:

ishare2 search l2

```
root@pnetlab:/# ishare2 search l2
============================
    Available QEMU images
============================
ID   NAME                                          SIZE
--   ----                                          ----
540  iosvl2-2020                                   86.2 MiB
930  viosl2-15.2.4.55e                             92.4 MiB
931  viosl2-adventerpriseK9-M_152_May_2018         102.2 MiB
932  viosl2-adventerprisek9-m-15.2.4055            92.4 MiB
933  viosl2-adventerprisek9-m-v152_6_0_81_e-20190423  39.2 MiB
934  viosl2-adventerprisek9-m.SSA.high_iron_20180619  42.5 MiB
935  viosl2-adventerprisek9-m.ssa.high_iron_20190423  42.9 MiB
936  viosl2-adventerprisek9-m.ssa.high_iron_20200929  85.6 MiB

8 QEMU images found for the term: "l2"

============================
    Available IOL images
============================
ID  NAME                                                    SIZE
--  ----                                                    ----
16  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20180510.bin  120.4 MiB
17  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin  120.4 MiB
18  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin  120.4 MiB
19  i86bi_linux_l2-advipservicesk9-ms.high_iron_20170202.bin      100.2 MiB
20  i86bi_linux_l2-ipbasek9-ms.high_iron_aug9_2017b.bin           100.8 MiB
21  i86bi_linux_l2-ipbasek9-ms.may8-2013-team_track.bin           58.9 MiB

6 IOL images found for the term: "l2"

==============================
    Available DYNAMIPS images
==============================
ID  NAME  SIZE
--  ----  ----

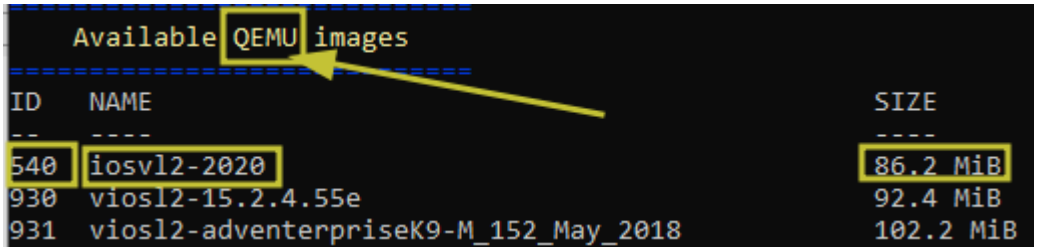
No DYNAMIPS images found for the term: "l2"

root@pnetlab:/#
```

Para poder descargar la imagen que nos interesa, la sintaxis es:

ishare2 pull <tipo de imagen> <id de imagen>

Por ejemplo, descargar la imagen iosvl2-2020, es necesario conoce el número id de la imagen, el cual es 540, el tipo es qemu, observemos los datos en la siguiente captura de imagen:



Teniendo ese dato, ejecutamos lo siguiente:

ishare2 pull qemu 540

```
root@pnetlab:/# ishare2 pull qemu 540 --overwrite
[!] IMAGE INFO
 - Image Name       : iosvl2-2020
 - Image Size       : 86.2 MiB
 - Image Type       : QEMU
 - Image ID         : 540
 - Image path       : /opt/unetlab/addons/qemu/iosvl2-2020
 - Using host       : https://labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/qemu/iosvl2-2 100%[=========================
/opt/unetlab/addons/qemu/iosvl2-2 100%[=========================
[+] DOWNLOAD COMPLETED!
cp: cannot create regular file '': No such file or directory
[-] Fixing permissions...

[+] Fix permissions command has been executed correctly
root@pnetlab:/#
```

Si buscamos imágenes del tipo iOL para cisco, ahí tendríamos imágenes L2 y L3:

```
root@pnetlab:/# ishare2 search iol
============================
    Available IOL images
============================
ID  NAME                                                    SIZE
--  ----                                                    ----
1   L2-ADVENTERPRISEK9-M-15.2-20150703.bin                  105.8 MiB
2   L2-ADVENTERPRISEK9-M-15.2-IRON-20151103.bin             108.3 MiB
3   L2-Adventerprisek9-ms.nov3_2015_high_iron.bin           108.3 MiB
4   L3-ADVENTERPRISEK9-M-15.4-2T.bin                        152.0 MiB
5   c2600-adventerprisek9-mz.124-25d.bin                    28.6 MiB
6   c2600-advsecurityk9-mz.124-15.t14.bin                   18.7 MiB
7   c3660-a3jk9s-mz.124-15.T14.bin                          40.5 MiB
8   c3660-a3jk9s-mz.124-25d.bin                             35.1 MiB
9   i86bi_Linux-L2-AdvEnterpriseK9-M_152_May_2018.bin       120.4 MiB
10  i86bi_Linux-L2-Adventerprisek9-ms.SSA.high_iron_20190423.bin  120.4 MiB
11  i86bi_Linux-L3-AdvEnterpriseK9-M2_157_3_May_2018.bin    176.2 MiB
12  i86bi_LinuxL2-AdvEnterpriseK9-M_152_May_2018.bin        120.4 MiB
13  i86bi_LinuxL3-AdvEnterpriseK9-M2_157_3_May_2018.bin     176.2 MiB
14  i86bi_LinuxL3-AdvEnterpriseK9-M2_157_3_May_2018.bin     176.2 MiB
15  i86bi_linux-adventerprisek9-ms.154-1.T_AntiGNS3.bin     145.6 MiB
16  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20180510.bin  120.4 MiB
17  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin  120.4 MiB
18  i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin  120.4 MiB
19  i86bi_linux_l2-advipservicesk9-ms.high_iron_20170202.bin  100.2 MiB
20  i86bi_linux_l2-ipbasek9-ms.high_iron_aug9_2017b.bin      100.8 MiB
21  i86bi_linux_l2-ipbasek9-ms.may8-2013-team_track.bin      58.9 MiB
22  i86bi_linux_l3-L3-ADVENTERPRISEK9-M-15.4-2T.bin          152.0 MiB

22 IOL images found
```

Descargaremos las imágenes con id "18 y 22", con esto tendremos imágenes L2 y L3:

ishare2 pull iol 18 && ishare2 pull iol 22

```
root@pnetlab:/# ishare2 pull iol 18  &&  ishare2 pull iol 22
[!] IMAGE INFO
 - Image Name       : i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin
 - Image Size       : 120.4 MiB
 - Image Type       : IOL
 - Image ID         : 18
 - Image path       : /opt/unetlab/addons/iol/bin/
 - Using host       : https://drive.labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/iol/bin//i86b 100%[============================================================>] 120.40M  22.7MB/s    in 5.4s
[+] Basic integrity checks passed. The downloaded image is the expected size.
[+] DOWNLOAD COMPLETED!
[!] IMAGE INFO
 - Image Name       : i86bi_linux_l3-L3-ADVENTERPRISEK9-M-15.4-2T.bin
 - Image Size       : 152.0 MiB
 - Image Type       : IOL
 - Image ID         : 22
 - Image path       : /opt/unetlab/addons/iol/bin/
 - Using host       : https://labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/iol/bin//i86b 100%[============================================================>] 151.97M  8.97MB/s    in 14s
[+] Basic integrity checks passed. The downloaded image is the expected size.
[+] DOWNLOAD COMPLETED!
```

Donde están esas imágenes? Estas se encuentran en la ruta siguiente:

/opt/unetlab/addons/

```
/opt/unetlab/addons#
```

En ese directorio, están los diferentes tipos de imagenes "dynamips, iol, qemu":

cd /opt/unetlab/addons/
/opt/unetlab/addons/ls

```
root@pnetlab:/opt/unetlab/addons# ls
dynamips  iol  qemu
```

Verificamos que imágenes tenemos en el directorio iol:

```
/opt/unetlab/addons# cd iol/
/opt/unetlab/addons/iol# _
```

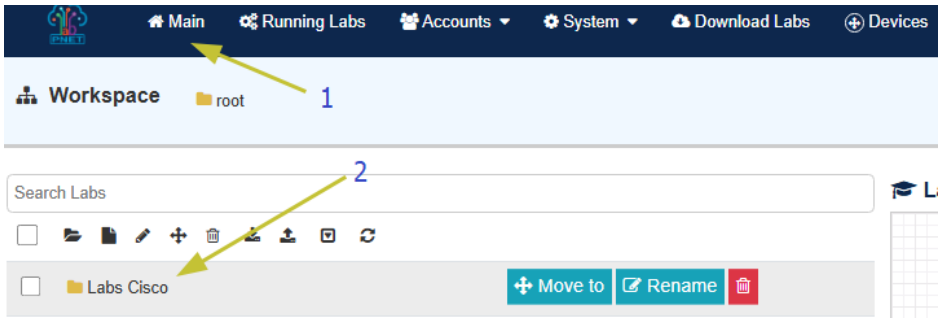
```
root@pnetlab:/opt/unetlab/addons# cd iol/
root@pnetlab:/opt/unetlab/addons/iol# cd bin/
root@pnetlab:/opt/unetlab/addons/iol/bin# ls
CiscoIOUKeygen.py                                i86bi_linux_l3-L3-ADVENTERPRISEK9-M-15.4-2T.bin  keepalive.pl
i86bi_linux_l2-adventerprisek9-ms.SSA.high_iron_20190423.bin  iourc
root@pnetlab:/opt/unetlab/addons/iol/bin# _
```

Ahora las imágenes qemu:

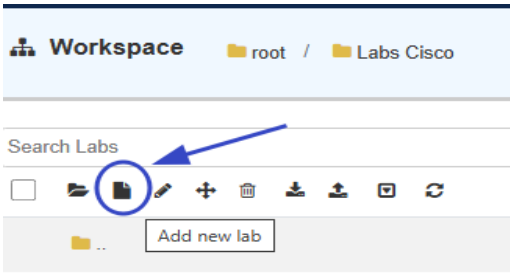
```
root@pnetlab:/opt/unetlab/addons/iol/bin# cd ..
root@pnetlab:/opt/unetlab/addons/iol# cd ..
root@pnetlab:/opt/unetlab/addons# cd qemu
root@pnetlab:/opt/unetlab/addons/qemu# ls
iosvl2-2020
root@pnetlab:/opt/unetlab/addons/qemu#
```

Creando un laboratorio y utilizar las imágenes

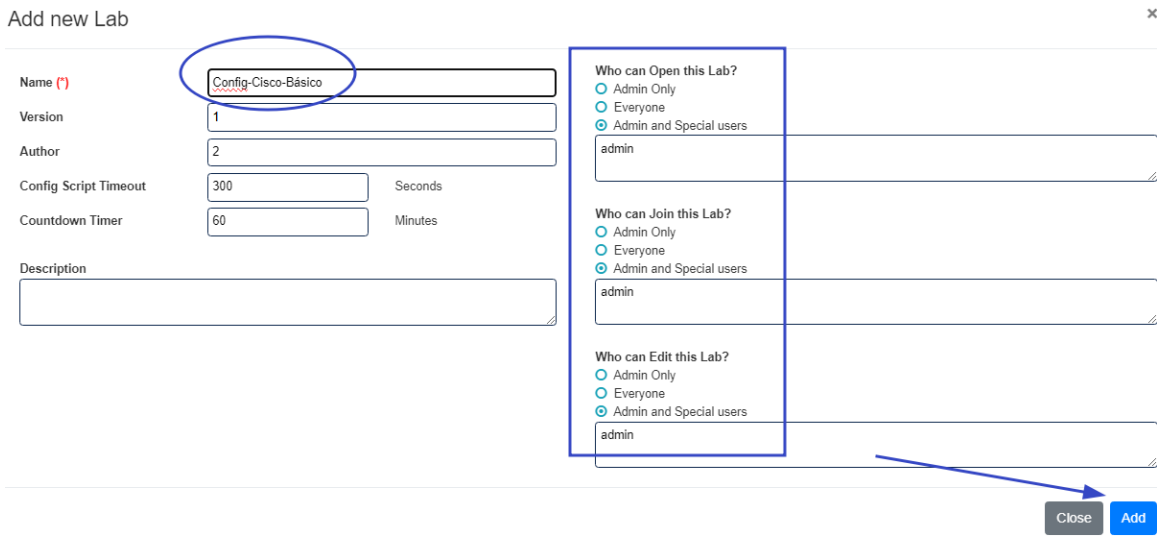
Procedemos a verificar en el entorno de PNetLab, para esto, vamos a crear una lab, en el directorio "***Labs Cisco***":



Creamos un lab llamado "***Config-Cisco-Básico***"

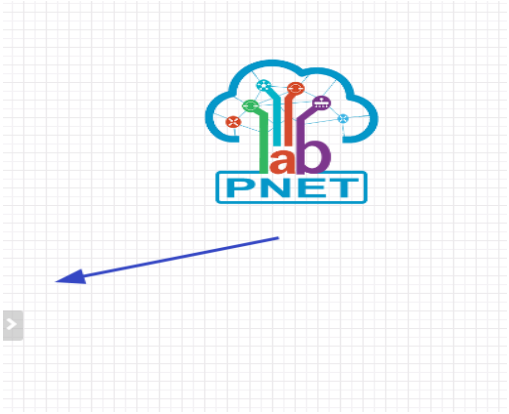


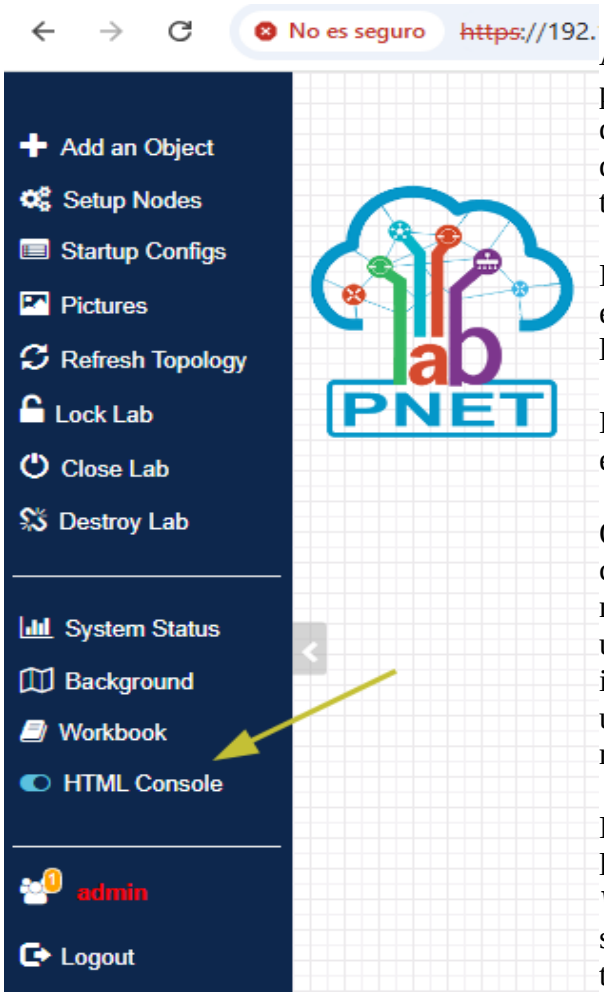
Aquí ingresamos datos que necesitamos:



Una vez agregado, nos redirigir a un enlace donde esta una especie de espacio de trabajo, ahí podremos ir recreando nuestro laboratorio:

En la flecha color gris, si hacemos clic, se desplegará una barra de menú con opciones:





Activamos la opción de consola "***HTML Console***", podrá ver otras opciones como, "***Destroy Lab***", aquí sale del lab y las configuraciones no se guardaran, esto dependerá del tipo de imagen que se trabaje, puesto que todo laboratorio por lo general queda limpio por defecto.

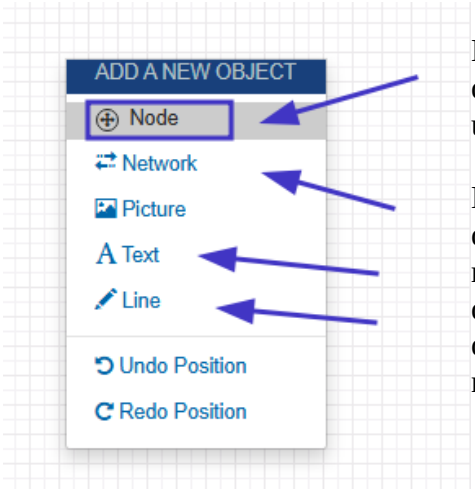
La opción "***Close Lab***", saldrá del lab, pero quedará ejecutada en segundo plano, por tanto, podrá regresar al laboratorio para finalizar alguna tarea pendiente.

La opción de "***System Status***", mostrará recursos que se este utilizando en ese momento del laboratorio creado.

Opción de "***Refresh Topology***", refrescará el laboratorio cuando se deje de trabajar en él por un determinado momento, ejemplo que se haya cerrado la sesión de usuario y regresa. Como también cuando subamos una imagen por ishare2, y en el laboratorio queramos usar una imagen recién descargado, para esto clic en refrescar la topología.

La opcón de "***Workbook***", podemos subir una guía del laboratorio creado, para seguir una practica, el ***Workbook***, debe estar prepara y en formato PDF, para subirlo y este disponible para el laboratorio que se este trabajando.

Esas serían las opciones mas utilizadas. Ahora vamos a agregar las imágenes descargadas:

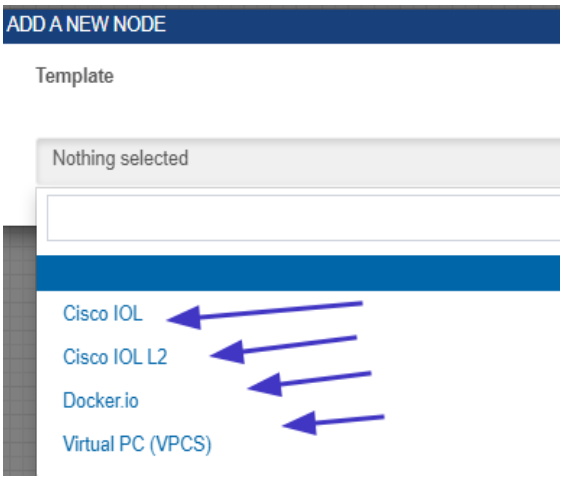


Estando en el área de trabajo, hacemos clic con el botón derecho del Mouse y se nos despliega un conjunto de opciones, entre esas utilizaremos la opción "***Node***" y "***Network***".

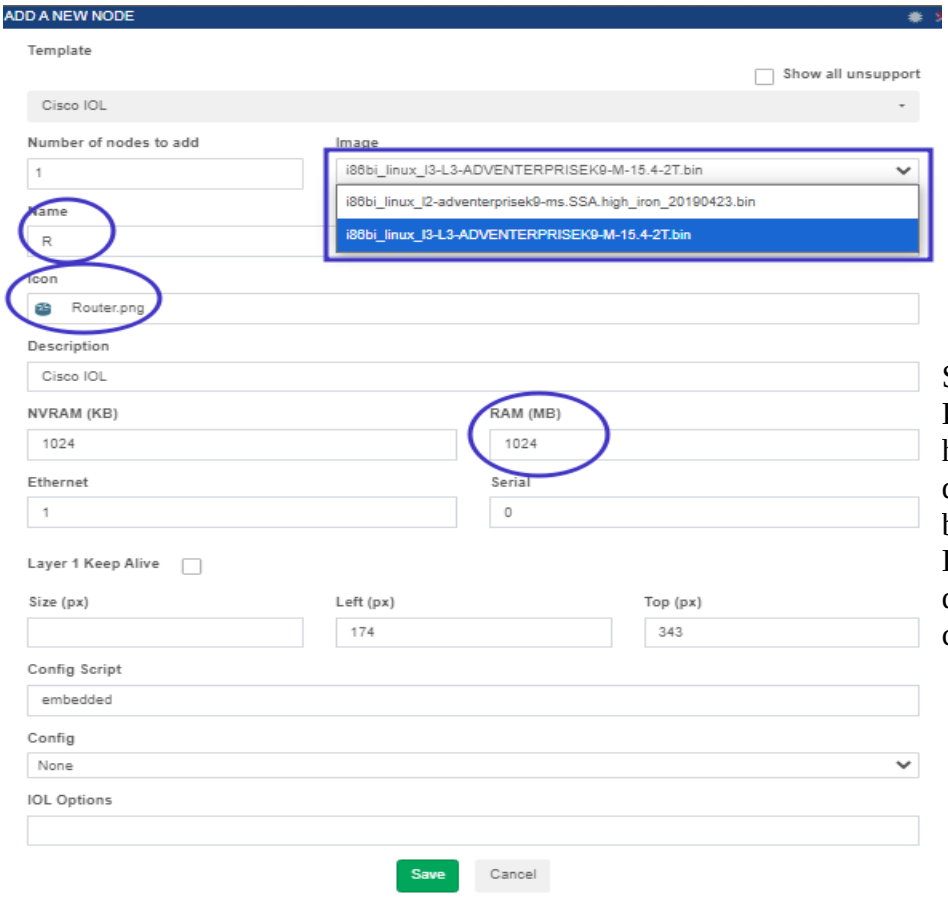
El "***Node***" es donde seleccionaremos la imagen o imágenes descargas para el laboratorio, el "***Network***" es un nube que representa la red externa al entorno virtual, el cual interactuará con dispositivos reales fuera del entorno, esto es una buena opción por que podemos probar nuestro entorno virtual con recursos reales de nuestra red.

En la opción de "***Node***", dispondremos de las imágenes descargadas a utilizar, como verá están las imágenes CISCO IOL, CISCO IOL2, pero también dispondremos de imágenes de Docker, tecnología que nos brinda un escenario más real, puesto que podemos crear laboratorios con imágenes de O.S. sea este Linux, Windows, entre otros como también con imágenes que tengan algún servicio como es Web, BD, etc...

Elijamos "***Cisco IOL***"



Se nos desplegará un formulario donde podemos ajustar o modificar según nuestra necesidad, tal es: nombre del dispositivo, icono a utilizar sea este un switch o router o subir una imagen del dispositivo para ser un poco mas creativo, también disponemos la opción de cambiar la RAM, pero esto se debe consultar cuanta RAM podríamos utilizar en un dispositivo especifico, esto para no gastar recursos exagerados en una imagen que al final no lo ocupara y podría ser utilizado para otra imagen que si lo requiera, sin embargo, cuando se selecciona una imagen PNetLab tratará de sugerir que cantidad de RAM necesita cada imagen. Sin duda es mejor consultar en los sitios de cada fabricante cuanto RAM utiliza, para aprovechar de mejor manera los recursos de la VM "***PnetLab***".

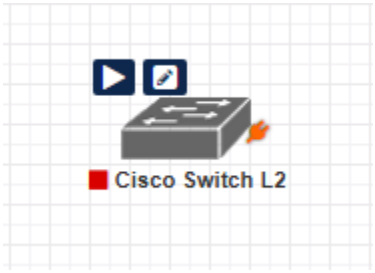


Seleccionaremos el de tipo L2 y cambiaremos el icono haciendo clic en la imagen que aparece un router y buscamos uno que sea switch. Luego cambiamos el tamaño de RAM a 256M, el nombre como SW01 y salvamos.

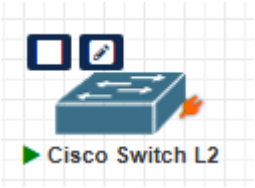
Buscamos un icono y lo seleccionamos.



Cuando salvamos en el formulario con todos los datos introducidos, aparece nuestro dispositivo en el área de trabajo:



Lo iniciamos, moviendo el puntero del Mouse sobre el icono del dispositivo y se visualizar unas imágenes pequeños, como en este caso una imagen del tipo "***Play***" y otro en forma de "***Lápiz***" pero es para modificar. Hacemos clic en "Play", vera que cambiara de estado, si es de color verde el "***Play***" estará listo para ingresar a la consola:



Inmediatamente podremos notar que si pasamos el puntero del Mouse en el icono, ya no aparece la imagen "***Play***", en cambio aparece como una especie de parada o "***Stop***". Procedemos ha hacer doble clic sobre el dispositivo y se nos abrirá una ventana auxiliar que será la consola o las consolas (consola o desktop) del dispositivo:

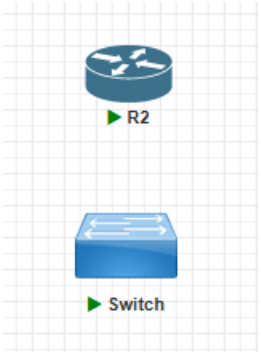


En mi caso agregue 2 dispositivos y descargue 2 imágenes:

```
root@pnetlab:/opt/unetlab/addons/qemu# ishare2 pull qemu 930 && ishare2 pull iol 21
[!] IMAGE INFO
 - Image Name       : viosl2-15.2.4.55e
 - Image Size       : 92.4 MiB
 - Image Type       : QEMU
 - Image ID         : 930
 - Image path       : /opt/unetlab/addons/qemu/viosl2-15.2.4.55e
 - Using host       : https://drive.labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/qemu/viosl2-1 100%[============================================================>]  92.38M  23.0MB/s    in 4.0s
[+] DOWNLOAD COMPLETED!
[-] Fixing permissions...

[+] Fix permissions command has been executed correctly
[!] IMAGE INFO
 - Image Name       : i86bi_linux_l2-ipbasek9-ms.may8-2013-team_track.bin
 - Image Size       : 58.9 MiB
 - Image Type       : IOL
 - Image ID         : 21
 - Image path       : /opt/unetlab/addons/iol/bin/
 - Using host       : https://labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/iol/bin//i86b 100%[============================================================>]  58.86M  7.94MB/s    in 6.3s
[+] Basic integrity checks passed. The downloaded image is the expected size.
[+] DOWNLOAD COMPLETED!
root@pnetlab:/opt/unetlab/addons/qemu#
```

Seguido de esto, actualice la topología o simplemente presione la tecla F5 para actualizar el navegador:



R2 × Switch ×

```
  Virtual-PPP         Virtual PPP interface
  Virtual-Template    Virtual Template interface
  Virtual-TokenRing   Virtual TokenRing
  vmi                 Virtual Multipoint Interface

Router#show run interface e
Router#show run interface ethernet
Router#show run interface ethernet ?
  <0-15>  Ethernet interface number

Router#show run interface ethernet 0
Router#show run interface ethernet 0
Router#show run interface ethernet 0/0
Building configuration...

Current configuration : 54 bytes
!
interface Ethernet0/0
 no ip address
 shutdown
end

Router#
```

R2 × Switch ×

```
* Technical Advisory Center. Any use or disclosure, in whole or in part, *
* of the IOSv Software or Documentation to any third party for any       *
* purposes is expressly prohibited except as otherwise authorized by     *
* Cisco in writing.                                                      *
**************************************************************************
Switch>
Switch>
*Dec 12 23:10:48.633: %PLATFORM-5-SIGNATURE_VERIFIED: Image 'flash0:/vios_l2-
isek9-m' passed code signing verification
Switch>
Switch>
Switch>
Switch>show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gi0/0, Gi0/1, Gi0/2, Gi0/3
                                                Gi1/0, Gi1/1, Gi1/2, Gi1/3
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
Switch>
```

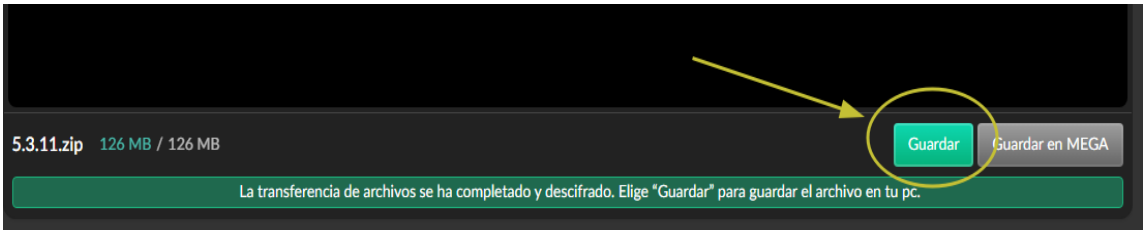
Actualización de PNetLab de la versión 4.2.10 a 5.3.11

Nos dirigimos al sitio siguiente para bajar la actualización: https://pnetlab.com/pages/releases

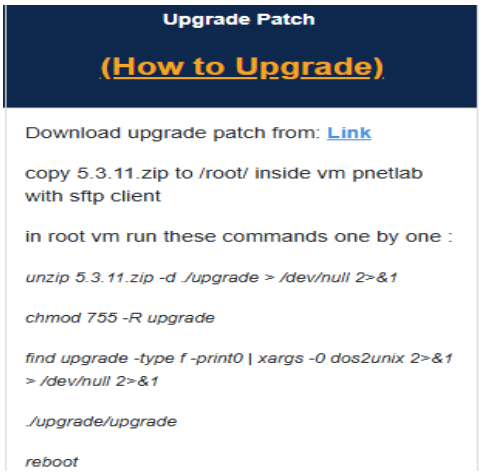


Hacemos clic en el hipervínculo “Link” para iniciar nuestra descarga, esto lo llevará a otra página, en este caso : https://mega.nz/file/6jIDhSbZ#rjwdd7vfZxrb-JmVSllSzye25FJYe9NfMg3B__UF964

Clic en el botón de “***Guardar***”



Una vez descargado el fichero en nuestro equipo, lo pasamos a nuestro servidor “PNetLab”, seguir la guía que aparece en el enlace de actualización de PnetLab:



```
C:\Users\PC\Downloads\Para PNetLab>scp 5.3.11.zip root@192.168.3.3:/root
root@192.168.3.3's password:
5.3.11.zip
```

Accedemos vía ssh al server “PNetLab”:

```
C:\>ssh root@192.168.3.3
root@192.168.3.3's password:
```

Luego nos vamos a la carpeta de /root:

```
root@pnetlab:/# cd /root
root@pnetlab:~#
```

Descomprimimos el fichero 5.3.11.zip según la guía, así:

```
root@pnetlab:~# unzip 5.3.11.zip -d ./upgrade > /dev/null 2>&1
```

Cambiamos permisos al directorio descomprimido:

```
root@pnetlab:~# chmod 755 -R upgrade
```

Ejecutamos esta línea de comandos:

```
root@pnetlab:~# find upgrade -type f -print0 | xargs -0 dos2unix 2>&1 > /dev/null 2>&1
root@pnetlab:~#
```

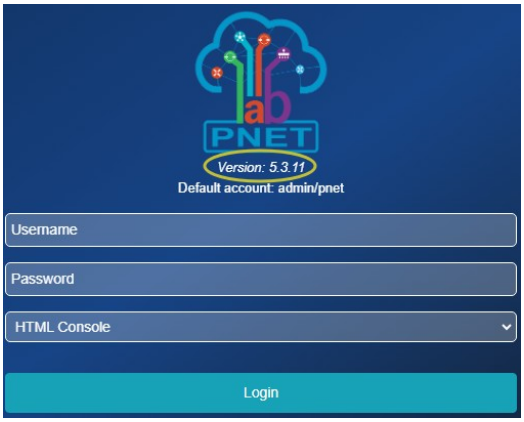
Con el siguiente linea de comandos, se actualizara y aplicará cambios en la versión de PnetLab:

```
root@pnetlab:~# ./upgrade/upgrade
 backup templates user for [4.2.10] [5.0.1] to /opt/unetlab/html/templates/backup_templates/backup_templates_user/
 backup pnetlab databases to /opt/unetlab/database_backup/
Hit:1 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Ign:4 https://pnetlab.labhub.eu.org ./ InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu bionic-security InRelease
Ign:6 https://pnetlab.labhub.eu.org ./ Release
Ign:7 https://pnetlab.labhub.eu.org ./ Packages
Ign:8 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:9 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:7 https://pnetlab.labhub.eu.org ./ Packages
Ign:8 https://pnetlab.labhub.eu.org ./ Translation-en
Ign:9 https://pnetlab.labhub.eu.org ./ Translation-en_US
Ign:7 https://pnetlab.labhub.eu.org ./ Packages
0% [Connecting to repo.pnetlab.com]
```

Si no presenta problemas al finalizar la actualización de paquetes para la nueva versión de PNetLab, procedemos a reiniciar nuestro servidor:

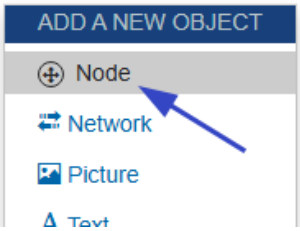
```
Upgraded successfully to 5.3.11
root@pnetlab:~# reboot
```

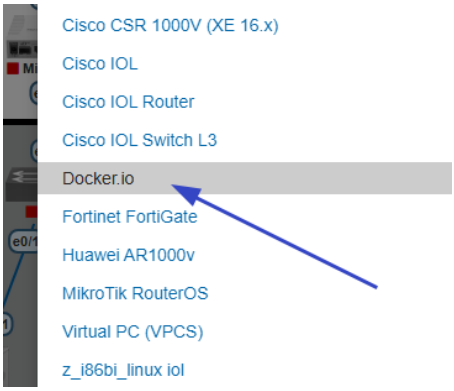
Podremos verificar nuestra nueva versión en el formulario de ingreso de credenciales para ingresar al Server PNetLab:

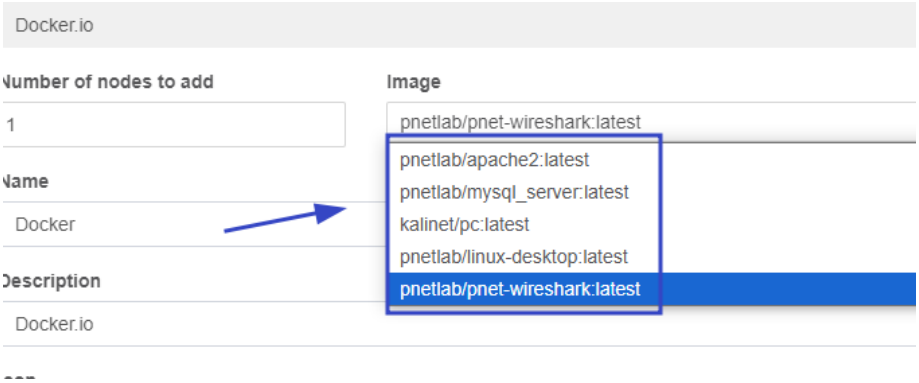


Imágenes Docker en PNetLab

Docker es un tipo de tecnología que vino a facilitar muchas tareas del día a día en centros de datos, como también desplegar de una manera transparente infraestructura a nivel de servicios, PNetLab viene incorporado con esta tecnología, cuando agregamos un dispositivo en nuestro laboratorio podemos agregar un router, switch de algun fabricante, pero también disponemos de la opción de agregar una instancia Docker (Servidor Web, Servidor BD, Desktop Windows, Desktop GNU/Linux, Appliance, etc.):







Y como cargo estas imágenes, donde las puedo descargar? Se mostrará lo básico para moverse en esta tecnología y ser capaz de integrar las instancias que se necesita, en caso que se desea realizar cosas mas avanzado debe de contar con una formación al respecto.

A continuación mostraremos unos comandos básicos que se debe de conocer para ir trabajando cómodo con Docker.

Para ver las instancias que están cargadas en Docker y están activas utilizaremos:

docker ps

```
root@pnetlab:/# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@pnetlab:/#
```

Para hacer una búsqueda de imágenes disponibles ejecutamos el siguiente comando:

docker search (alguna distribución)

```
root@pnetlab:/# docker search debian
NAME                      DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
debian                    Debian is a Linux distribution that's compos…   5123      [OK]
arm32v7/debian            Debian is a Linux distribution that's compos…   78
arm64v8/debian            Debian is a Linux distribution that's compos…   40
i386/debian               Debian is a Linux distribution that's compos…   17
amd64/debian              Debian is a Linux distribution that's compos…   15
debian/eol                End of Life Debian versions (pointing at arc…   13
s390x/debian              Debian is a Linux distribution that's compos…   6
smartentry/debian         debian with smartentry                          6                     [OK]
mips64le/debian           Debian is a Linux distribution that's compos…   5
debian/snapshot           "debian", but with sources.list pointing to …   5
arm32v5/debian            Debian is a Linux distribution that's compos…   4
ppc64le/debian            Debian is a Linux distribution that's compos…   4
dockette/debian           My Debian Sid | Jessie | Wheezy Base Images     2                     [OK]
debian/buildd             https://hub.docker.com/_/debian/ but --varia…   2
treehouses/debian                                                         2
tutum/debian              Simple Debian docker images with SSH access     2
corpusops/debian          debian corpusops baseimage                      0
ciready/debian            CI-ready Debian images with CI-related build…   0
debian/archvsync                                                          0
telkomindonesia/debian    All-in-One Debian (9.x) Base Image Repository   0
voxpupuli/debian                                                          0
jitesoft/debian           Debian base image.                              0
vptech/debian             Docker images of Debian.                        0
barebuild/debian                                                          0
vulhub/debian                                                             0
```

Para descargar una imagen, tomaremos como ejemplo debian:

docker pull debian

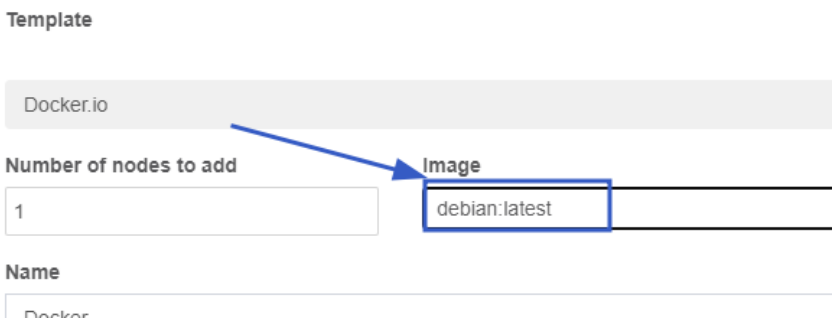
```
root@pnetlab:/# docker pull debian
Using default tag: latest
latest: Pulling from library/debian
fdf894e782a2: Pull complete
Digest: sha256:17122fe3d66916e55c0cbd5bbf54bb3f87b3582f4d86a755a0fd3498d360f91b
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
root@pnetlab:/#
```

Con el siguiente comando podremos verificamos la imagen descargada, el cual, fue “Debian” y ademas todas las imágenes que tengamos listo para utilizarse:

docker images

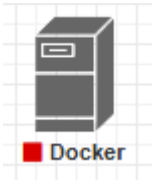
```
root@pnetlab:/# docker images
REPOSITORY                TAG       IMAGE ID       CREATED        SIZE
debian                    latest    ff869c3288a4   11 days ago    117MB
pnetlab/apache2           latest    5ead3e49a8e8   3 years ago    598MB
pnetlab/mysql_server      latest    a1c0c1b950c8   3 years ago    1.4GB
kalinet/pc                latest    88c06deddf31   3 years ago    3.65GB
pnetlab/linux-desktop     latest    de7a3304ba9d   4 years ago    2.43GB
pnetlab/pnet-wireshark    latest    7c4e9d9a8b94   4 years ago    996MB
root@pnetlab:/#
```

Verificamos si esta imagen de “Debian” este listo para utilizarse en PNetLab, para esto agregamos un nodo en el espacio de trabajo, seleccionamos “***Docker.io***”:

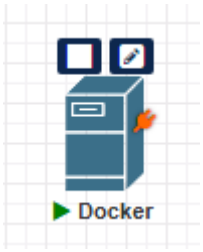


Y lo iniciamos, esperando que cambie su estado de “***Stop***” a “***Start***”:

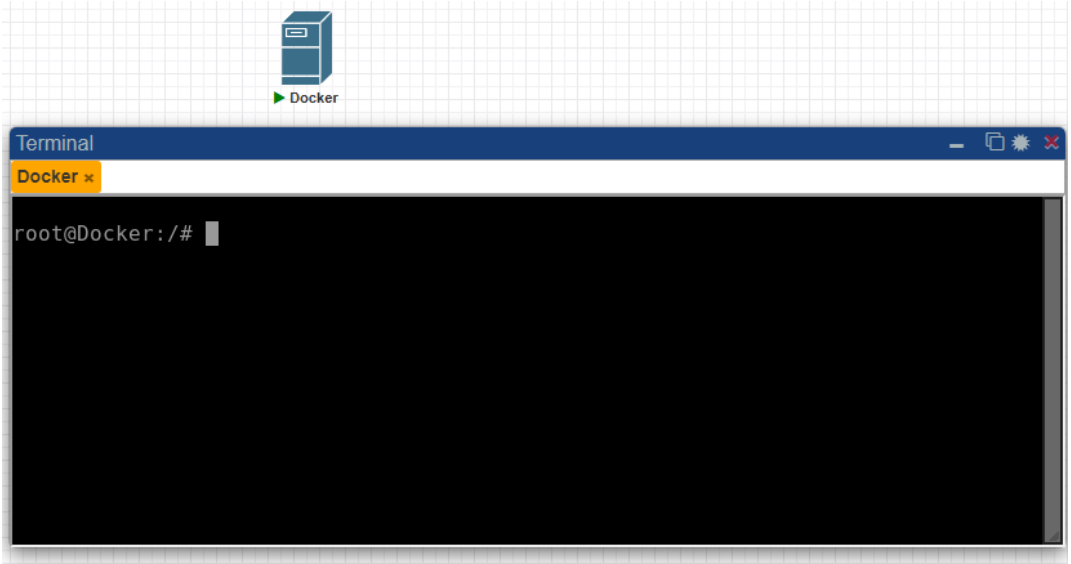
Primer estado “***Stop***”



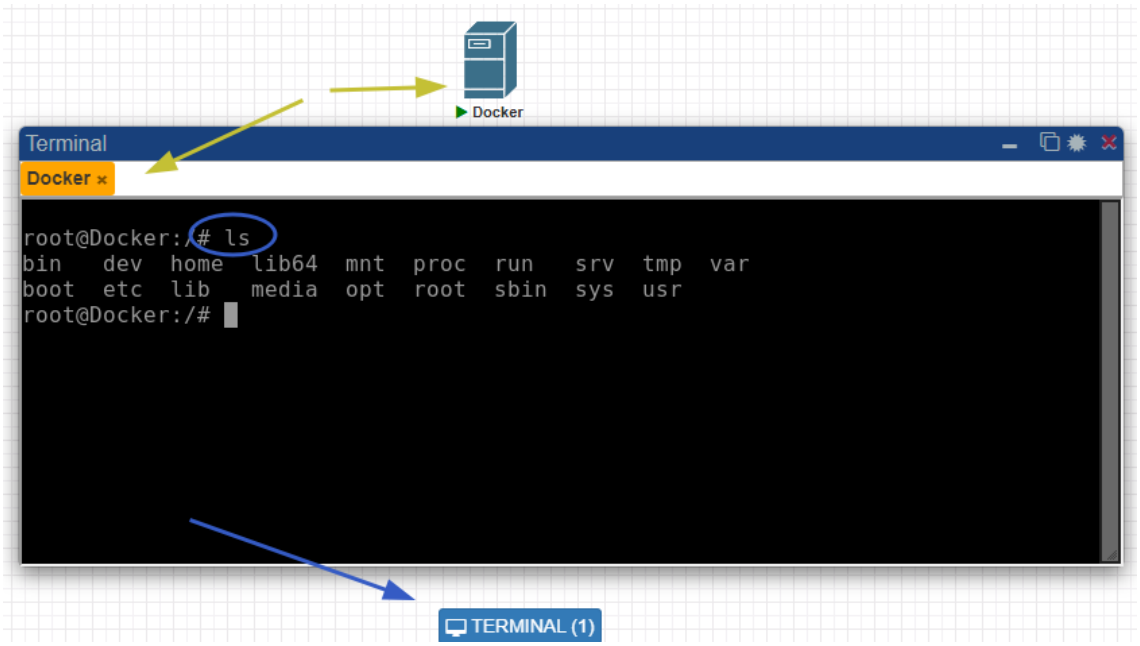
Estado iniciado “***Start***”.



Verificamos que este listo y operando, esto lo logramos haciendo doble clic sobre el icono:



Verificamos algunos comandos, como listar directorios, recuerde que estas imágenes son pequeñas en tamaño, por tanto ligeras y puede que no disponga de muchas aplicaciones, sea el caso que si desea descargar alguna imagen que tenga algún servicio corriendo como es apache, o una imagen que tenga un escritorio. Observe que el icono del Server Debian tiene un nombre descriptivo llamado Docker, esto lo cambiamos al momento de agregar el nodo, ademas miramos que trabaja sin problemas al ejecutar un comando básico que tiene los GNU/Linux, como es el de listar ficheros y directorios, ademas en la parte inferior, en caso de que cierre la ventana de terminal, podrá abrirla dando doble clic en el icono una vez mas o hacer clic en el botón "***TERMINAL (1)***"

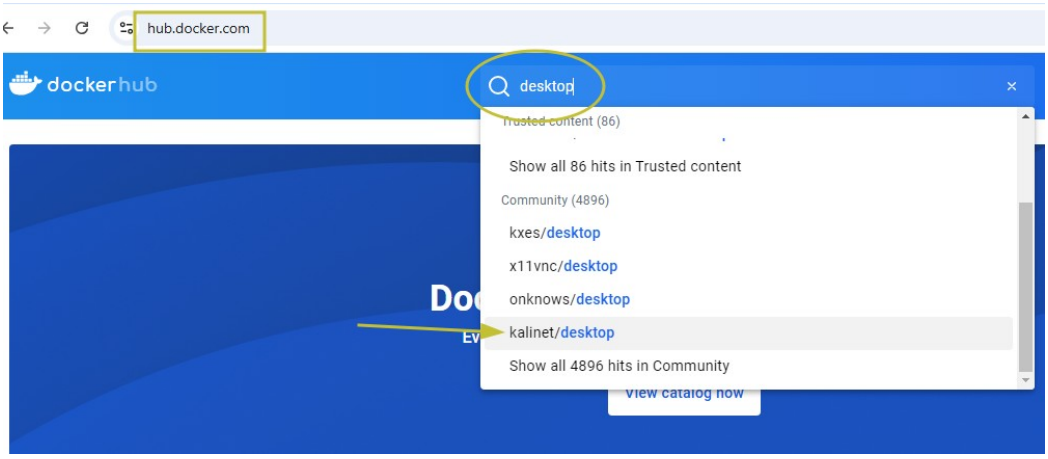


Seguimos con la verificación de esta imagen, si que esta cargada o no, pero, esta vez vía SHELL, desde el servidor de PNetLab, ejecutamos el siguiente comando de Docker:

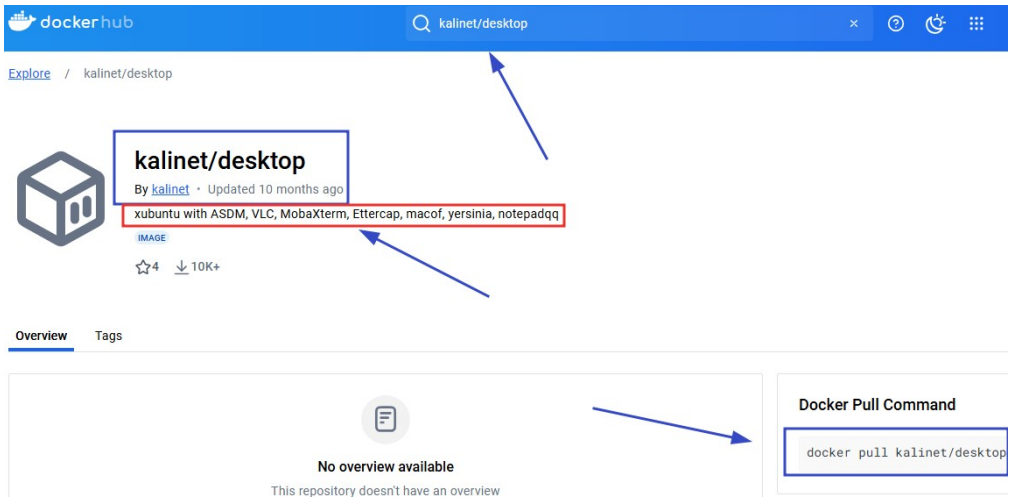
docker ps

```
CONTAINER ID   IMAGE           COMMAND   CREATED          STATUS          PORTS     NAMES
df3168fde85f   debian:latest   "bash"    11 minutes ago   Up 8 minutes              docker1
root@pnetlab:~# _
```

Si deseamos buscar una instancia con escritorio por ejemplo, ubuntu desktop o de centos, podemos consultar el sitio de imagenes de docker en Docker Hub, su sitio es https://hub.docker.com , por ejemplo seleccionamos "***kalinet/desktop***":

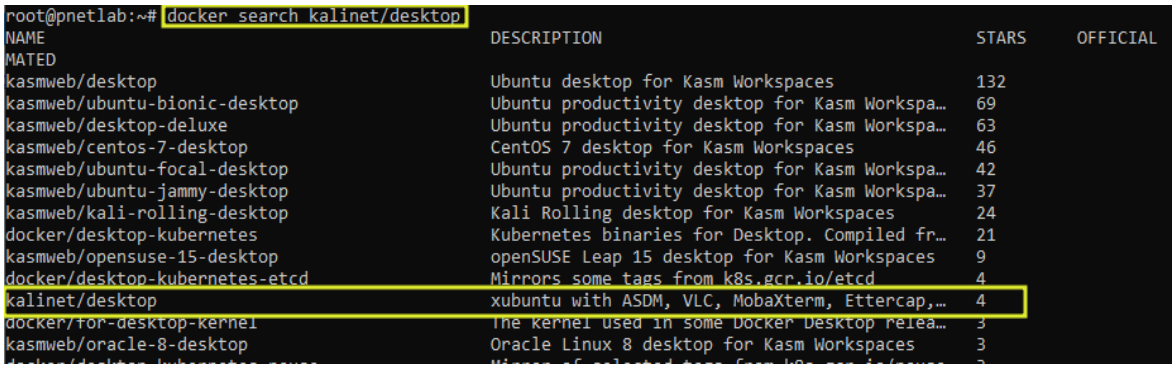


Un poco de información de proyecto de Kalinet, la imagen esta basado en "***Xubuntu***" y ademas tiene herramientas como ASDM, VLC, Etthercap, etc...

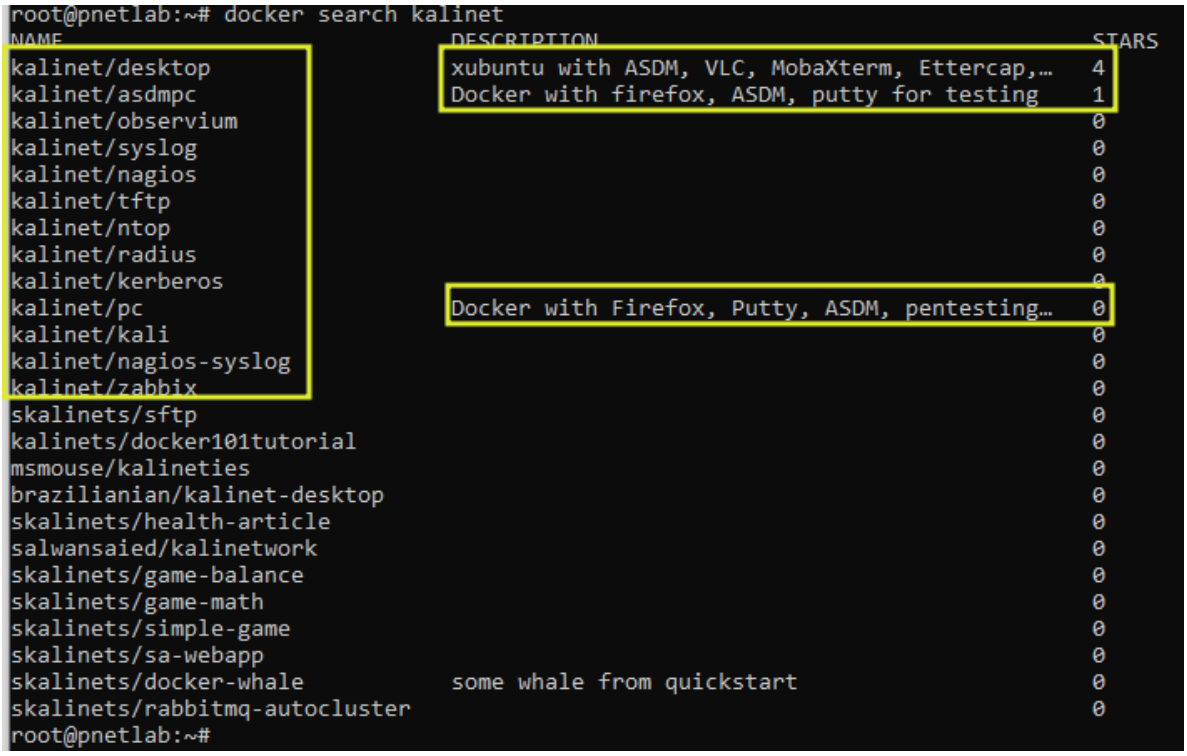


Vía SHELL podemos buscar esa imagen, del proyecto de Kalinet o todas las preparadas por el mismo:

docker search kalinet/desktop

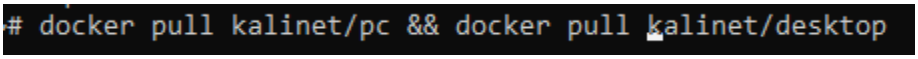


docker search kalinet



Descargamos kalinet/pc y kalinet/desktop y luego comprobamos via SHELL y PNetLab que ya lo tengamos integrado:

docker pull kalinet/pc && docker pull kalinet/desktop



```
root@pnetlab:~# docker pull kalinet/pc && docker pull kalinet/desktop
Using default tag: latest
latest: Pulling from kalinet/pc
Digest: sha256:5c7b71173f44e0d653bbafde6f1cfaa505ddd373c3faba35e4e6e1d1784bda41
Status: Image is up to date for kalinet/pc:latest
docker.io/kalinet/pc:latest
```

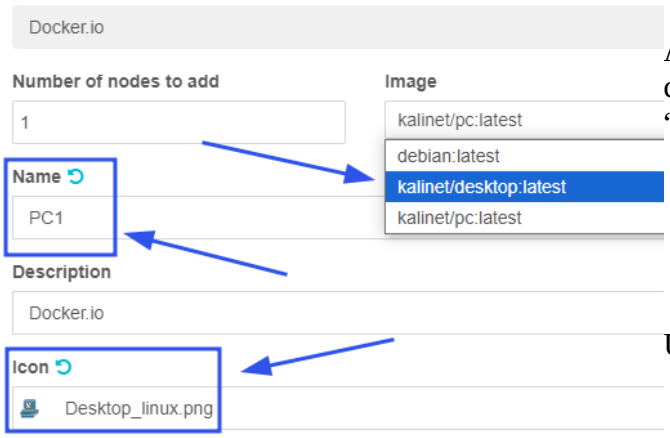
```
Using default tag: latest
latest: Pulling from kalinet/desktop
4451f5c7eb7a: Pull complete
256df2f17882: Pull complete
7b90b0c5270a: Pull complete
23e5806411f7:
2a67a1088f39: Download complete
```

Verificamos lo descargado:

docker images

```
root@pnetlab:~# docker images
REPOSITORY        TAG       IMAGE ID       CREATED          SIZE
debian            latest    ff869c3288a4   11 days ago      117MB
kalinet/desktop   latest    d2e1a6defe37   22 months ago    6.58GB
kalinet/pc        latest    88c06deddf31   3 years ago      3.65GB
root@pnetlab:~#
```

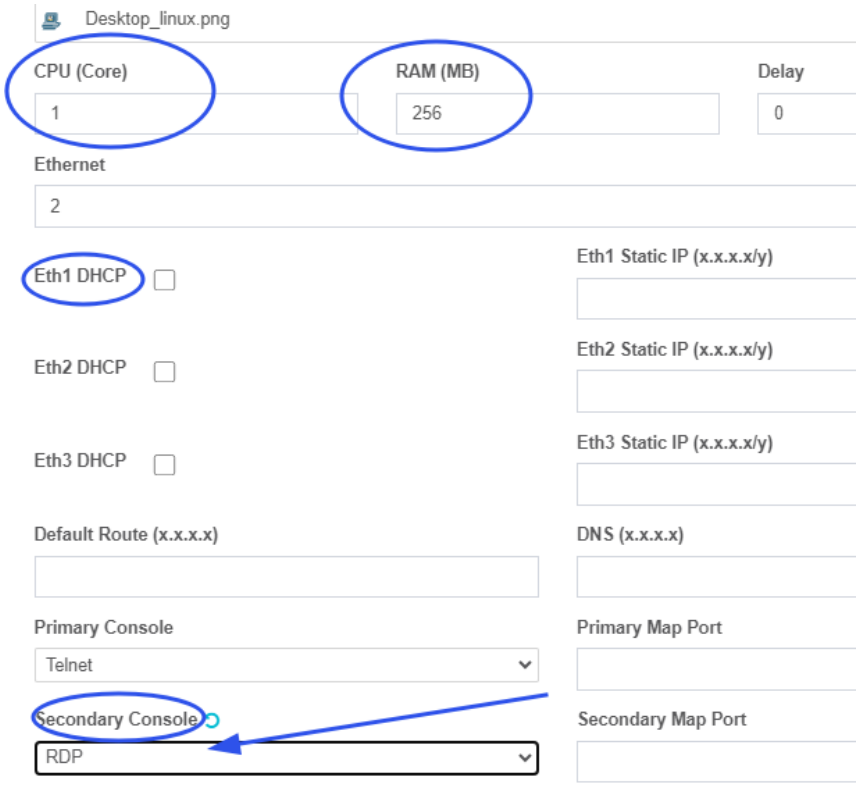
Toca verlo en PNetLab, vamos agregando nodos y elegimos docker, luego las imágenes:



Aquí están las imágenes a utilizar y de paso cambiamos el nombre en la casilla "***Name***" como "PC1" y el "***icono***":

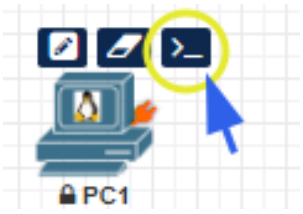
Usaremos la imagen "***Kalinet/Desktop:latest***"

Prestar suma atención con la opción en la casilla "***Secundary Console***", aquí le indicamos que abra una terminal con escritorio. Aprovechando podemos cambiar parámetros de CPU (cuantos core), cantidad de RAM, que la tarjeta de red "***Eth1 DHCP***" quede habilitado cuando activemos un Router o Switch como servidor DHCP, y demás:

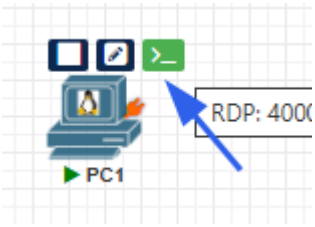


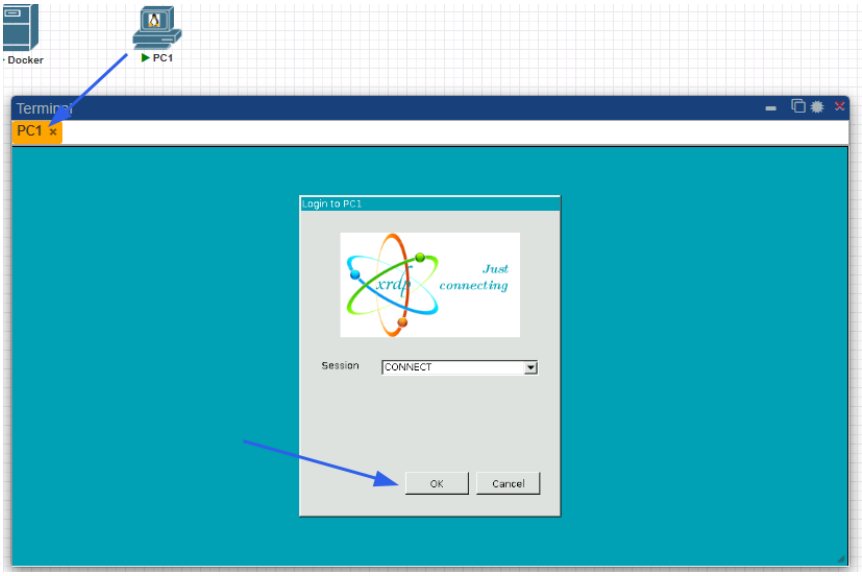
Arrancamos la máquina una vez agregado el nodo y podremos observar que hay una pequeña imagen nueva cuando pasamos sobre el icono:

Esto sería para abrir una consola con escritorio, esto es valido activarlo para las imágenes que tengan instalado el escritorio, esto es valido también para imágenes Windows.

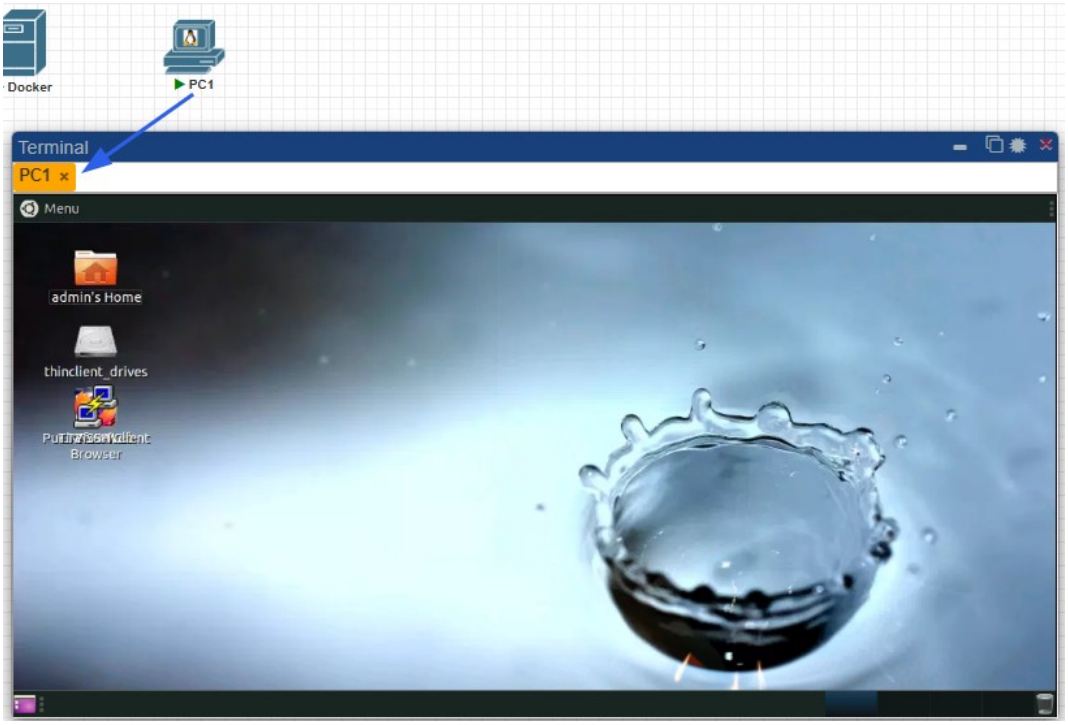


Una vez arrancado, hacemos clic en la segunda imagen con símbolo de "***prompt***", esto abrirá la consola en modo desktop para esa instancia:

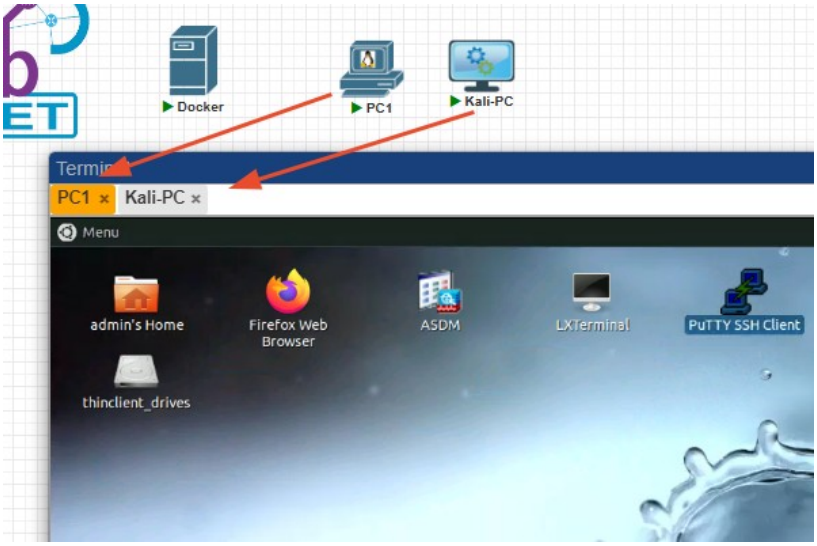




La ventana la podremos dimensionar:



Ya tenemos imágenes de equipos desktop, ya disponemos con imágenes de Router y Switch de Cisco, a continuación podremos ver ambas instancias de Docker corriendo sin problemas (Recuerde que esta en dependencia de las capacidades de Hardware que se tiene en el servidor para realizar esta tarea):



Imágenes que necesitamos en una infraestructura de red con varios elementos a contar, Routers, Switches L2 y L3, Appliance FortiGate 7.x, Mikrotik 7.x(mostraremos la herramienta a utilizar Winbox), equipos Desktop y Servers basados en GNU/Linux, equipos Desktop y Servidor basado en Windows (mostraremos su descarga, para esto usaremos ishare2).

Descargaremos las imagenes de ***FortiGate 7.x***, ***Mikrotik 7.x***, usaremos ishare2:

Realizaremos una búsqueda de FortiGate 7.x: →

```
root@pnetlab:~# ishare2 search fortinet
```

Utilizaremos esta imagen: →

```
375  fortinet-FGT-v7.0.1-build0157      72.4 MiB
376  fortinet-FGT-v7.0.10.M-build0450   71.5 MiB
377  fortinet-FGT-v7.0.11.M-build0489   71.6 MiB
```

Esta imagen es del tipo qemu, lo descargamos haciendo un pull con ishare2:

```
root@pnetlab:~# ishare2 pull qemu 375
```

Verifiquemos que no haya un error de descarga:

```
root@pnetlab:~# ishare2 pull qemu 375
[!] IMAGE INFO
 - Image Name        : fortinet-FGT-v7.0.1-build0157
 - Image Size        : 72.4 MiB
 - Image Type        : QEMU
 - Image ID          : 375
 - Image path        : /opt/unetlab/addons/qemu/fortinet-FGT-v7.0.1-build0157
 - Using host        : https://drive.labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/qemu/forti 100%[=================================================>]  72.38M
[+] DOWNLOAD COMPLETED!
[-] Extracting: fortinet-FGT-v7.0.1-build0157.tgz file...
[+] Extracted: /opt/unetlab/addons/qemu/fortinet-FGT-v7.0.1-build0157. Image ready to use.
[-] Fixing permissions...

[+] Fix permissions command has been executed correctly
root@pnetlab:~#
```

Buscamos imágenes de Mikrotik 7.x

```
658  mikrotik-7.1      35.0 MiB
659  mikrotik-7.1.3    36.3 MiB
660  mikrotik-7.2.1    36.6 MiB
661  mikrotik-7.2.3    36.8 MiB
662  mikrotik-7.5      37.2 MiB
663  mikrotik-7.6      39.1 MiB

29 QEMU images found for the term: "mikrotik"
```

Lo descargamos haciendo un pull, siempre verificamos al final de la búsqueda que tipo de imagen es, en este caso es del tipo qemu:

```
root@pnetlab:~# ishare2 pull qemu 658
[!] IMAGE INFO
 - Image Name        : mikrotik-7.1
 - Image Size        : 35.0 MiB
 - Image Type        : QEMU
 - Image ID          : 658
 - Image path        : /opt/unetlab/addons/qemu/mikrotik-7.1
 - Using host        : https://labhub.eu.org
[!] DOWNLOADING IMAGE
/opt/unetlab/addons/qemu/mikro 100%[=================================================>]  35.03M
[+] DOWNLOAD COMPLETED!
[-] Extracting: mikrotik-7.1.tgz file...
[+] Extracted: /opt/unetlab/addons/qemu/mikrotik-7.1. Image ready to use.
[-] Fixing permissions...

[+] Fix permissions command has been executed correctly
root@pnetlab:~#
```

Descargaremos Desktop del tipo windows, para esto hacemos una búsqueda con la palabra “win”, así nos mostrará todas las imágenes disponibles:

```
root@pnetlab:~# ishare2 search win
```

La lista es grande y al final verán que son del tipo qemu, descargaremos como cliente el ID 1199 Windows10-x64 lite:

```
1198  win-10-x64-VL19                    6.7 GiB
1199  win-10-x64-lite                    7.8 GiB
1200  win-10-x86-20H2v3                  5.1 GiB
1201  win-10-x86-21H1v1                  4.5 GiB
1202  win-10ENT                          8.8 GiB
1203  win-10pro64-20H2                   9.4 GiB
1204  win-11-x64-DEV                     5.8 GiB
1205  win-11-x64-SE                      5.8 GiB
1206  win-7-Ent-main                     2.3 GiB
1207  win-7-lite-u4                      2.8 GiB
```

Para servidores del tipo Windows Server, bajaremos 2019 y 2019 ID 1225 y 1226:

```
1218  winserver-2008-lite-u1             2.7 GiB
1219  winserver-2012R2                   3.8 GiB
1220  winserver-2019                     7.6 GiB
1221  winserver-S2008-R2-x64             3.8 GiB
1222  winserver-S2012-R2-x64             4.5 GiB
1223  winserver-S2012-R2-x64             4.4 GiB
1224  winserver-S2012-R2-x64-rev2        4.3 GiB
1225  winserver-S2016-R2-x64             5.3 GiB
1226  winserver-S2019-R2-x64-rev3        6.0 GiB
```

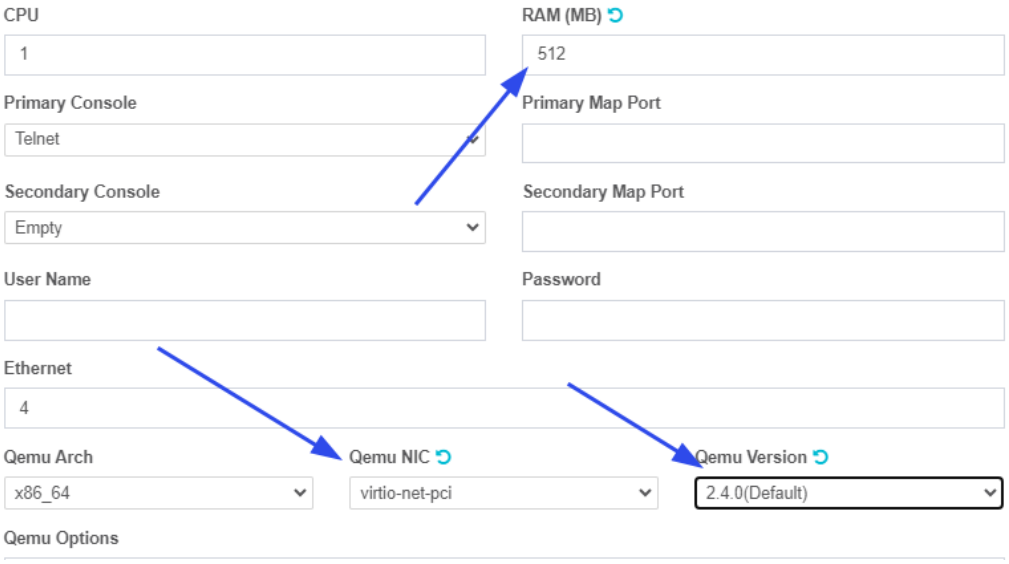
Hacemos un “pull” a las marcadas en amarillo, en una sola linea, esto tardara mucho y dependerá de la conexión a internet que tenga:

```
# ishare2 pull qemu 1199 && ishare2 pull qemu 1225 && ishare2 pull qemu 1226
```

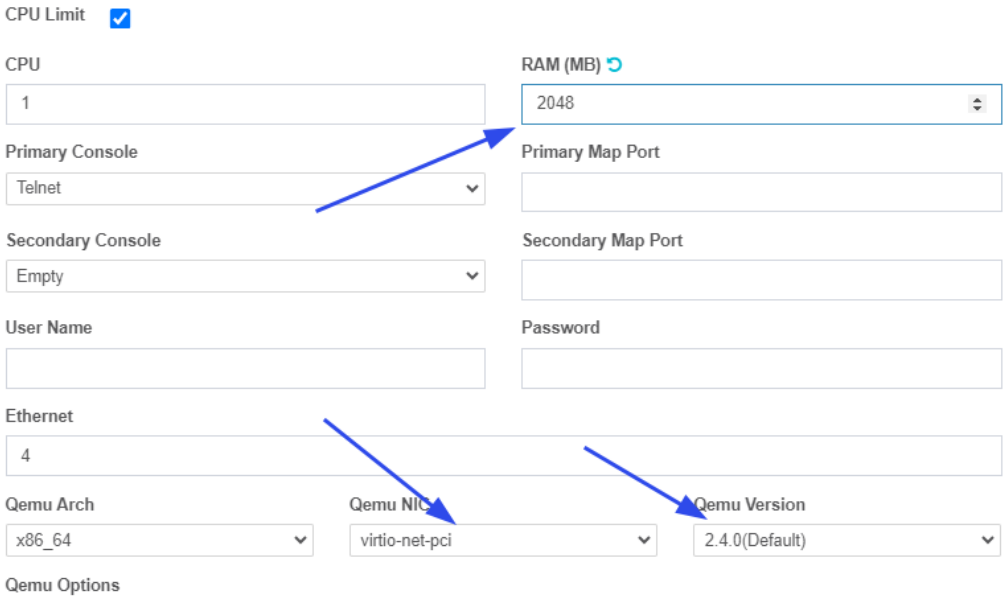
Una vez realizado la descarga de las imágenes que necesitaremos, será necesario prestar mucha atención a los recursos que deben tener los equipos Desktop, en este caso los de tipo Windows, trataremos de ajustarlos para que tengan lo necesario en RAM y CPU, de la misma manera, para las imágenes de los Firewall de FortiGate y Mikrotik, estas imágenes tienen limite por su licencia, por tanto no podremos subir mas de 2G de RAM y no mas de 1CPU.

A continuación unos parámetros importantes para las imágenes de equipos FortiGate y Mikrotik, recordando que podemos subir una imagen que represente el fabricante como el nombre descriptivo.

Configuración recomendada para equipo Mikrotik:



Configuración recomendada para equipo FortiGate:

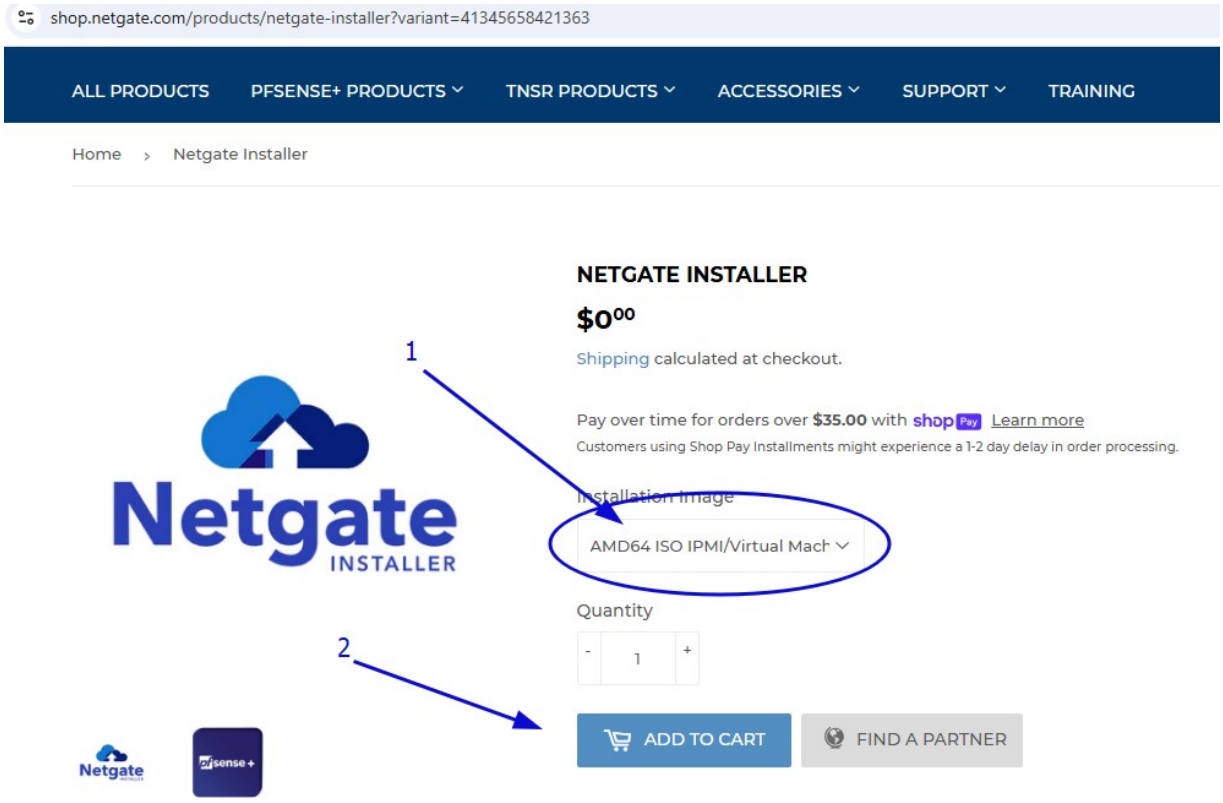


Creando imágenes personalizada de OPNsense y Pfsense en PNetLab

Debemos descargar las imágenes de tipo ISO de ambos software:

Para el caso de Pfsense, el fichero debe ser del tipo "***AMD64 ISO IPMI/Virtual Machines***"

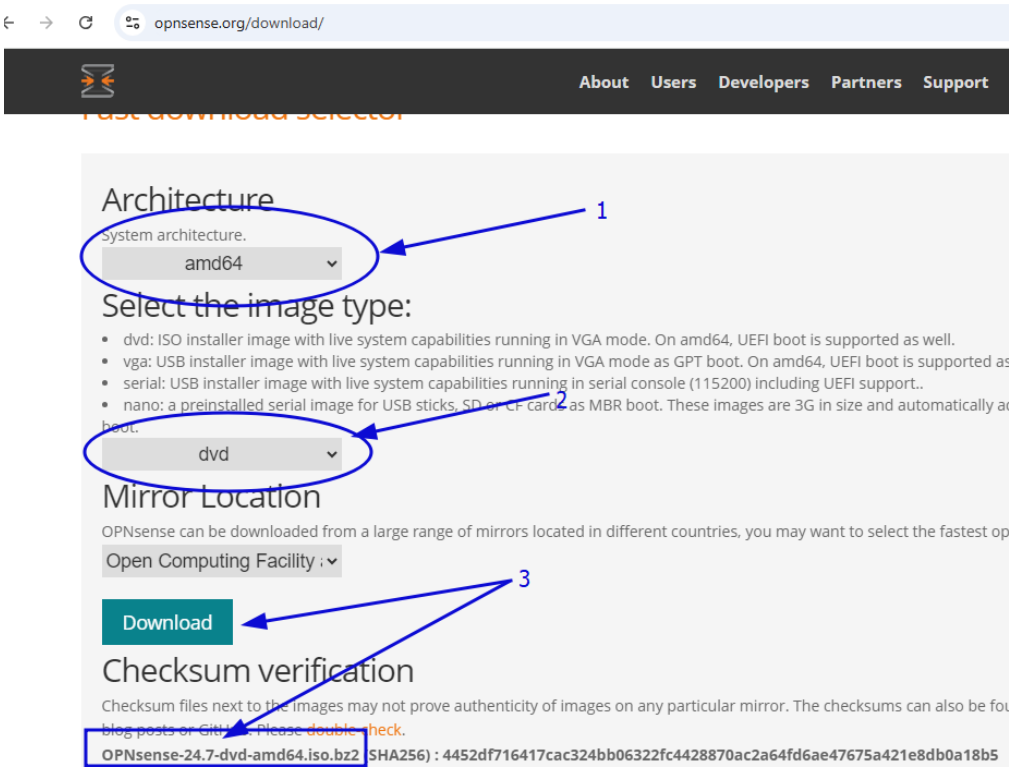
https://shop.netgate.com/products/netgate-installer



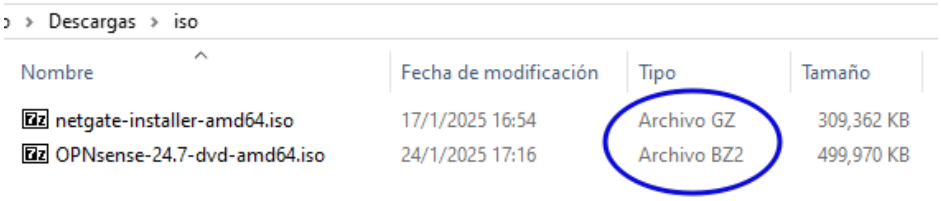
Sigan los pasos del sitio web para descargarlo.

Para OPNsense, nos vamos al sitio web oficial: https://opnsense.org/download/

Seleccione según los pasos:



Ambos software estarán comprimidos, por tanto, debemos descomprimirlo para luego copiarlo al PNetLab:

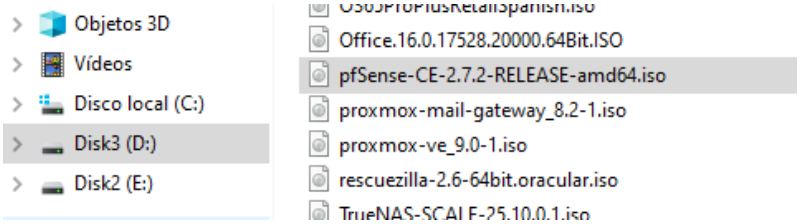


Observará que netgate (pfsense) el formato comprimido es “gz” y el de OPNsense es “bz2”, ambos podrán descomprimirse con alguna herramienta que soporte estas extensiones, en mi caso tengo instalado 7zip, mas información: https://7-zip.org/download.html

Se procede a descomprimir ambos ficheros, el cual da como resultado ficheros con extensión “ISO”, estos ficheros hay que trasladarlos en carpetas con el nombre y versión por tema de identificar más cómodo cada solución, estas carpetas debe crearse en el servidor de PNetLab.

Para este ejemplo realizare la creación de la imagen de Pfsense2.7.2, sin embargo el proceso sería igual con Opnsensex.x.

Una vez descargado y descomprimido nuestra imagen de Pfsense, copiaremos esta imagen a nuestro servidor PNetLab:



Abrimos nuestro “cmd” y nos vamos hasta el directorio de nuestra imagen:

```
C:\Users\PC>d:

D:\>cd iso

D:\iso>dir
 El volumen de la unidad D es Disk3
 El número de serie del volumen es: 262C-ECF0

 Directorio de D:\iso

21/02/2026  15:06    <DIR>          .
21/02/2026  15:06    <DIR>          ..
21/08/2025  12:28       790,626,304 debian-13.0.0-amd64-netinst.iso
03/06/2025  08:43     5,563,336,704 es-es_windows_server_2022_x64_dvd_c25dea55.iso
13/02/2026  11:53     4,768,530,432 es-mx_windows_10_enterprise_ltsc_2021_x64_dvd_f6aaf3
31/01/2025  12:25     3,291,686,912 HBCD_PE_x64.iso
23/07/2025  11:35     2,258,632,704 issabel5-USB-DVD-x86_64-20240430.iso
23/04/2025  14:23     4,439,457,792 kali-linux-2025.1a-installer-amd64.iso
23/04/2025  14:23     4,551,966,720 kali-linux-2025.1a-installer-purple-amd64.iso
10/11/2023  14:28     7,144,996,864 Microsoft Office 2024 v2312 Build 17103.20000.iso
20/01/2025  14:00        68,904,830 Microsoft Toolkit.7z
19/09/2024  10:02     1,247,728,128 netgate-installer-v1.0-RC-amd64-20240919-1435.img
19/09/2024  10:02       989,732,864 netgate-installer-v1.0-RC-amd64-20240919-1435.iso
21/02/2026  15:06    <DIR>          netgate-installer-v1.1.1-RELEASE-2.8-ISO
21/02/2026  15:04    <DIR>          netgate-installer-v1.1.1-RELEASE-img
07/09/2024  08:35     4,911,529,984 O365ProPlusRetailSpanish.iso
31/03/2024  07:52     7,906,344,960 Office.16.0.17528.20000.64Bit.ISO
17/10/2025  16:45       874,672,128 pfSense-CE-2.7.2-RELEASE-amd64.iso
23/04/2025  14:27     1,579,073,536 proxmox-mail-gateway_8.2-1.iso
```

Procedemos a copiar esta imagen al servidor, para esto necesitamos conocer la ip del servidor y el directorio donde estará dicha imagen de disco instalador, este directorio debe representar el nombre de nuestra maquina, en este caso, pfsense2.7.2. Nos conectamos al servidor PNetLab:

```
C:\Users\PC>ssh root@192.168.3.3
root@192.168.3.3's password:
```

nos dirigimos hasta la ruta donde debe crearse la carpeta para esta imagen instalador y maquina qemu para pfsense2.7.2:

```
root@pnetlab:/# cd opt/unetlab/addons/qemu/
root@pnetlab:/opt/unetlab/addons/qemu#
```

Aquí crearemos la carpeta llamada pfsense2.7.2:

```
root@pnetlab:/opt/unetlab/addons/qemu# mkdir pfsense-CE-RELEASE-2.7.2/
```

Ahora desde Windows copiamos el fichero instalador de pfsense-CE-2.7.2-RELEASE-amd64.iso, en el directorio que se creo "/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2" a partir del comando "cmd", de la siguiente forma:

```
C:\Users\PC>d:

D:\>cd iso

D:\iso>scp pfSense-CE-2.7.2-RELEASE-amd64.iso root@192.168.3.3:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2
root@192.168.3.3's password:
pfSense-CE-2.7.2-RELEASE-amd64.iso                                          0%    0     0.0KB/s   --:-- ETA
```

Regresamos al server, ingresaremos al directorio pfsense-CE-RELEASE-2.7.2 y realizaremos unos ajustes:

cambiamos el nombre del fichero pfsense-CE-2.7.2-RELEASE-amd64.iso a pfsense2.7.2, esto por comodidad, luego cambiamos el nombre a cdrom.iso:

```
root@pnetlab:/opt/unetlab/addons/qemu# cd pfsense-CE-RELEASE-2.7.2/
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# ls
pfSense-CE-2.7.2-RELEASE-amd64.iso
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# mv pfSense-CE-2.7.2-RELEASE-amd64.iso cdrom.iso
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2#
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# ls
cdrom.iso
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2#
```

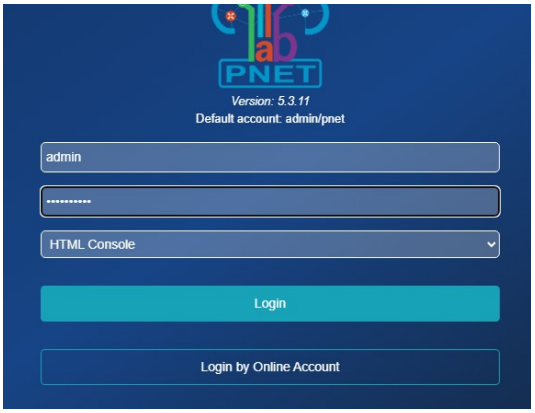
Crearemos el disco virtual que tendrá nuestro pfsense, el cual, estará instalado, con un tamaño de 15G:

```
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# qemu-img create -f qcow2 virtioa.qcow2 15G
Formatting 'virtioa.qcow2', fmt=qcow2 size=16106127360 cluster_size=65536 lazy_refcounts=off refcount_bits=16
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2#
```

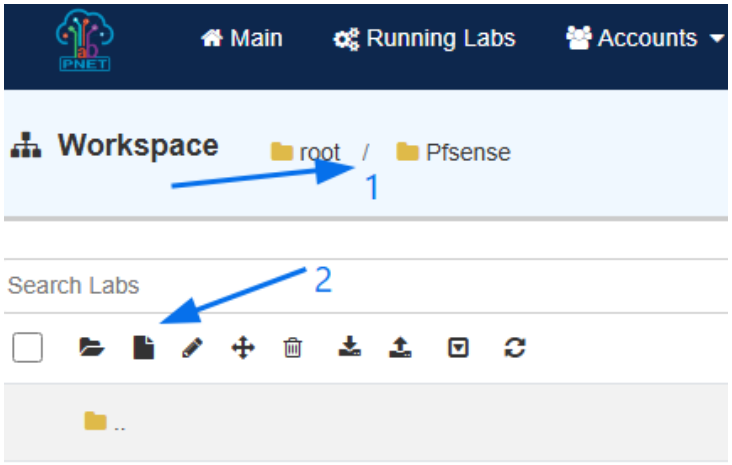
Como último paso, subimos de nivel de directorio y aplicamos permisos necesarios para indicarle a PNetLab que pueda hacer cambios:

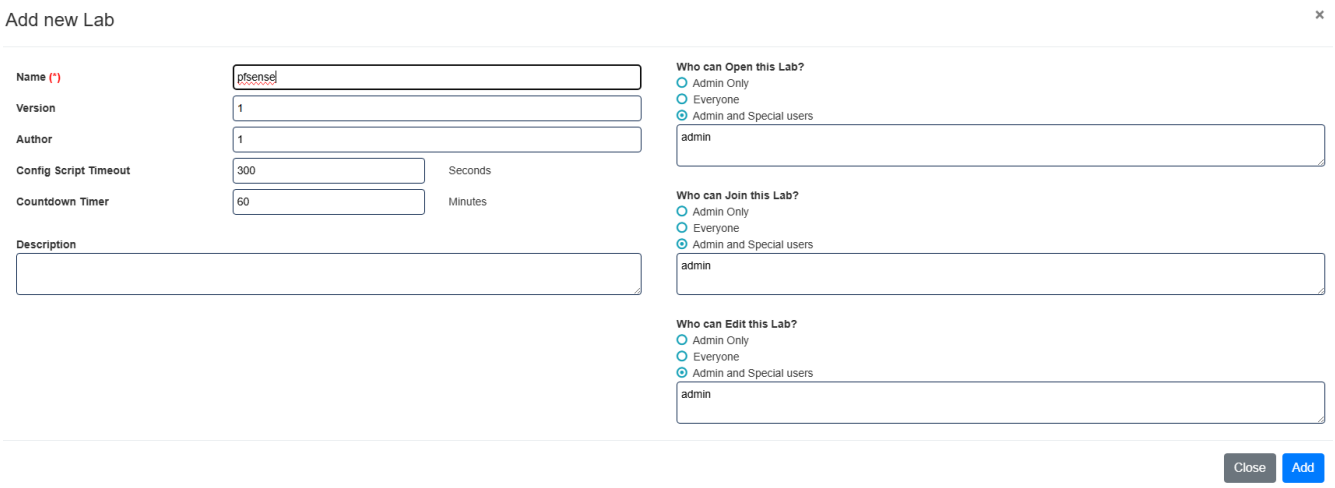
```
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2#
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# cd ..
root@pnetlab:/opt/unetlab/addons/qemu# /opt/unetlab/wrappers/unl_wrapper -a fixpermissions
```

Toca ir al entorno de PNetLab para crear nuestro lab con pfsense2.7.2:

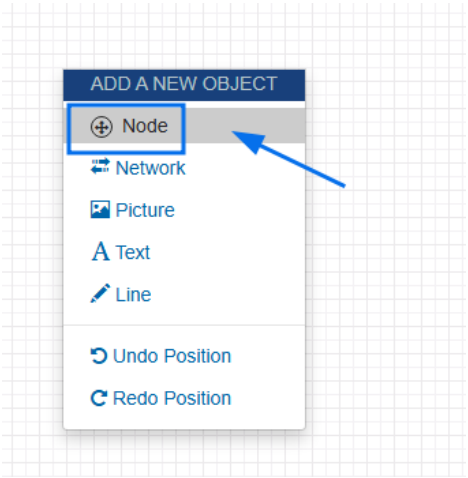


Creamos un lab para pfsense:

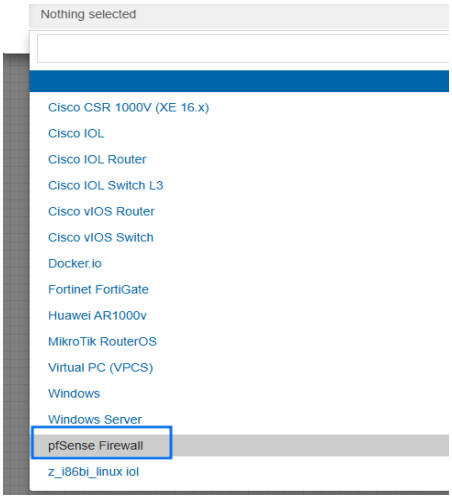


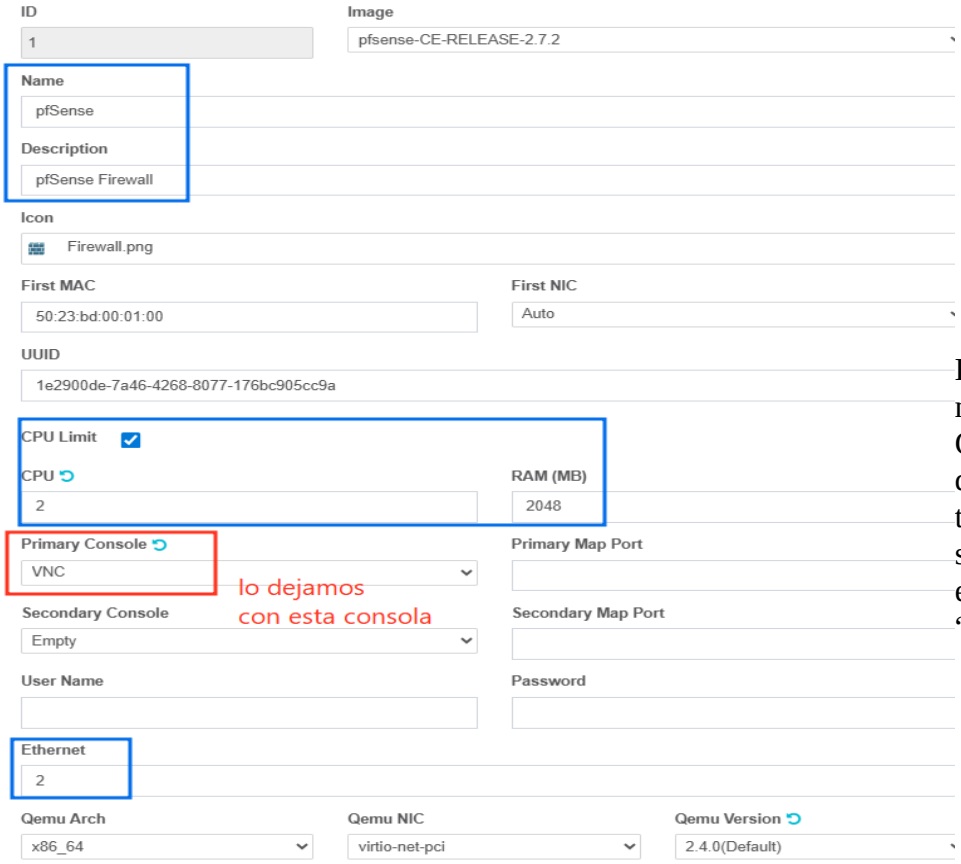


Una vez creado, agregamos un nodo:



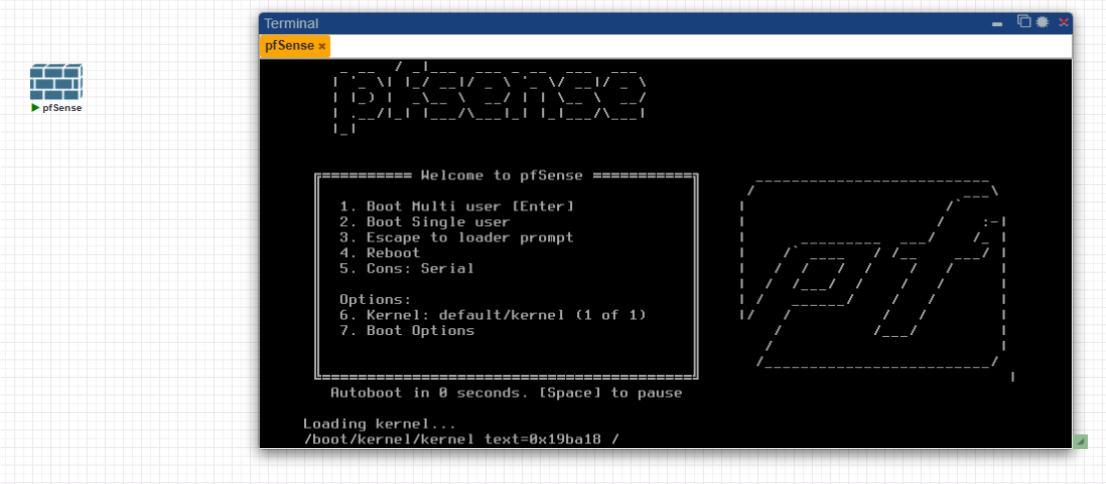
Buscamos nuestro pfsense:



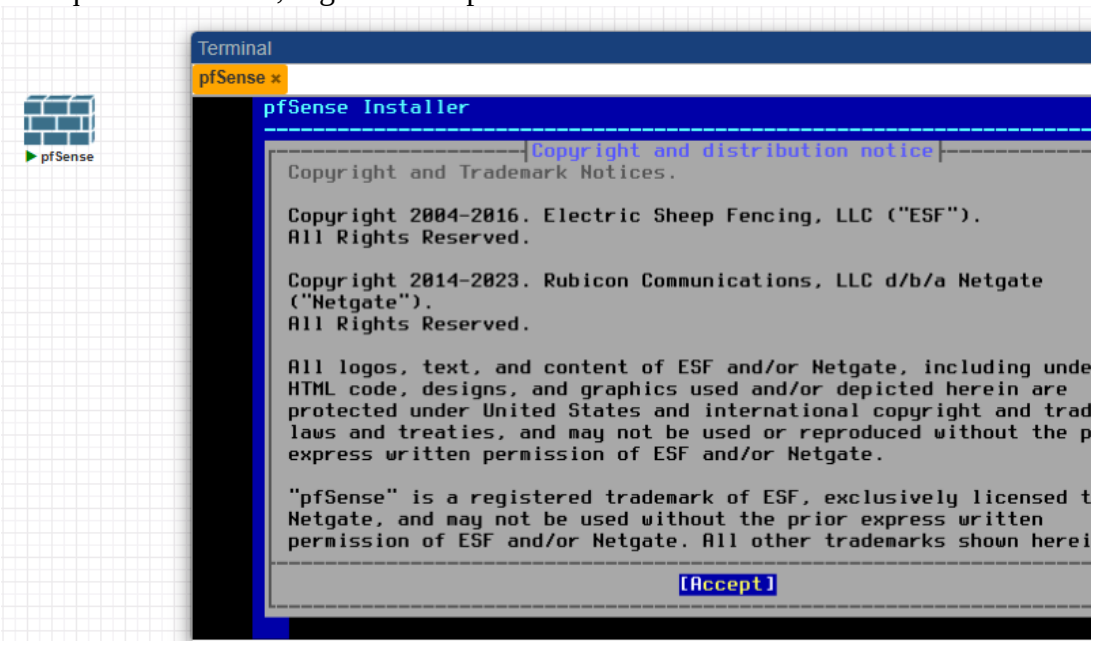


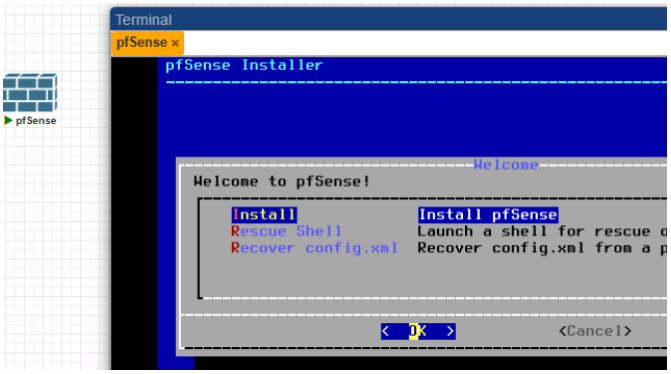
En este punto, podemos modificar la cantidad de CPU, RAM, nombre, descripción y cantidad de tarjetas de red, lo que si o si, es modificar la consola, en este caso debe ser “VNC”:

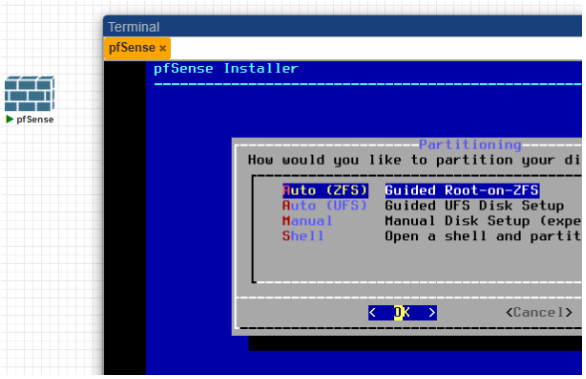
Iniciamos la maquina pfsense:

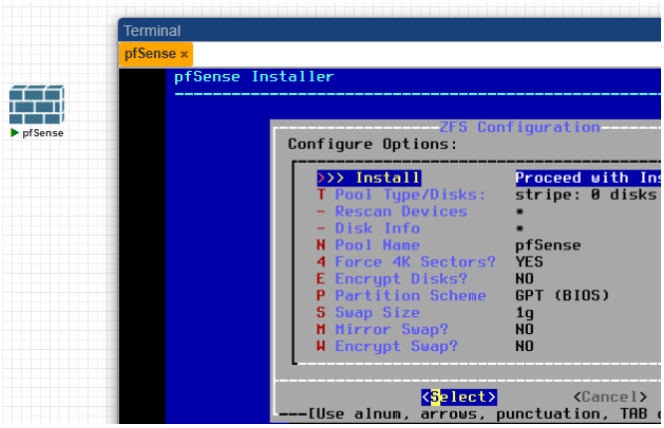


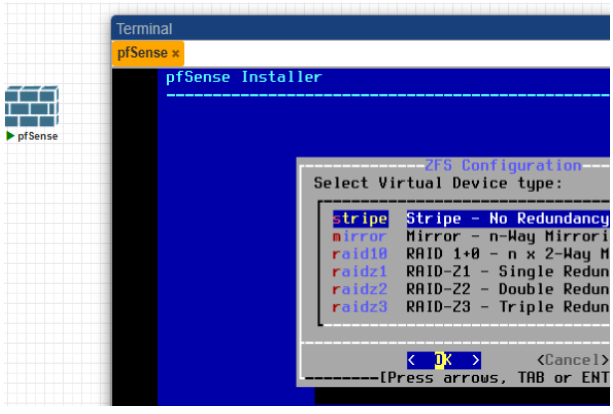
Típico de cualquier instalación, seguimos los pasos:

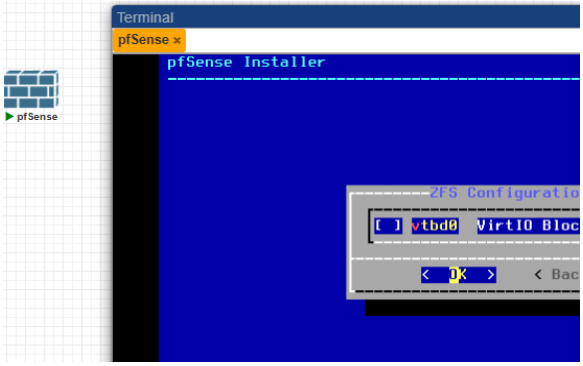


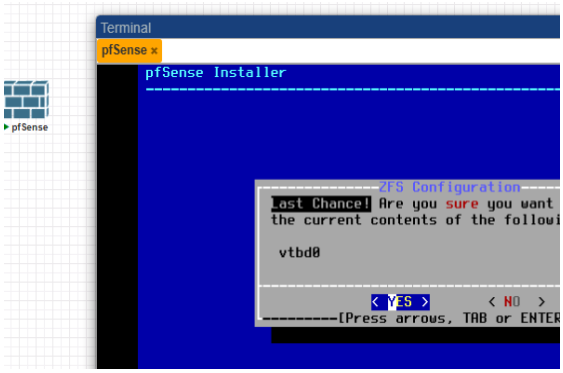


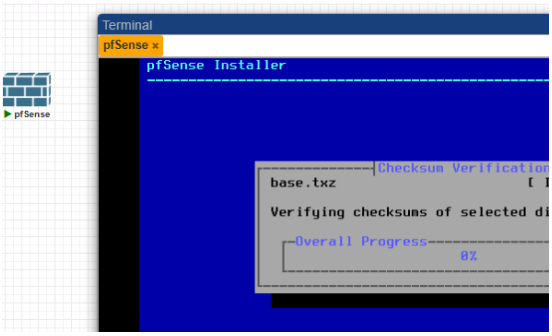


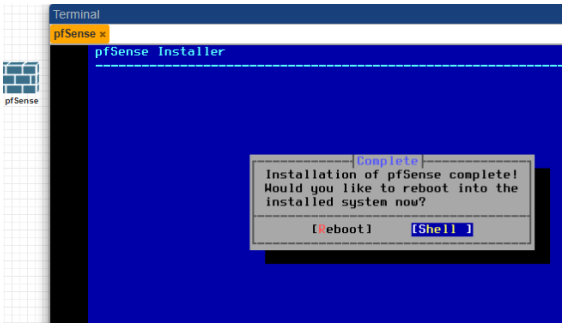












Seleccione la “**Shell**” en el asistente de instalación, luego ejecutamos el comando “**poweroff**”, esto apagara la maquina pfsense.

Acto seguido nos vamos en la CLI de PNetLab y nos dirigimos al directorio siguiente:

```
root@pnetlab:/opt/unetlab/tmp/1
```

En este directorio, se guarda temporalmente la imagen modificada, ya con datos de instalación de pfsense, lo que haremos es utilizarlo como imagen por defecto, para no iniciar todo el proceso de creación del pfsense cada vez que creamos un laboratorio, para esto, nos vamos al siguiente directorio creado y hacemos un “**commit**” al disco virtual temporal para que sea la imagen por defecto para futuros laboratorios:

```
root@pnetlab:/opt/unetlab/tmp/1/1# ls
monitor.sock  virtioa.qcow2  wrapper.txt
root@pnetlab:/opt/unetlab/tmp/1/1# qemu-img commit virtioa.qcow2
Image committed.
```

Finalizamos con ajustar los permisos:

```
root@pnetlab:/opt/unetlab/addons# unl_wrapper -a fixpermissions
```

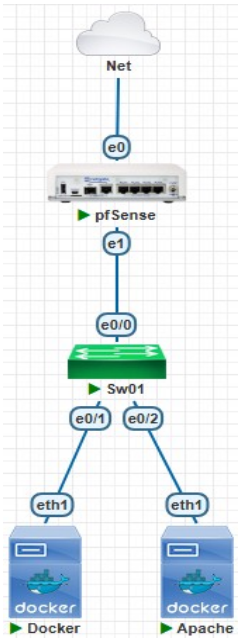
Eliminamos el fichero **cdrom.iso** en el directorio de pfsense:

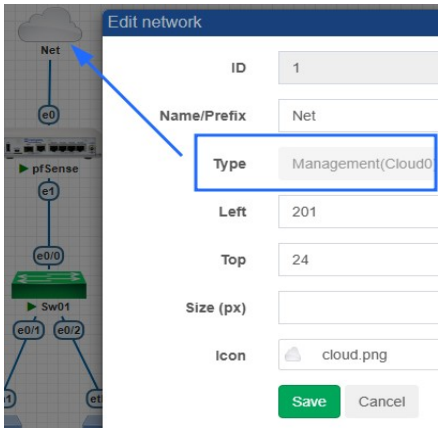
```
root@pnetlab:/opt/unetlab/addons/qemu/pfsense-CE-RELEASE-2.7.2# rm cdrom.iso
```

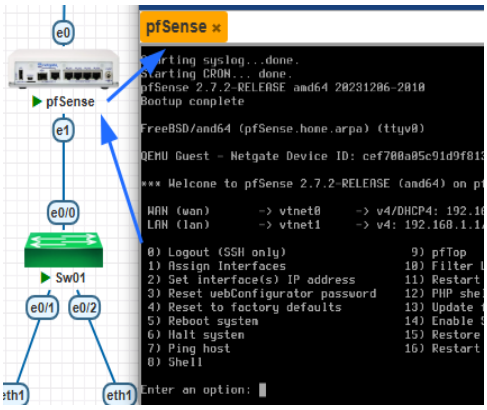
El usuario por defecto para pfsense es "**admin**" y el password por defecto es "**pfsense**".

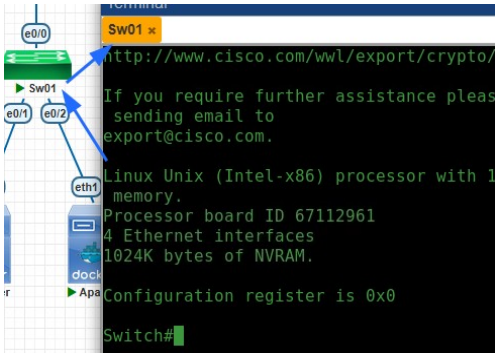
Ya la imagen de disco de pfsense podrá usarse en futuro, solo quedaría trabajar en los laboratorios que se proponga realizar.

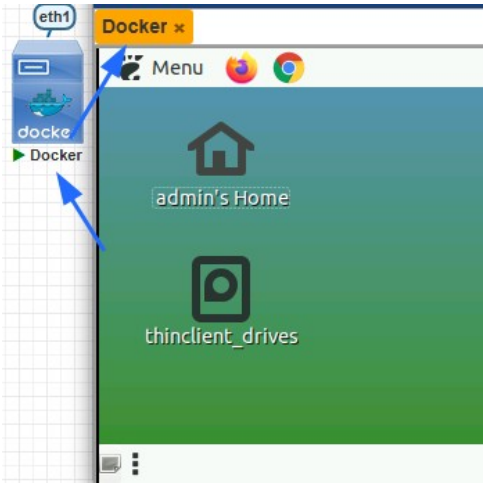
A continuación se muestra un lab donde esta nuestra solución Firwall/Router pfsense con la versión 2.7.2 junto con un Switch L2 (Cisco IOL), 2 computadoras uno con interfaz grafica y la otra maquina como servidor corriendo Apache, ademas un una nube, el cual representa la wan (esto esta detrás de otro segmento de red que da internet al laboratorio en PNetLab) esto es un ejemplo básico para ir trabajando en soluciones básicas a mas complejas:

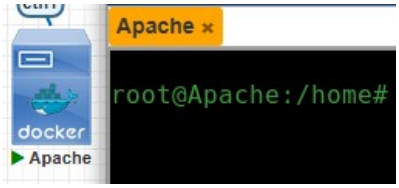




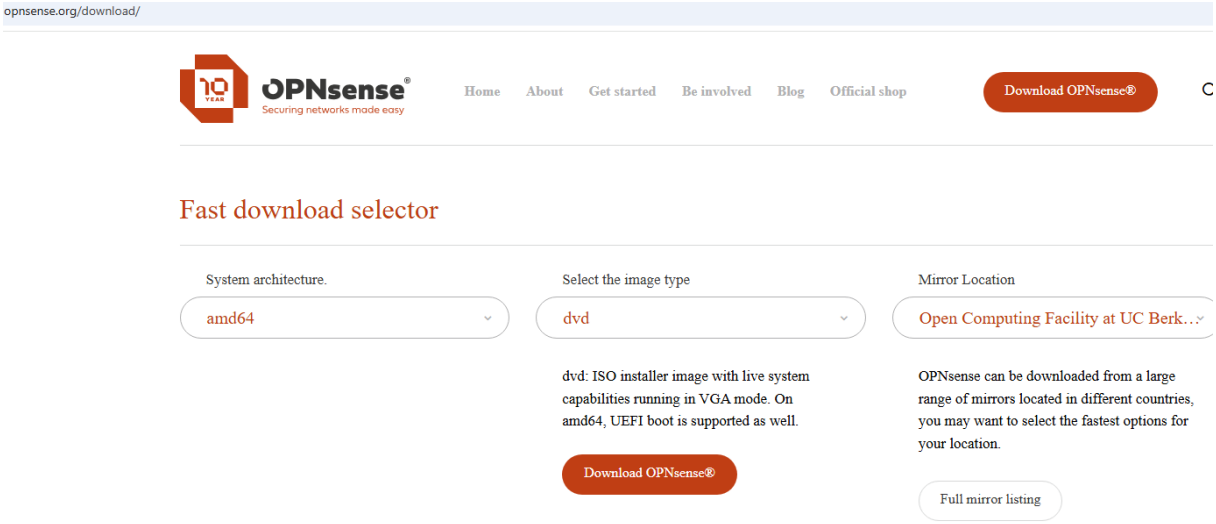




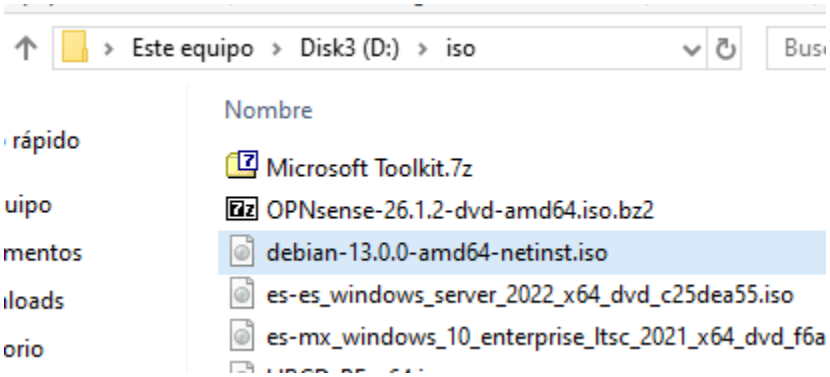




Creando imagen virtual para Opnsense y ser utilizado en PNetLab, para esto descargaremos el fichero .iso de su sitio oficial:



En mi caso, guardo la imagen iso instalador en otro disco de mi equipo desktop:



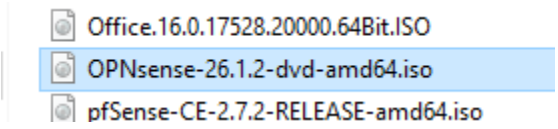
Me conecto al server PNetLab, para crear el siguiente directorio:

```
root@pnetlab:/opt/unetlab/addons/qemu# mkdir opnsense-26.1.2
```

Ingresamos hasta el directorio creado:

```
root@pnetlab:/opt/unetlab/addons/qemu# cd opnsense-26.1.2/
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

Ahora toca copiar el iso instalador de Opnsense una vez descomprimido, hacia nuestro servidor PNetLab específicamente en la carpeta recién creada:




```
D:\iso>scp OPNsense-26.1.2-dvd-amd64.iso root@192.168.3.3:/opt/unetlab/addons/qemu/opnsese-26.1.2
```

En la CLI de nuestro servidor, listamos el contenido del directorio de opnsense-26.1.2, esto para verificar que se haya copiado correctamente:

```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# ls
OPNsense-26.1.2-dvd-amd64.iso
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

Procedemos a cambiar el nombre de fichero "**OPNsense-26.1.2-dvd-amd64.iso"** a "**cdrom.iso"**:

```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# mv OPNsense-26.1.2-dvd-amd64.iso cdrom.iso
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

Creamos el disco virtual para que se instale OPNsense, con un tamaño de "**15G"** de espacio:

```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# qemu-img create -f qcow2 virtioa.qcow2 15G
Formatting 'virtioa.qcow2', fmt=qcow2 size=16106127360 cluster_size=65536 lazy_refcounts=off refcount_bits=16
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

Listamos el contenido de nuestro directorio de OPNsense:

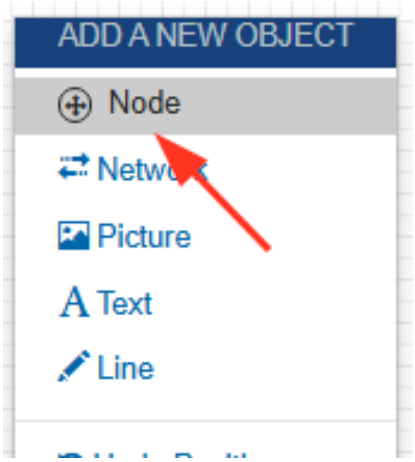
```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# ls
cdrom.iso  virtioa.qcow2
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

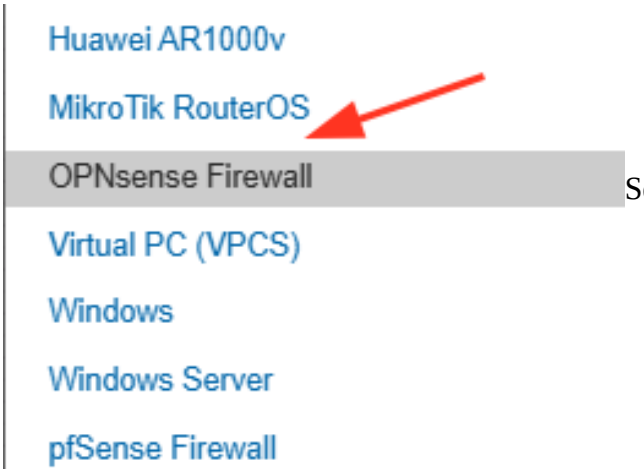
Procedemos a aplicar permisos a nuestro directorio de imágenes:

```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# unl_wrapper -a fixpermissions
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2#
```

Procedemos a crear nuestro lab de OPNsense para guardar el contenido ya instalado y ser utilizado en futuros labs, nos dirigimos a nuestro entorno de PNetLab:

Agregando un nodo





Seleccionamos nuestra imagen de OPNsense

En el formulario para este nodo (OPNsense), se puede ir modificando la cantidad de hilos para el CPU, RAM, números de puertos de red y lo importante, cambiar el tipo de consola para visualizar en pantalla la línea de comando (CLI) de OPNsense, para esto se selecciona la consola del tipo VNC, finalizamos haciendo clic en el botón de salvar:

Template

Show all unsupport

OPNsense Firewall

Number of nodes to add: 1

Image: opnsense-26.1.2

Name: OPNsense

Description: OPNsense Firewall

Icon: Firewall.png

First MAC:

First NIC: Auto

UUID:

CPU Limit

CPU: 2

RAM (MB): 4096

Primary Console: VNC

Primary Map Port:

Secondary Console: Empty

Secondary Map Port:

User Name:

Password:

Ethernet: 4

Qemu Arch: x86_64

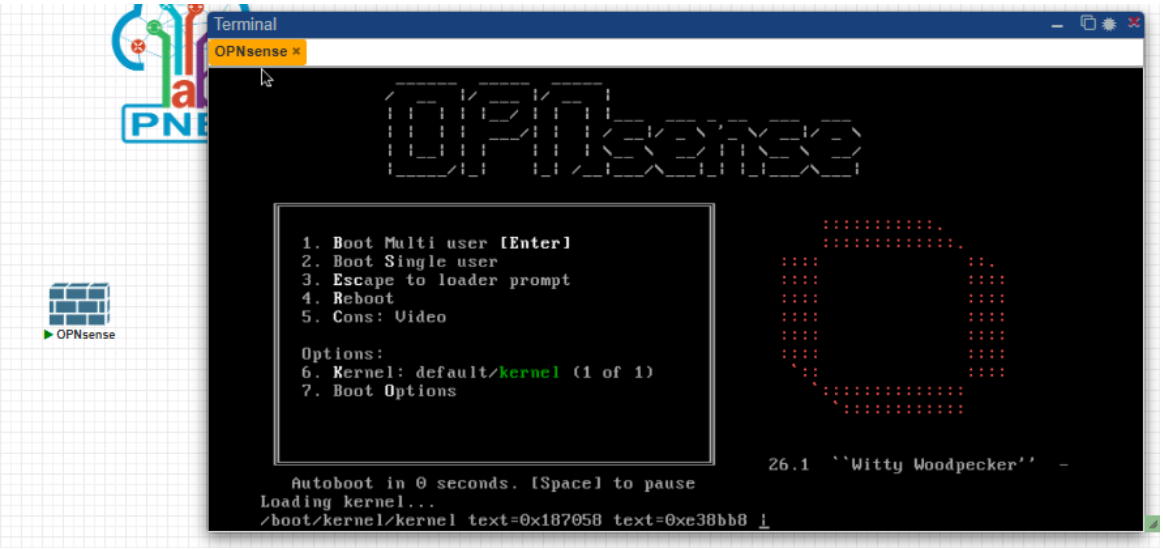
Qemu NIC: virtio-net-pci

Qemu Version: 4.1.0

Procedemos a arrancar nuestro OPNsense:

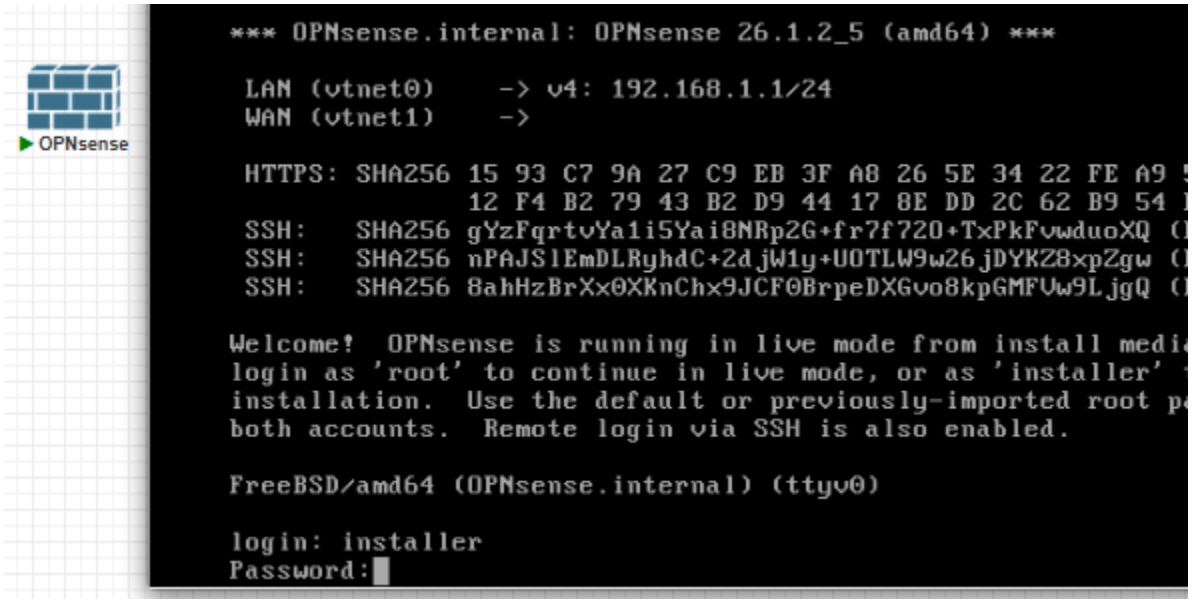


Una vez arrancado el OPNsense, hacemos doble clic sobre el icono y podremos observar que esta arrancando y cargando el boot para su instalación:

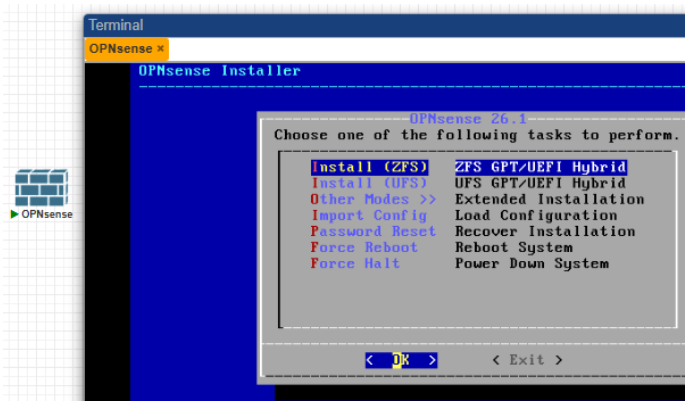


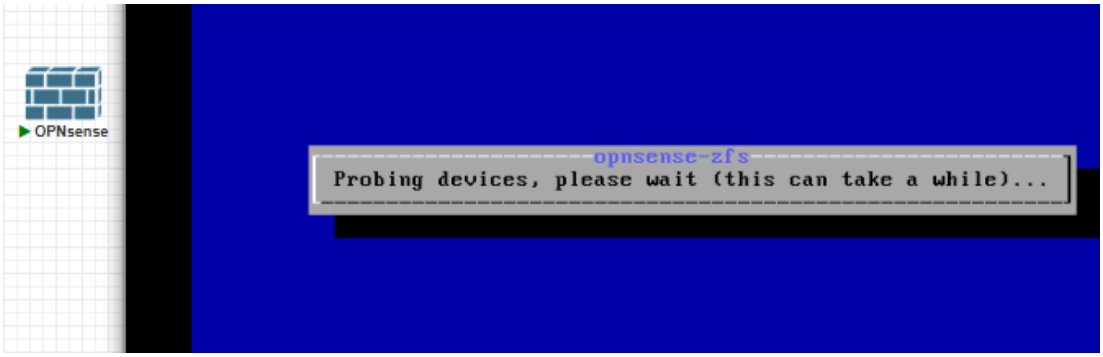
Una vez que termine de cargar el entorno de OPNsense del instalador, escribiremos según nos indica el sitio oficial de su documentación:

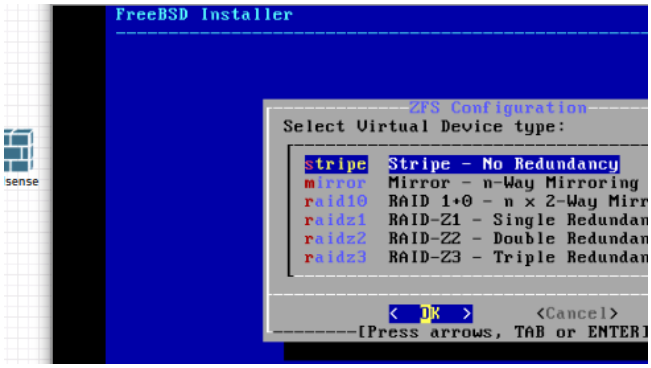


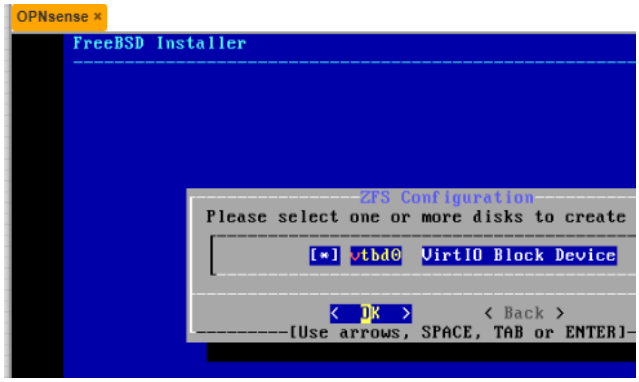


El paso anterior indica que inicialicemos el asistente de instalación, procedemos como todo asistente de instalación en la mayoría de los O.S y/o software, seguir sus pasos:

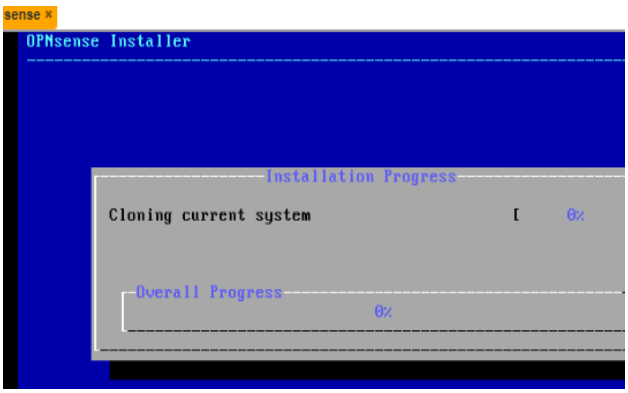


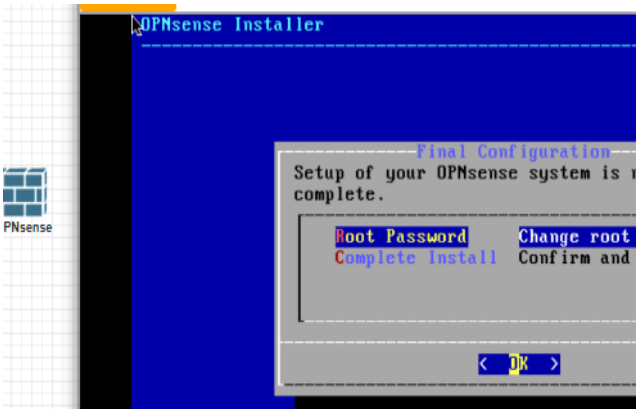


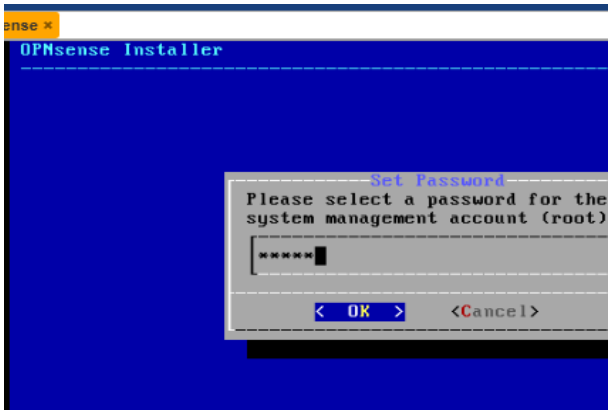

FreeBSD Installer
ZFS Configuration
Select Virtual Device type:
stripe Stripe - No Redundancy
mirror Mirror - n-Way Mirroring
raid10 RAID 1+0 - n x 2-Way Mirr
raidz1 RAID-Z1 - Single Redundan
raidz2 RAID-Z2 - Double Redundan
raidz3 RAID-Z3 - Triple Redundan
< OK >
<Cancel>
[Press arrows, TAB or ENTER]

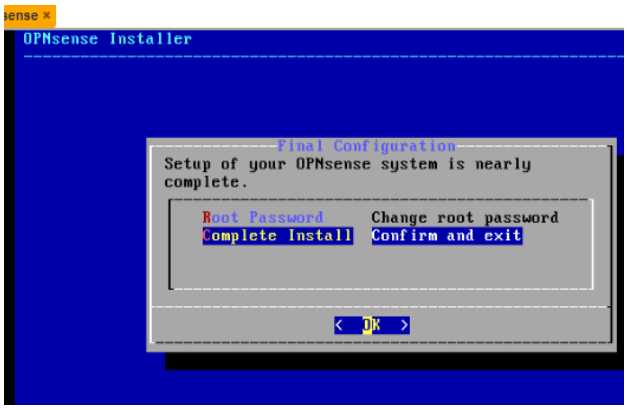

OPNsense
FreeBSD Installer
ZFS Configuration
Please select one or more disks to create
[*] vtbd0 VirtIO Block Device
< OK >
< Back >
[Use arrows, SPACE, TAB or ENTER]

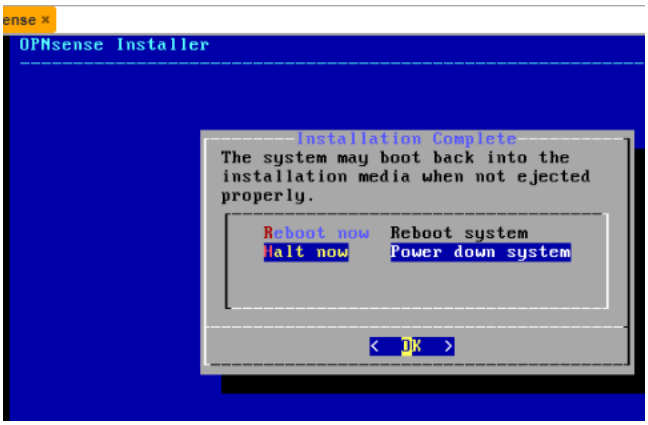

FreeBSD Installer
ZFS Configuration
Last Chance! Are you sure you want to
the current contents of the following
vtbd0
< YES >
< NO >
[Press arrows, TAB or ENTER]

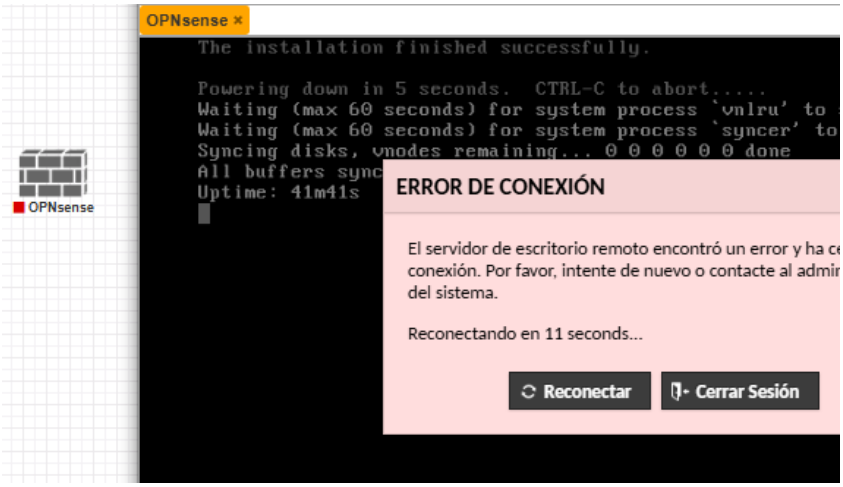

OPNsense Installer
Installation Progress
Cloning current system [0%
Overall Progress
0%


OPNsense Installer
Final Configuration
Setup of your OPNsense system is
complete.
Root Password Change root
Complete Install Confirm and
< OK >


OPNsense Installer
Set Password
Please select a password for the
system management account (root)
*****
< OK >
<Cancel>







Acto seguido, toca ir a este directorio, aquí se guarda temporalmente la imagen modificada, ya con datos de instalación para nuestro caso OPNsense, lo que haremos es utilizarlo como imagen por defecto, para no iniciar todo el proceso de instalación de OPNsense cada vez que creamos un laboratorio nuevo, para esto, nos vamos al siguiente directorio creado y hacemos un "**commit**" al disco virtual temporal para que sea la imagen por defecto para futuros laboratorios:

```
root@pnetlab:/opt/unetlab/tmp/1/1# ls
monitor.sock  virtioa.qcow2  wrapper.txt
root@pnetlab:/opt/unetlab/tmp/1/1# qemu-img commit virtioa.qcow2
Image committed.
```

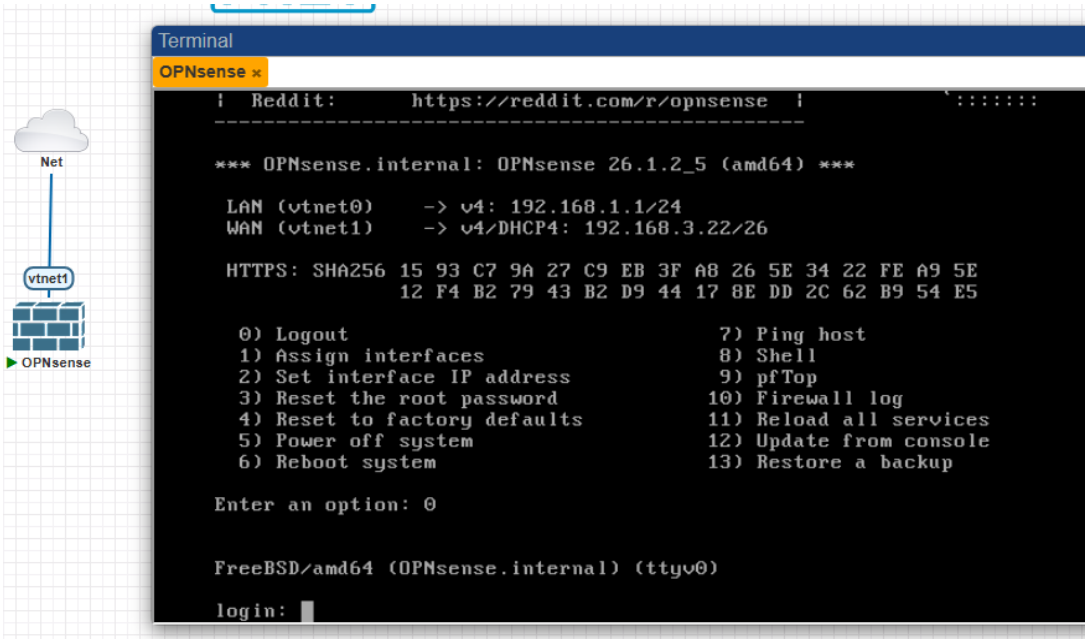
Borramos en la carpeta de OPNsense el fichero "**cdrom.iso**":

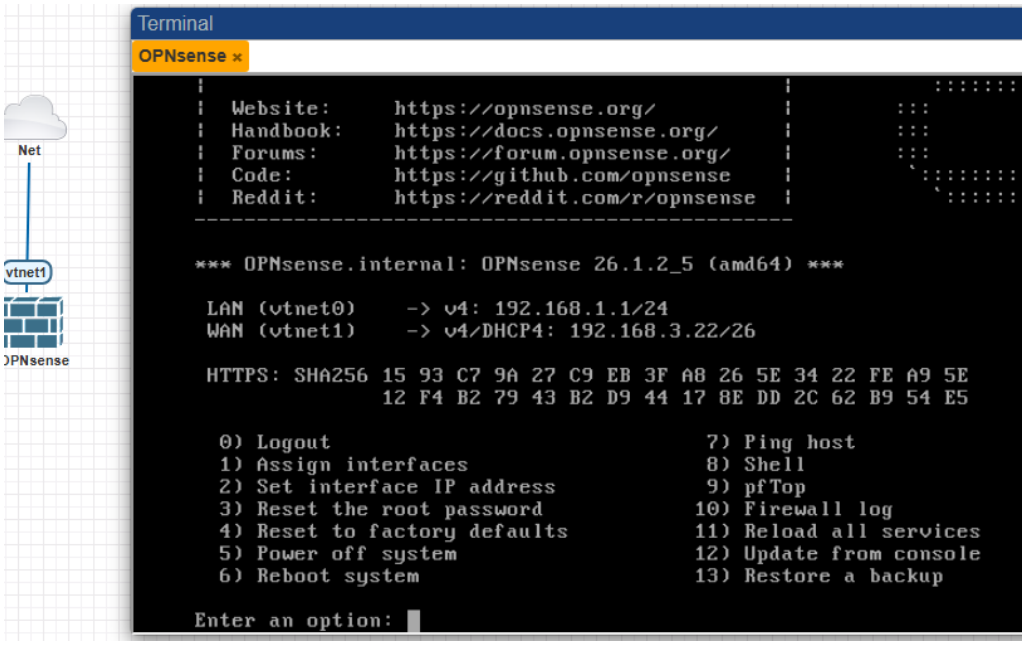
```
root@pnetlab:/opt/unetlab/addons/qemu/opnsense-26.1.2# rm cdrom.iso
```

Finalizamos con ajustar los permisos:

```
root@pnetlab:/opt/unetlab/addons# unl_wrapper -a fixpermissions
```

Recrearemos una laboratorio de prueba con 2 tarjetas de red para OPNsense, el usuario es "**root**" y el password para este caso le coloque en el paso de asistente de instalación uno de prueba "**admin**":

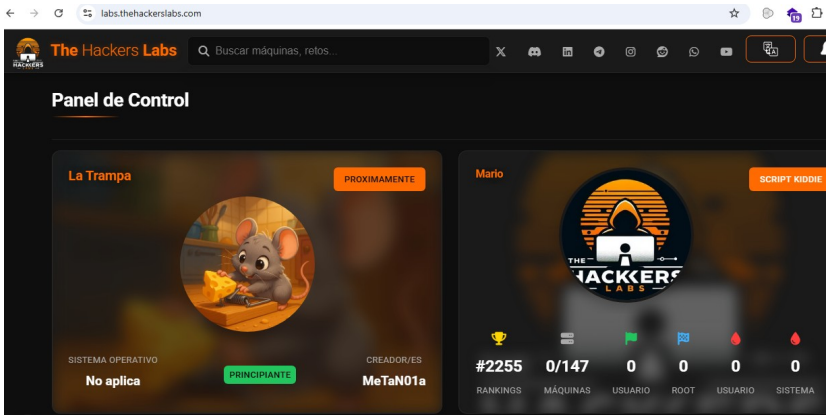


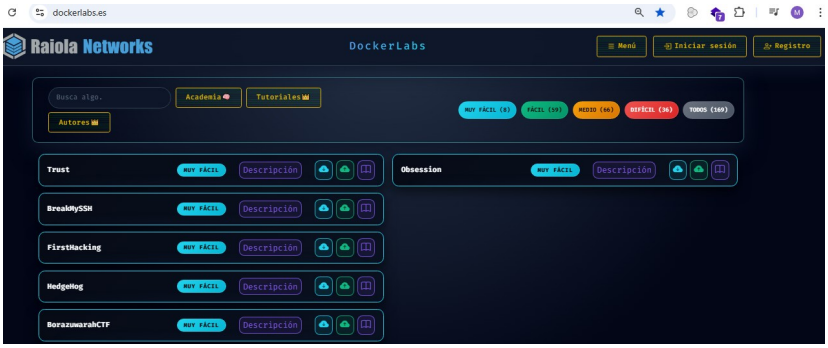


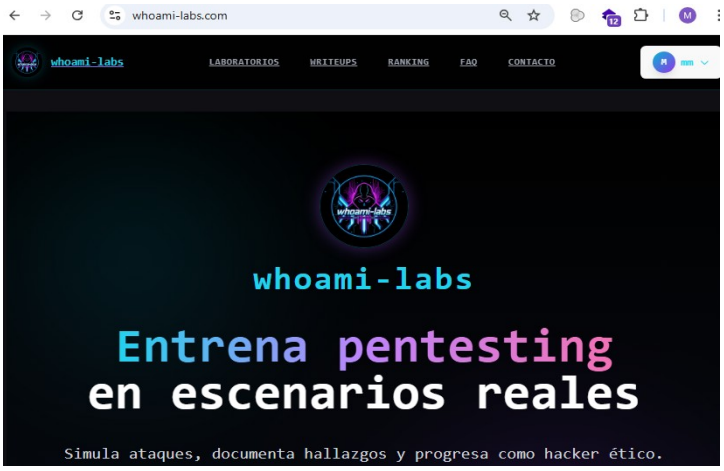
Mas información sobre la creación de ambas soluciones, pueden apoyarse en las siguientes urls:
https://www.eve-ng.net/index.php/3380-2/
https://www.eve-ng.net/index.php/documentation/howtos/opnsense-firewall/

Implementando laboratorios para prácticas en Ciberseguridad

En este paso, implementaremos un laboratorio para ciberseguridad, par esto, trabajaremos con recursos libres de utilizar para este tema. Dos recursos que estaremos trabajando los laboratorios, están en las siguientes url: https://labs.thehackerslabs.com/, https://dockerlabs.es/ y otra plataforma muy interesante y nueva para mi, el cual agradezco al desarrollador poder aportar a la comunidad realizando este documento practico https://whoami-labs.com/.







Cabe mencionar que estos laboratorio, en el caso de El pingüino de Mario – dockerlabs.es y la plataforma whoami-labs.com, los labs están basados en contenedores Docker, por tanto, la persona que vean este apartado, deben de manejar lo básico del tema, son máquinas GNU/Linux, están preparadas para ser cargados a nuestro PNetLab. Para el caso de los laboratorios en la plataforma de https://labs.thehackerslabs.com/, esta basado en máquinas preparados en Virtual Box, sin embargo esto es posible importarlo y llevarlo a Proxmox y recrearlo. Toma echar mano para ir creando nuestro entorno de trabajo, empezamos con https://dockerlabs.es.

Descargamos un lab de ejemplo desde la url de dockerlabs.es:



El lab a elegir es "***BreakMySSH***"

Nos abre mega y lo que debemos ver es el tamaño del fichero en formato "***zip***", damos clic en el botón de "***Descargar***"



Una vez descargado en nuestro equipo, lo descomprimimos y lo copiaremos a nuestro servidor PNetLab:

Si estas en GNU/Linux, ejecutar el siguiente comando: ***unzip brekmyssh.zip***

Si estas usando Windows: usas cualquier programa como 7zip, winzip, winrar, etc.

Toca copiarlo al servidor PNetLab:

scp breakmyssh.tar root@ip_server_pnetlab:/directorio_destino ejemplo:

scp breakmyssh.tar root@192.168.3.3:/opt

Nos logueamos vía ssh a nuestro servidor PNetLab:

ssh root@192.168.3.3

```
C:\Users\PC>ssh root@192.168.3.3
root@192.168.3.3's password:
```

Nos ubicamos en el directorio de destino de nuestra imagen docker:

cd /opt

```
root@pnetlab:~# cd /opt
root@pnetlab:/opt# _
```

Cargamos la imagen con el comando:

docker load -i breakmyssh.tar

```
root@pnetlab:/opt# docker load -i breakmyssh.tar
```

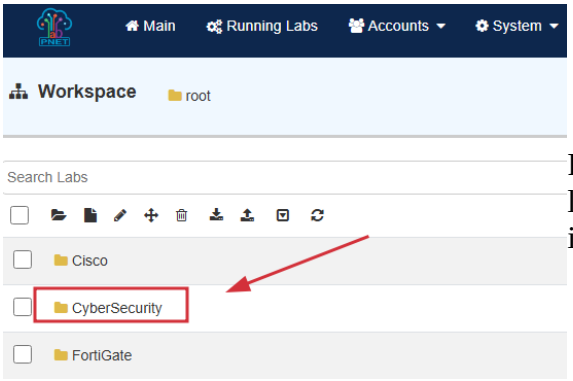
Verificamos que este la imagen cargada en nuestro PNetLab:

docker images

```
root@pnetlab:/opt# docker images
```

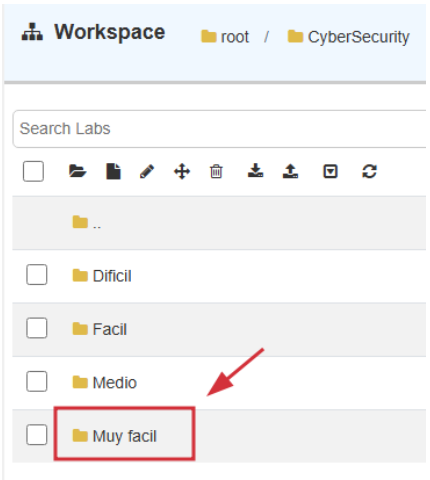
```
dockerlabs          latest    4bbe8c200887   18 months ago   269MB
aguademayo          latest    7c95151e347f   18 months ago   1.4GB
injection           latest    7f560ad3f94b   18 months ago   798MB
breakmyssh          latest    f470b37d67e9   18 months ago   111MB
hiddencat           latest    670fe7dea428   18 months ago   666MB
secretjenkins       latest    687a7f99d37c   18 months ago   576MB
chocolatelovers     latest    a969ddae54e4   18 months ago   368MB
library             latest    436d5fbb01a4   18 months ago   647MB
firsthacking        latest    0141c7bda17c   18 months ago   269MB
```

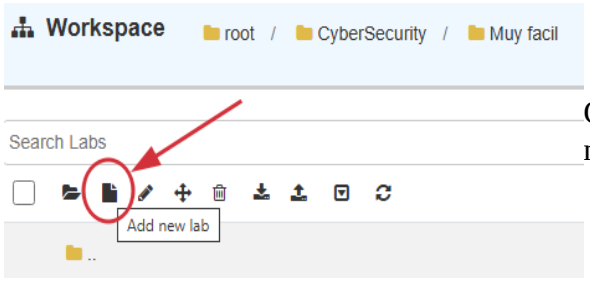
Montemos un lab con nuestro contenedor vulnerable listo para practicar, nos vamos al navegador y abrimos PNetLab:



Previamente, ya había creado un directorio de trabajo llamado "***CyberSecurity***", esto por temas de orden, ingresamos a ese espacio de trabajo:

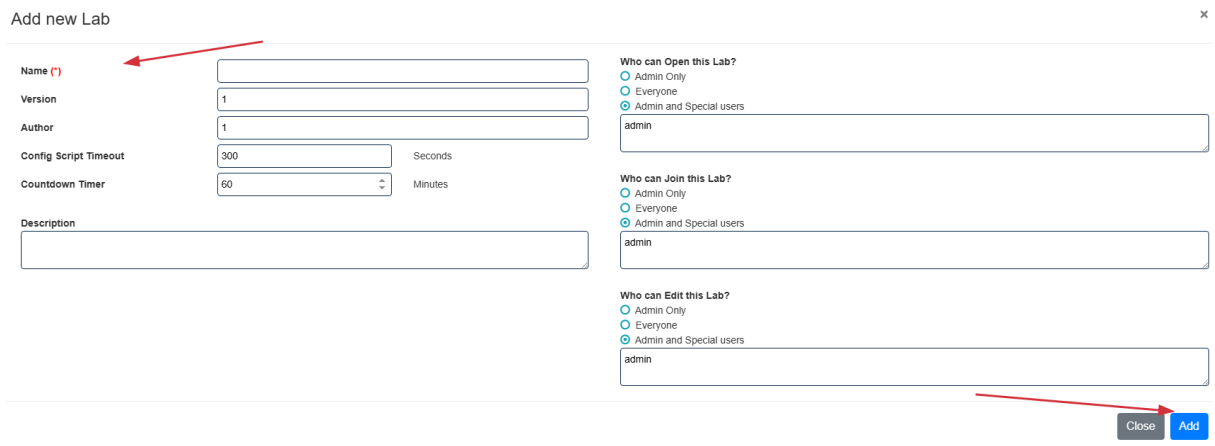
Hacemos clic en el directorio "***Muy facil***", esto es mi caso, recuerden que deben ir creando sus espacios de trabajo, para que tengan orden según las maquinas docker a practicar:



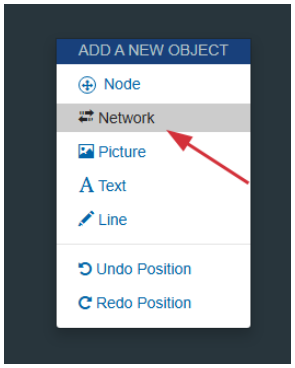


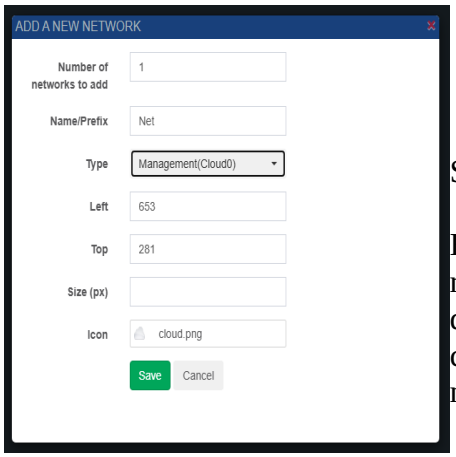
Creamos un nuevo laboratorio haciendo clic según a como muestra la imagen:

Rellenamos el formulario, con el nombre de la máquina docker, alguna descripción, etc.., al final, clic en el botón "***Add***"



Estando en el espacio de trabajo del laboratorio, agregaremos una red que estará conectada a nuestra red interna física, el cual trabajaría como ISP.

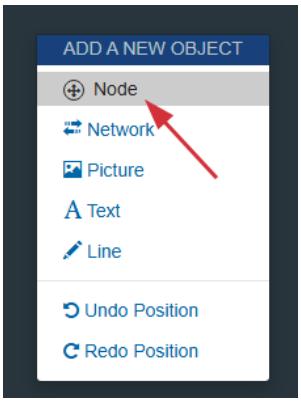


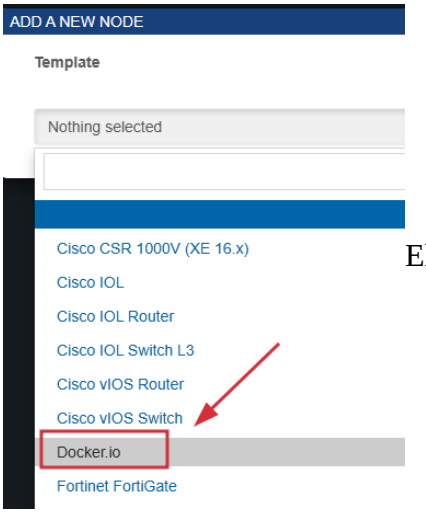


Seleccionamos en ***Type → Management(Cloud0)***, clic en "***Save***"

Este "***Management(Cloud0)***", en la configuración de interfaz de red de la VM de PNetLab, este hace un puente a nuestra red física que alcanza nuestros equipos físicos. Así podemos interactuar con cualquier equipo físico conectado a esta interfaz hacia nuestra máquina docker vulnerable para practica.

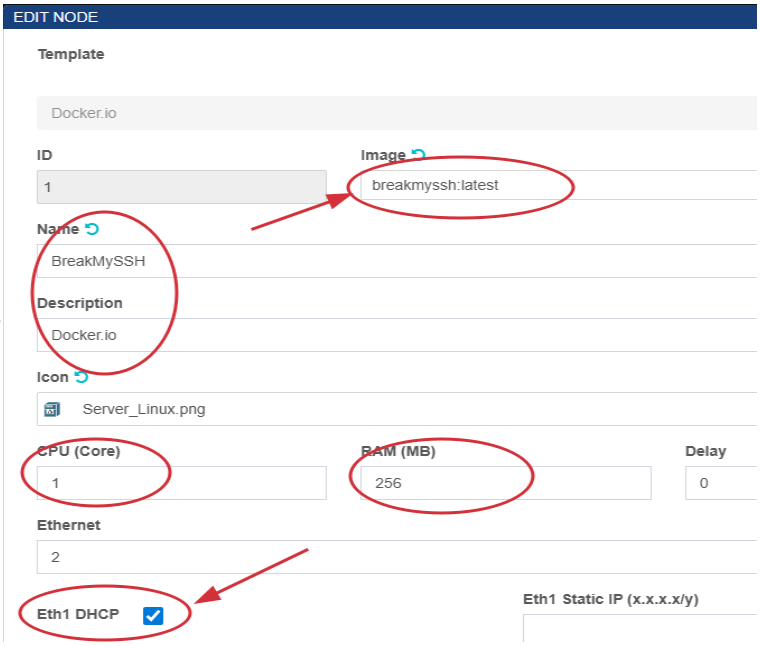
Ahora, toca agregar la máquina docker "***breakmyssh***", agregando un nodo y eligiendo dicha máquina:

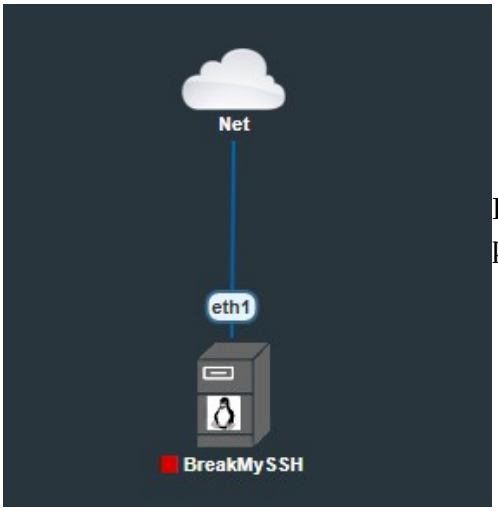




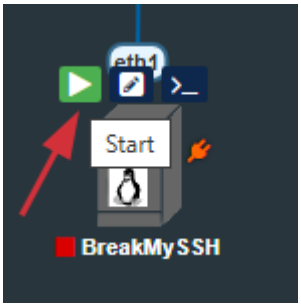
Elegimos "***Docker.io***" :

Al final, seleccionamos nuestro contenedor docker “breakmyssh:latest, algún nombre, descripción, algún icono, cantidad de RAM, y habilitamos modo DHCP en ETH1”:

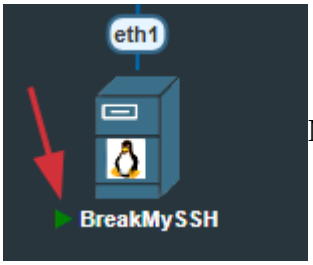




Laboratorio listo, ahora lo ejecutamos, solo la máquina docker, puesto que el nodo que dice “**Net**”, no necesita inicializar.

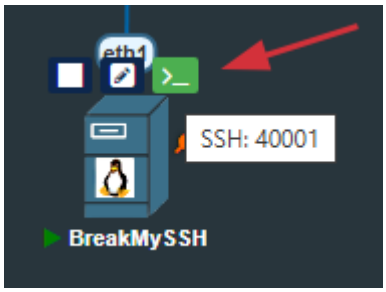


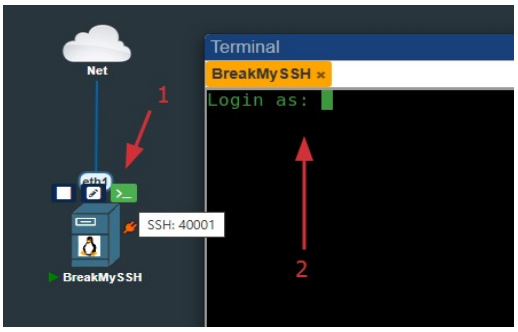
Observara que al hacer clic en el icono de “**play**”, toma otro color, en este caso a verde.

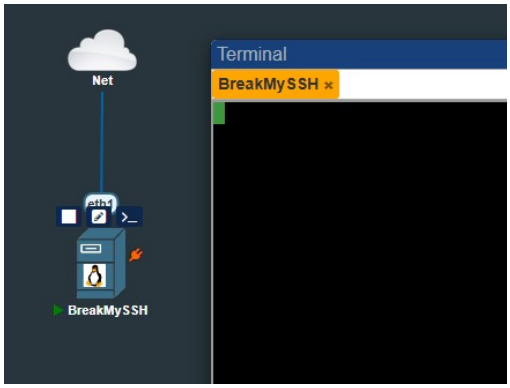


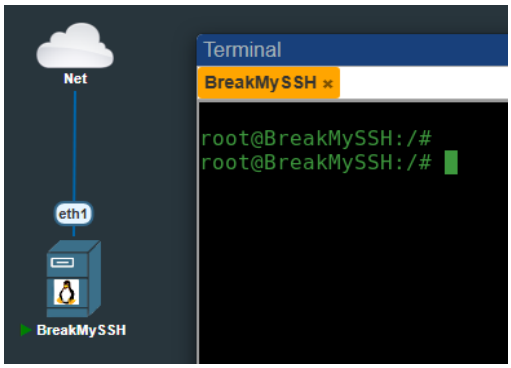
Máquina inicializada, procederemos a verificar.

Para verificar que este inicializada sin errores, pasamos el puntero del mouse sobre el icono y se nos aparece el icono de un prompt de color verde, al hacer clic ahí, nos solicitara usuario y constraseña, pero si hacemos doble clic sobre la máquina docker y hacemos enter, aparecera el prompt sin solicitar usuario y contraseña, o sea, ya logeado, esto para verificar que ip esta utilizando en este momento y empezar a realizar nuestras prácticas.









TIPS:

Para copiar múltiples ficheros zip hacia nuestro servidor PNetLab:

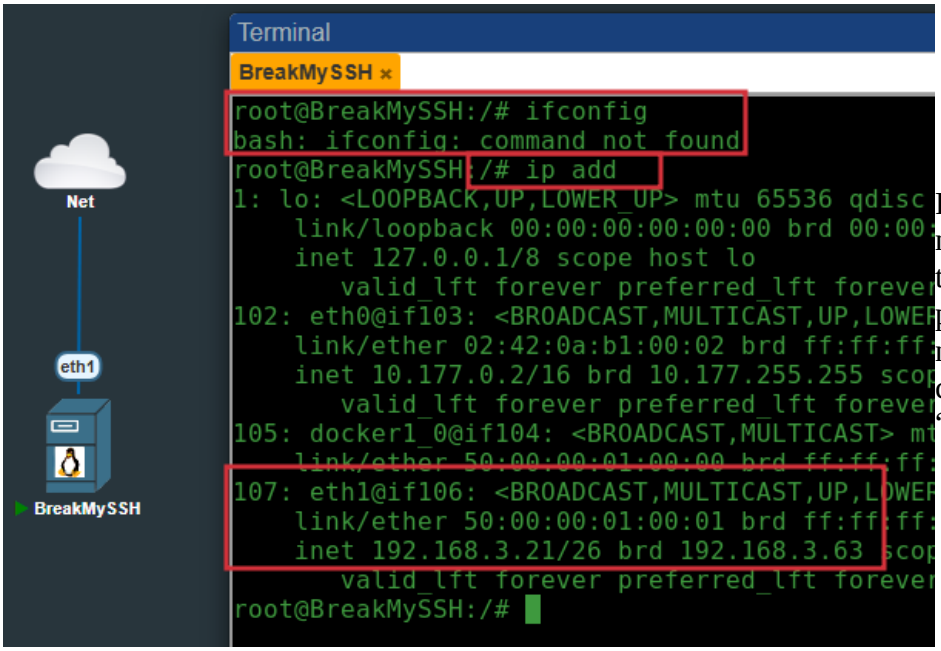
scp *.zip root@192.168.3.3:/opt

Para descomprimir múltiples ficheros zip:

ssh root@192.168.3.3
cd /opt
unzip '*.zip'

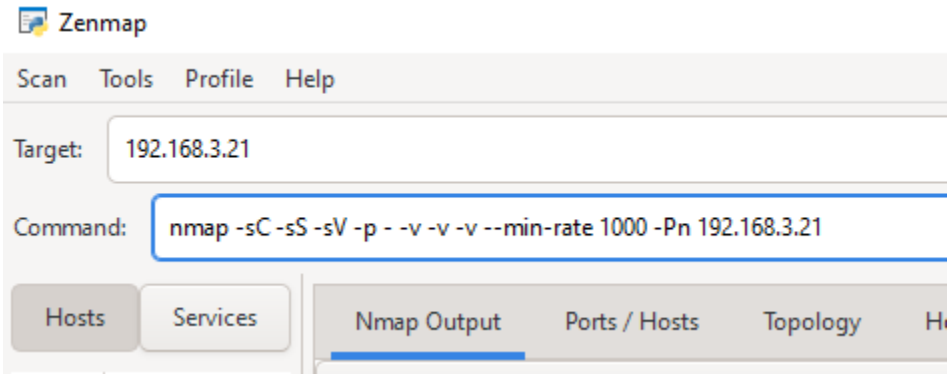
Una gran ayuda para poder cargar todas la imágenes de docker, estando todos los ficheros “tar” en un mismo directorio, podremos ejecutar la siguiente linea de comando:

ls -1 *.tar | xargs --no-run-if-empty -L 1 docker load -i



Para verificar la ip de la máquina docker e ir trabajando en nuestra practica, podemos ejecutar ***ifconfig***, si no funciona, ***ip add***: en mi caso me resolvió la ip “***192.168.3.21***”:

Para válida, empezamos a scanner la máquina docker, en mi caso lanzaré “**nmap**” en su versión GUI para windows “***zenmap***”:



Resultado del scan:

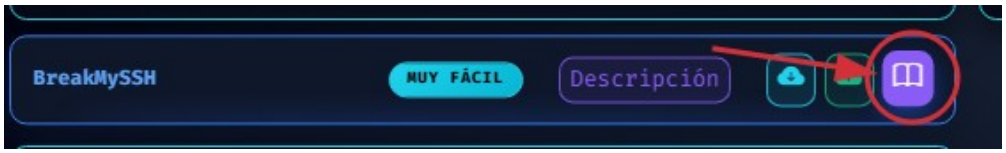
```
Scanned at 2025-11-25 17:06:03 Hora estándar, América Central for 10s
Not shown: 65532 closed tcp ports (reset)
PORT     STATE    SERVICE    REASON          VERSION
22/tcp   open     ssh        syn-ack ttl 63 OpenSSH 7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 1a:cb:5e:a3:3d:d1:da:c0:ed:2a:61:7f:73:79:46:ce (RSA)
| ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDfOr49bj2kh3ab2WutTu6Jx7NA7OKSxzp42bJU4nqtQlICZbjiBXhOa1ZKOfUfNv
U3TQHwtUBsNSR3qFscgJQlniCyrfH/
4rbUk5jiLYN6y8NjctGvsvwPE+cCiFVge76qyfzmZdaf5gJT9DKDt47iBkrngCODYrqqt+Bbl9ZEGh5SUfDqYfsFMIvlsSj
|   256 54:9e:53:23:57:fc:60:1e:c0:41:cb:f3:85:32:01:fc (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBLJ77V//dhC1BX2KXpMNu
|   256 4b:15:7e:7b:b3:07:54:3d:74:ad:e0:94:78:0c:94:93 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICFLUqv+frul58FgQLXP91bNrTRC9d1X545DZJ0wsw6z
8110/tcp filtered unknown    no-response
8111/tcp filtered skynetflow no-response

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 17:06
Completed NSE at 17:06, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 17:06
Completed NSE at 17:06, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 17:06
Completed NSE at 17:06, 0.00s elapsed
Read data files from: C:\Program Files (x86)\Nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.67 seconds
           Raw packets sent: 65537 (2.884MB) | Rcvd: 65535 (2.621MB)
```

Con esto ya podemos trabajar en las máquinas que nos brinda la plataforma de https://dockerlabs.es

Podemos ir agregando los pasos a resolver de las máquinas que están en esta plataforma y adjuntarlo en cada lab, para que podamos seguir un mecanismo seguro hasta no depender de ello, esto lo logramos bajando algún writeup de alguna máquina, en este caso el de "***breakmyssh***"

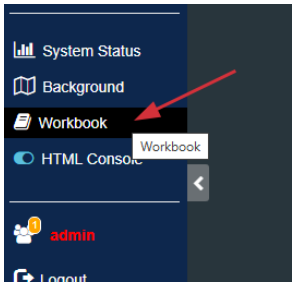
Bajamos alguna guía (WriteUp), puede que sean varios colaboradores, o algunas veces nadie lo ha resulto el cual no nos mostrará ninguna resolución de alguna maquina:

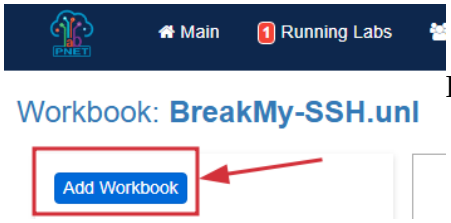




Podemos observar que hay varios colaboradores que ya han resulto esta máquina, podemos ver cada uno de ellos, al hacer clic en uno de ellos nos reenvía a otro enlace donde esta la solución, copiamos el contenido y lo guardamos en formato pdf para subirlo a nuestro laboratorio.

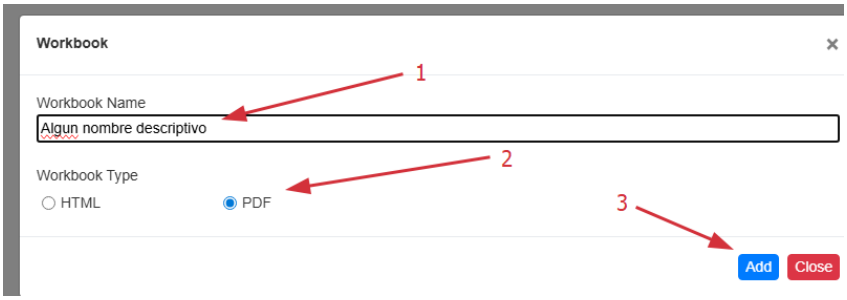
Una vez copiado el contenido, guardado y exportado en formato pdf, vamos al laboratorio y hacemos clic en "***Workbook***" éste está en el lado lateral izq del espacio de trabajo del laboratorio:

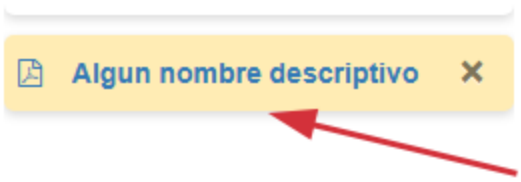




Hacemos clic en "***Add Workbook***":

Agregamos un nombre y seleccionamos el formato a subir que es del tipo pdf, luego clic en "***add***"



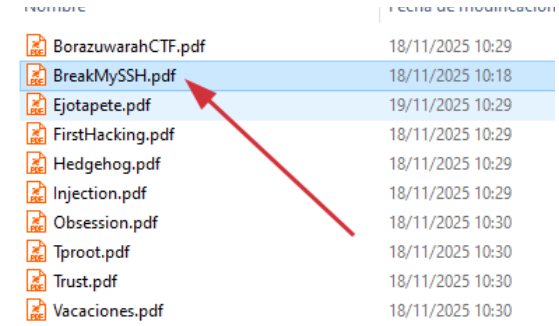


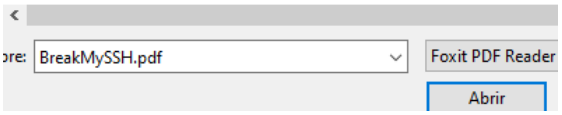
Clic sobre el nombre del lab y se nos despliega al lado derecho un botón de subir un archivo pdf:

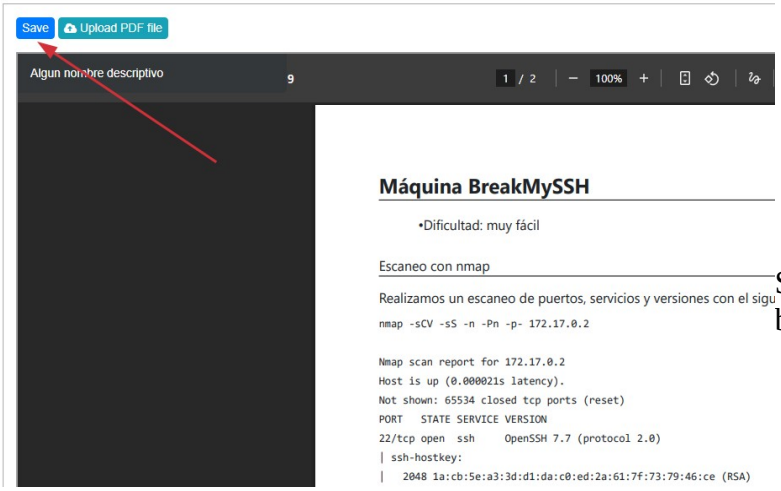
Subimos el fichero pdf de la solución del lab o alguna actividad en específico:



Buscamos y seleccionamos el fichero pdf de la máquina o laboratorio a resolver:

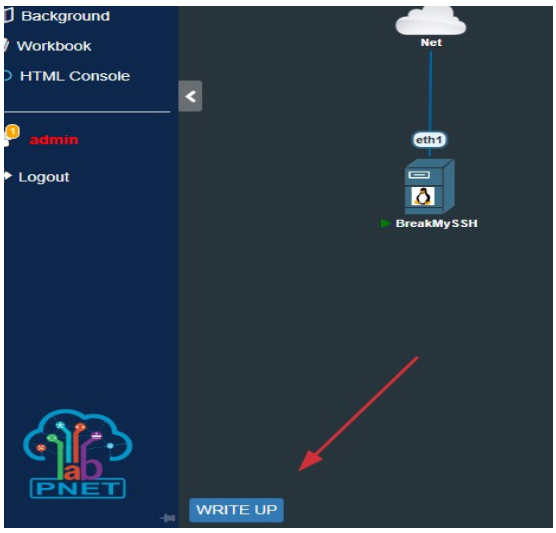


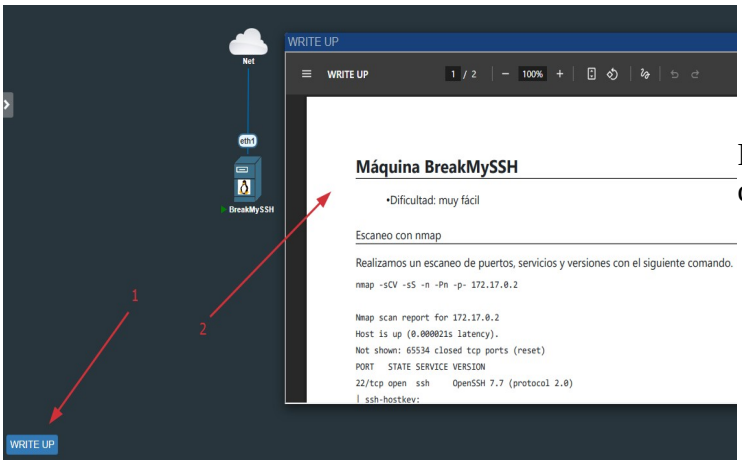




Se abre el documento y hacemos clic en el botón "***Save***"

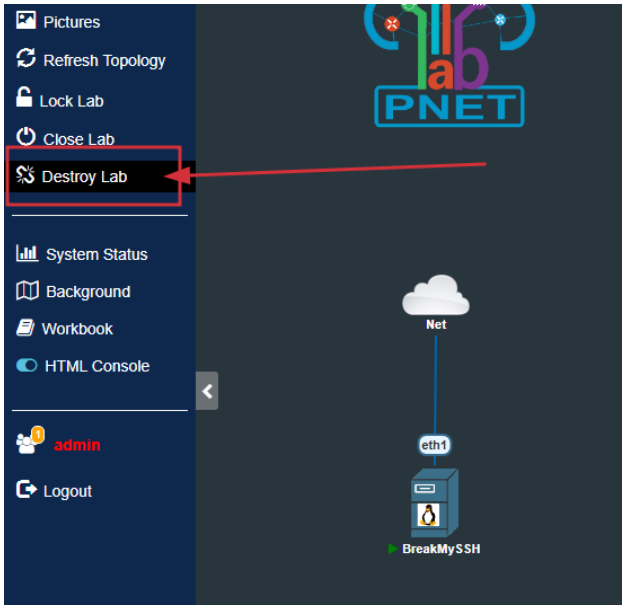
Regresamos al laboratorio y podemos ver en la parte inferior izquierda un botón con la descripción del documento pdf con el nombre que le haya dado:

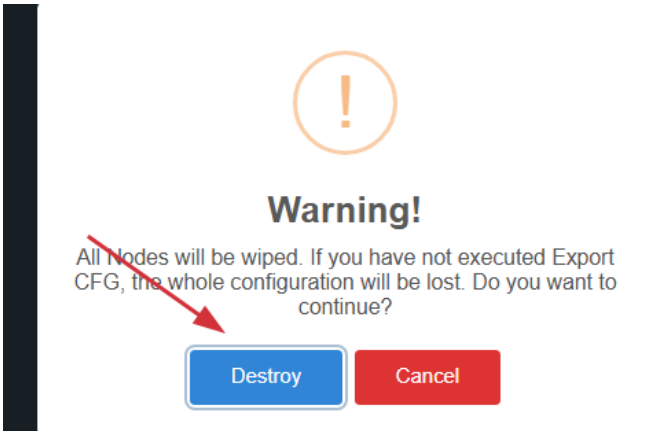




Probamos si abre y listo, podemos ver su contenido:

Finalizando el laboratorio, haciendo clic en "***Destroy Lab***":

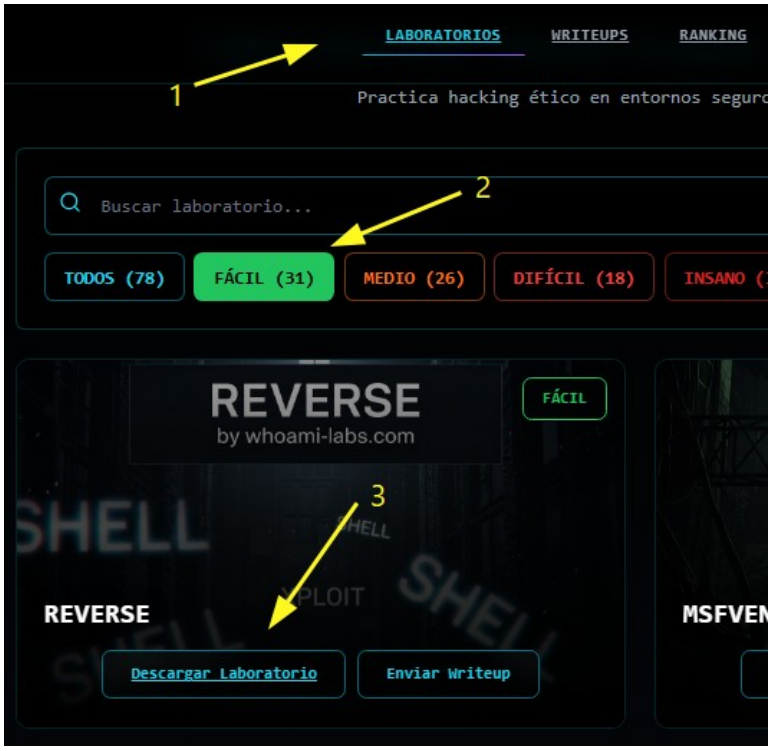


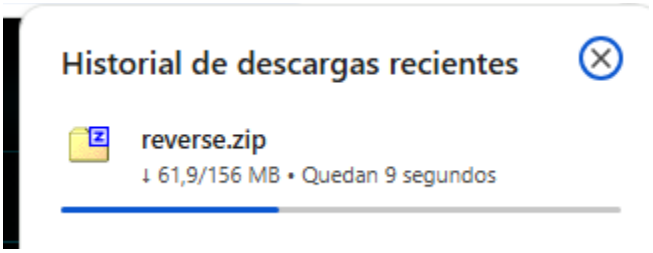


Confirmamos que deseamos destruir nuestro laboratorio (Finalizar nuestro Lab, detener y apagar las maquinas), clic en botón "***Destroy***":

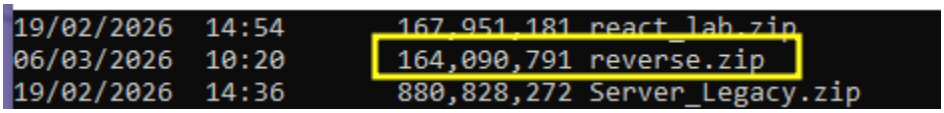
Con la plataforma de https://whoami-labs.com/, es similar el procedimiento al mostrado anteriormente, pero damos inicio práctico para agregarlo a nuestro entorno de PNetLab:

Nos dirigimos a nuestra plataforma https://whoami-labs.com/, nos registramos y nos vamos a la sección de laboratorios, los cuales, están categorizados por "**Fácil, Medio, Difícil, Insano**", para esto descargamos un laboratorio del tipo "**Fácil**" cuya maquina es "**REVERSE**":

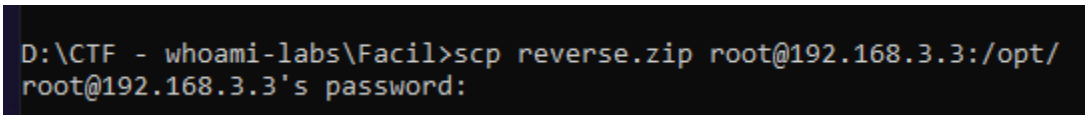




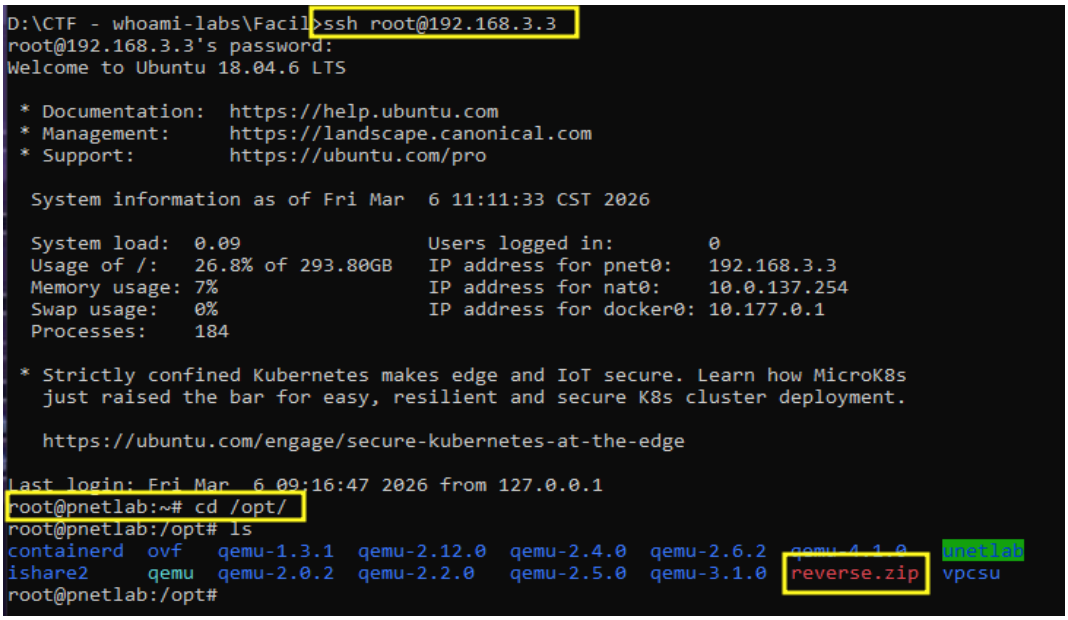
Nos dirigimos a nuestro directorio donde hemos guardado nuestro laboratorio:



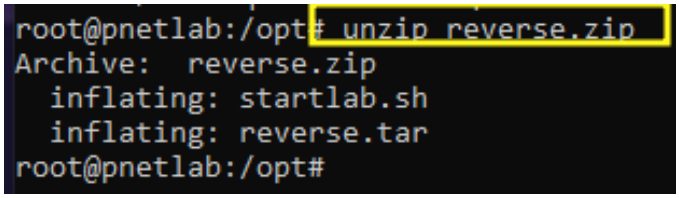
Copiamos nuestra máquina “reverse.zip”a nuestro server PNetLab:



Nos conectamos a nuestro server PNetLab y procederemos a descomprimir la máquina:



Descomprimimos:



Borramos ficheros que de momento no son necesarios para importar nuestra máquina:

```
root@pnetlab:/opt# rm reverse.zip startlab.sh
```

Toca cargar nuestra máquina "**reverse**" a nuestra plataforma PNetLab para que aparezca en las imágenes a utilizar:

```
root@pnetlab:/opt# docker load -i reverse.tar
66e9c7c89f57: Loading layer [==================================================>]  30.26MB/30.26MB
294a4de3dc9e: Loading layer [==================================================>]  133.3MB/133.3MB
5e1e8eb2633c: Loading layer [==================================================>]  2.13MB/2.13MB
0b78ab0da26a: Loading layer [==================================================>]    113B/113B
bd044d00ee73: Loading layer [==================================================>]  3.728kB/3.728kB
5a25c2f34bb9: Loading layer [==================================================>]    514B/514B
5d6109fb3ecc: Loading layer [==================================================>]    243B/243B
1410612eb6e2: Loading layer [==================================================>]    108B/108B
c573873158d6: Loading layer [==================================================>]  5.019kB/5.019kB
ea1878ccaa6e: Loading layer [==================================================>]  5.028kB/5.028kB
ebec967b9c15: Loading layer [==================================================>]    333B/333B
Loaded image: reverse:latest
root@pnetlab:/opt#
```

Verificamos que se haya importado, esto lo logramos con el siguiente comando:

```
root@pnetlab:/opt# docker images
REPOSITORY                TAG       IMAGE ID       CREATED         SIZE
reverse                   latest    d9f6d585caa0   11 days ago     464MB
aidor                     latest    c8178db94dfe   3 months ago    282MB
grooti                    latest    03ed9e6feb27   7 months ago    1.95GB
ejotapete                 latest    63b62f101614   7 months ago    540MB
winfake                   latest    399d077fb96d   7 months ago    336MB
ping_ctf                  latest    409e06e30057   8 months ago    620MB
```

Ya podemos eliminar al fichero importado para ahorrar espacio:

```
root@pnetlab:/opt# rm reverse.tar
```

Para poder copiar varios ficheros descargados en nuestra pc y querer trasladarlo todos a nuestro server PNetLab, podremos hacer lo siguiente:

Para copiar múltiples ficheros zip hacia nuestro servidor PNetLab:

scp *.zip root@192.168.3.3:/opt

Para descomprimir múltiples ficheros zip:

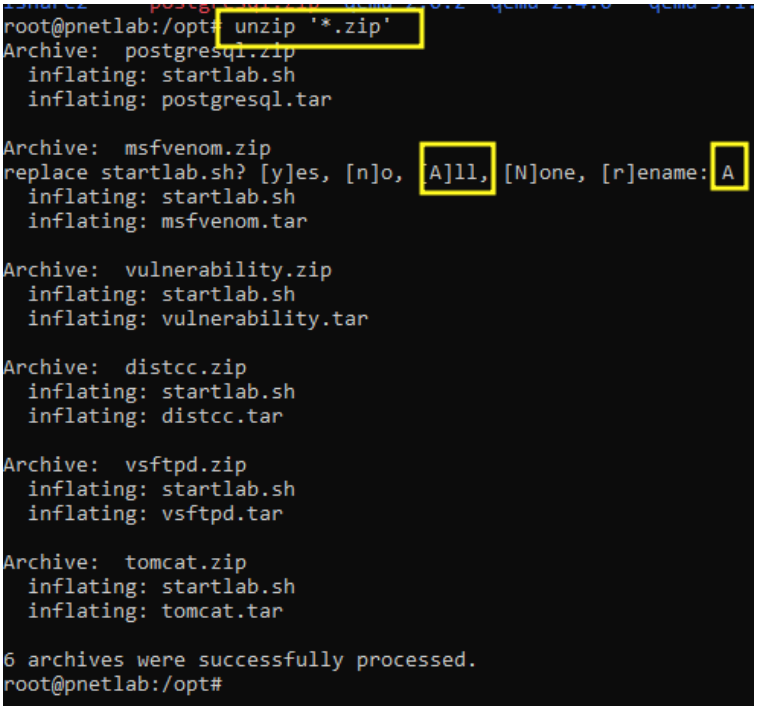
ssh root@192.168.3.3
cd /opt
unzip '*.zip'

Una gran ayuda para poder cargar todas la imágenes de docker, estando todos los ficheros “tar” en un mismo directorio, podremos ejecutar la siguiente linea de comando:

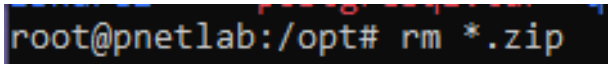
ls -1 *.tar | xargs --no-run-if-empty -L 1 docker load -i

En mi caso tengo ya en el server las maquinas “**msfvenom.zip, distcc.zip, tomcat.zip, vsftpd.zip, postfreesql.zip y vulnerability.zip**”, descomprimimos e importamos las máquinas:

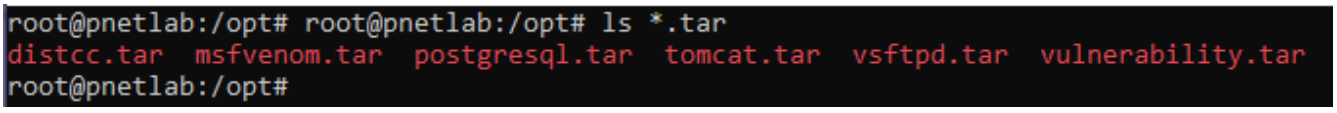
hay una advertencia al descomprimir varios ficheros zip, esto porque todos los ficheros en su interior tiene un fichero que se llama startlab.sh, este se repite en todos los ficheros, por eso respondemos reemplazar todo “**[A]**”



Ahorramos espacio borrando todos los ficheros zip que ya no ocupamos:

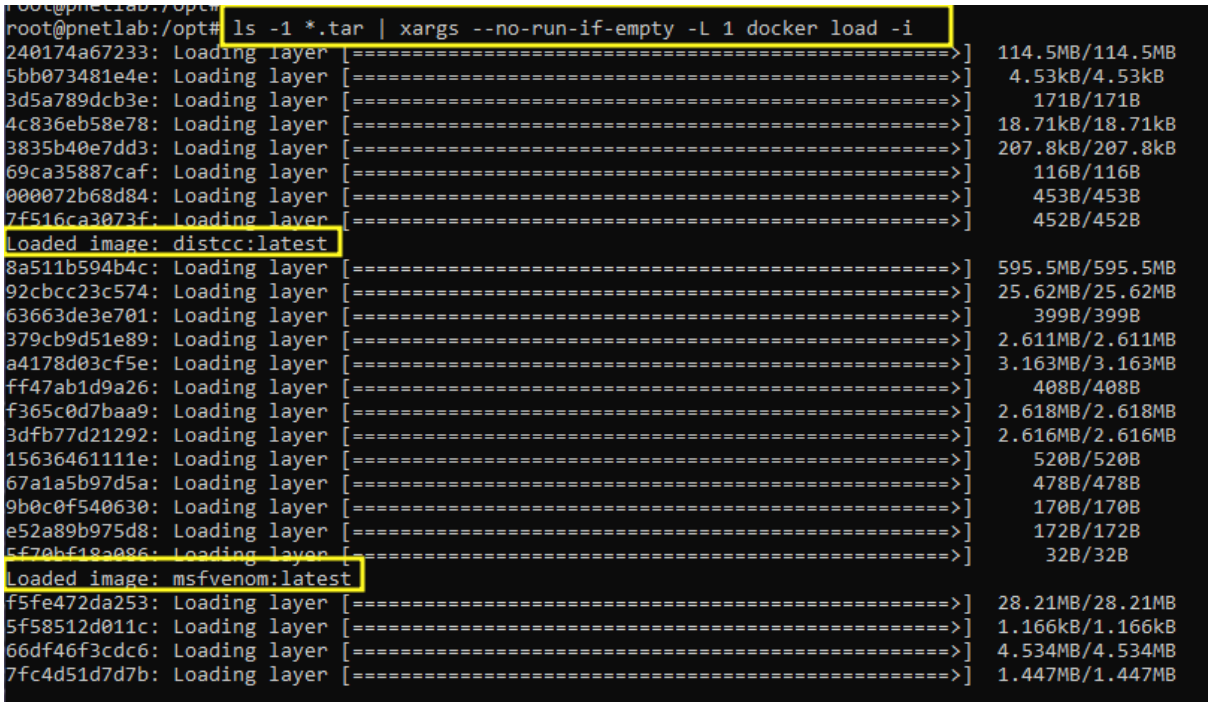


Verificamos que este los ficheros descomprimidos en su formato para importarlos “.tar”:



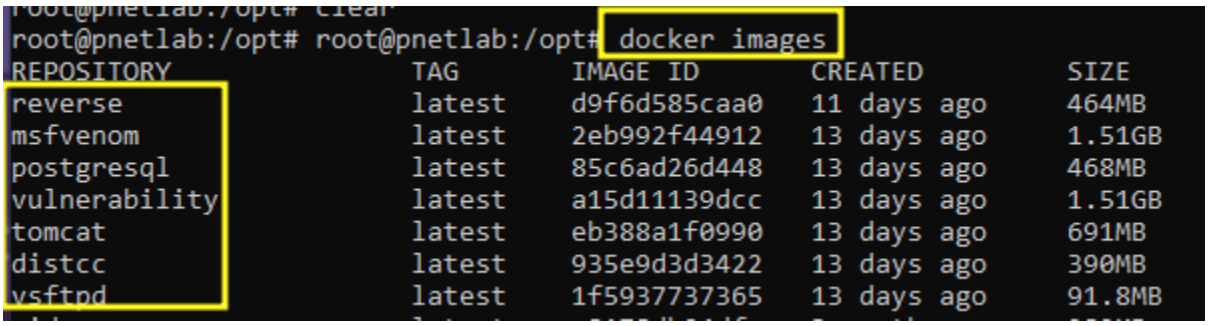
Ahora importamos nuestras maquinas docker:

ls -1 *.tar | xargs --no-run-if-empty -L 1 docker load -i

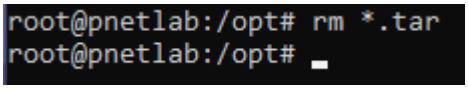


Toca esperar, dependiendo de sus recursos en su server PNetLab que se le haya asignado (CPU, RAM y espacio en disco), tomara su tiempo.

Verificamos que se haya importado correctamente:

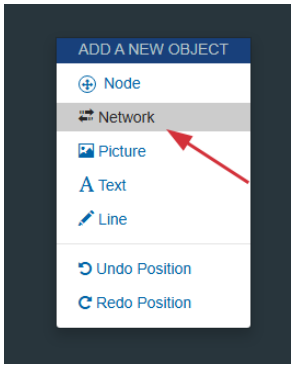


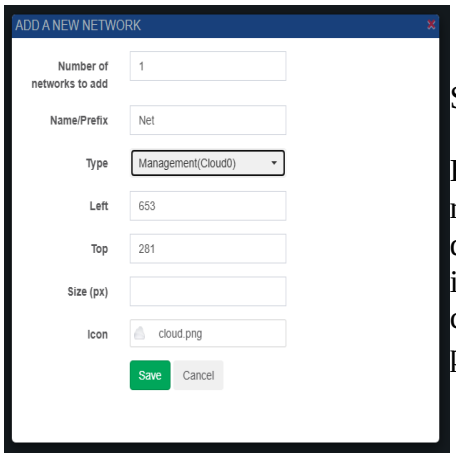
Finalizamos ahorrando espacio, para esto eliminamos los ficheros "**.tar**"



Toca ir a nuestra plataforma de PNetLab y crear un lab, probarlo haciendo un escan de puertos para no extender esta demostración:

Estando en el espacio de trabajo del laboratorio, agregaremos una red que estará conectada a nuestra red interna física, el cual trabajaría como ISP.

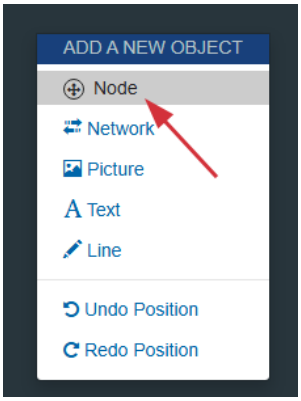


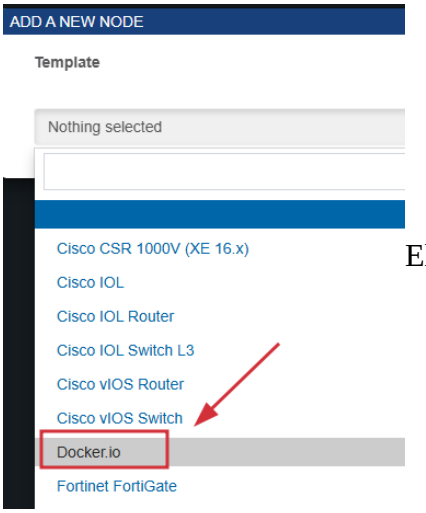


Seleccionamos en ***Type → Management(Cloud0)***, clic en "***Save***"

Este "***Management(Cloud0)***", en la configuración de interfaz de red de la VM de PNetLab, este hace un puente a nuestra red física que alcanza nuestros equipos físicos donde nos dara acceso a internet. Así podemos interactuar con cualquier equipo físico conectado a esta interfaz hacia nuestra máquina docker vulnerable para practica.

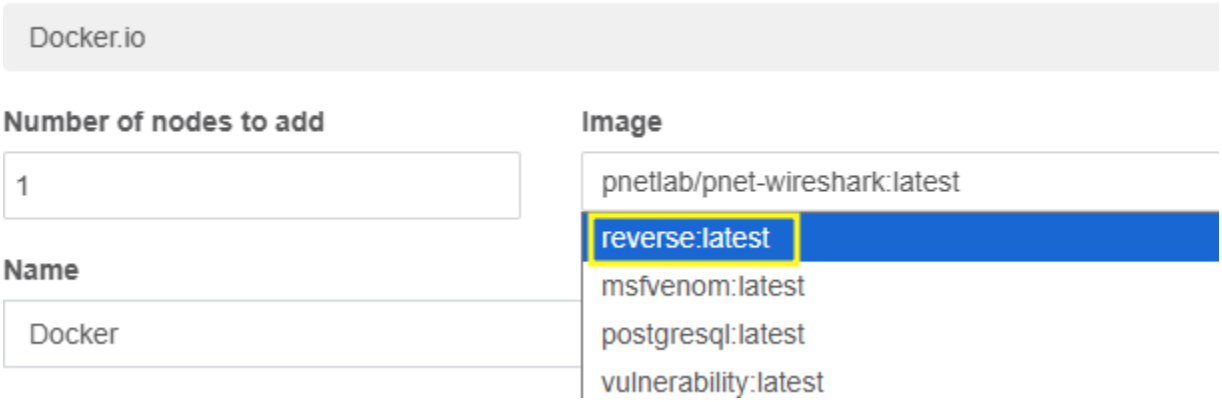
Ahora, toca agregar la máquina docker agregando un nodo:



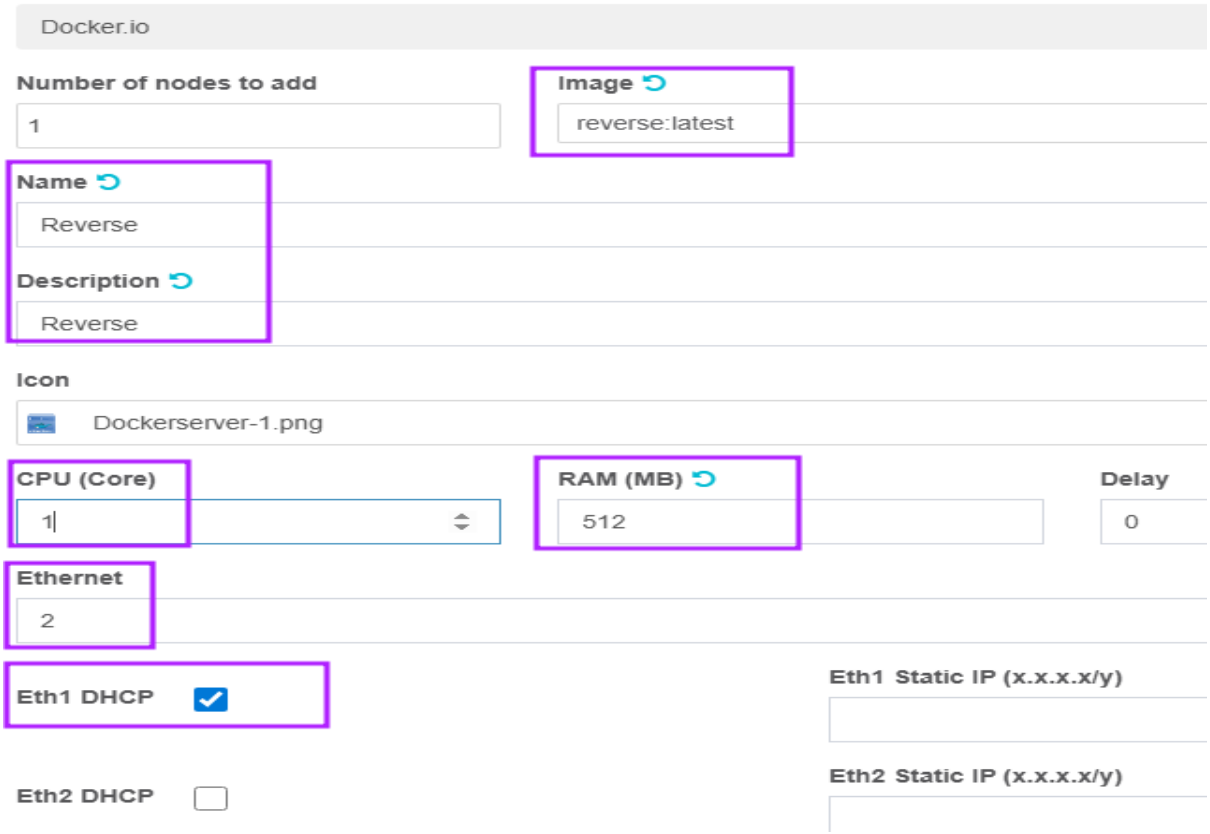


Elegimos "***Docker.io***" :

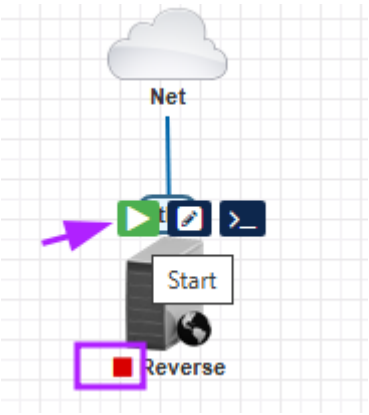
Al final, seleccionamos nuestro contenedor docker "**reverse:latest**", algún nombre, descripción, algún icono, cantidad de RAM, y habilitamos modo DHCP en ETH1":



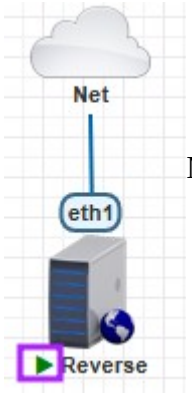
Modificamos lo necesario y ademas dejamos el "**Eth1**" que permita obtener dirección IP vía DHCP de nuestra red donde tenemos nuestro equipos físicos con herramientas de pentesting, por último, ajustamos la consola para esta máquina, el cual debe ser del tipo "**TELNET**" como primario y "**SSH**" como secundario, finalizamos haciendo clic en el botón "**save**" o salvar/guardar:





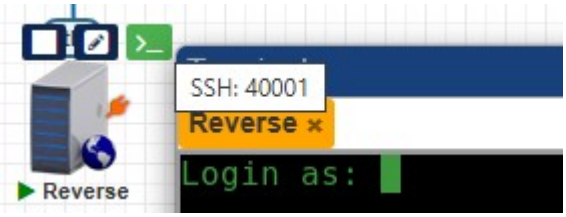


Observara que al hacer clic en el icono de "**play**", toma otro color, en este caso a verde.

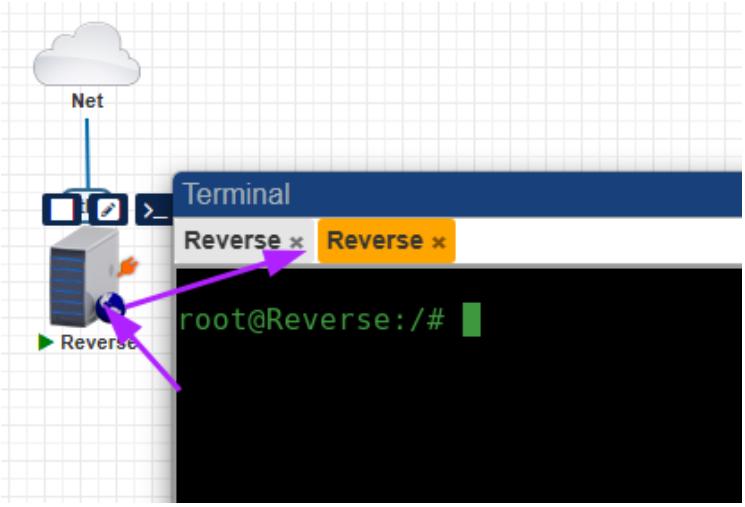


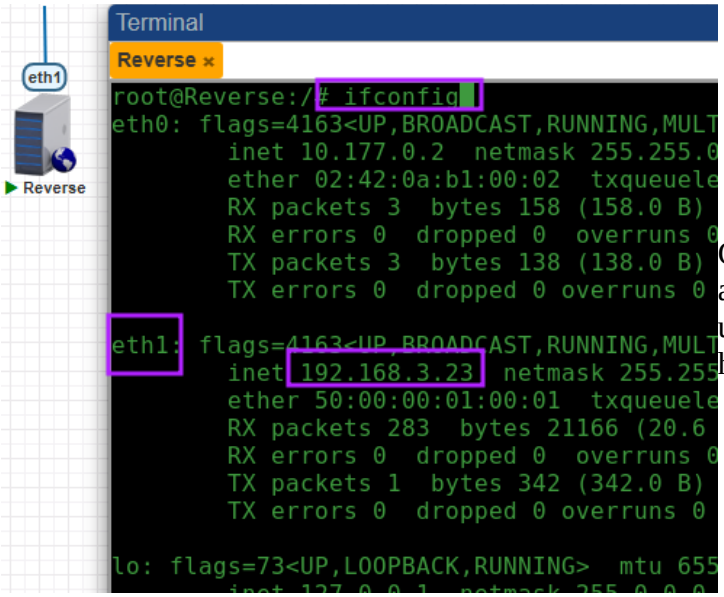
Máquina inicializada, procederemos a verificar.

Para verificar que este inicializada sin errores, pasamos el puntero del mouse sobre el icono y se nos aparece el icono de un prompt de color verde, al hacer clic ahí, nos solicitara usuario y constraseña, pero si hacemos doble clic sobre la máquina docker y hacemos enter, aparecera el prompt sin solicitar usuario y contraseña, o sea, ya logeado, esto para verificar que ip esta utilizando en este momento y empezar a realizar nuestras prácticas.



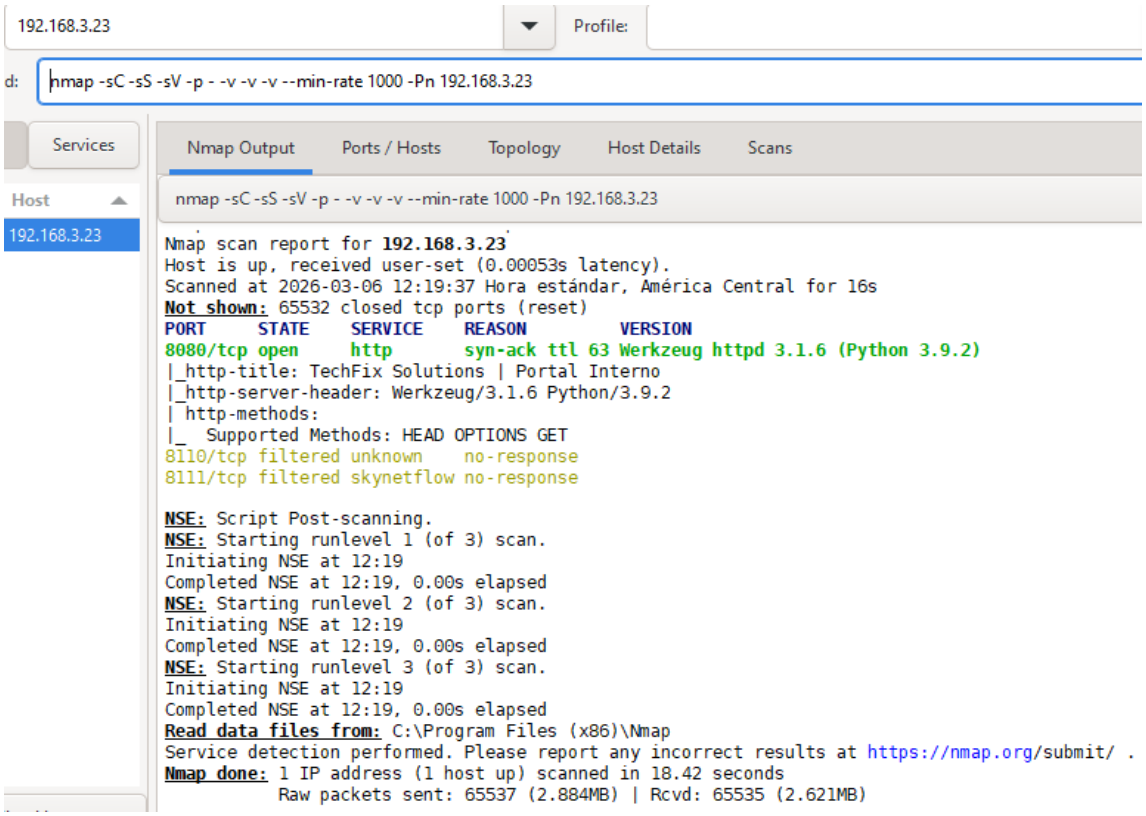
Haciendo doble clic sobre el icono, aparece el promt o shell:

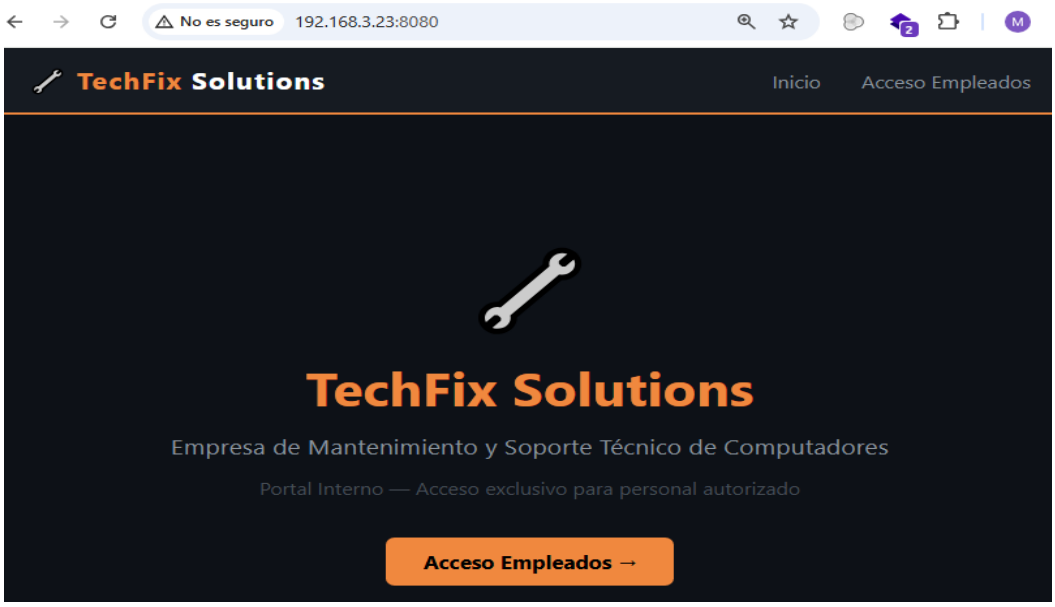




Con doble clic, podemos ver que ip le fue asignado a nuestra maquina para poder realizar una prueba de scan desde nuestro equipo de hacker ético:

Lanzamos nuestro scanner de puerto, ponemos nuestros parámetros de escan y lo lanzamos, podremos ver que encontró servicios activos, con esto se da concluido la demostración de esta máquina, listo para poder realizar prácticas:





Pendiente plataforma https://labs.thehackerslabs.com/.

Fuente:

https://www.pnetlab.com/pages/main
https://www.proxmox.com/en/
https://www.docker.com/
https://hub.docker.com/
https://github.com/ishare2-org/ishare2-cli
https://dockerlabs.es/
https://labs.thehackerslabs.com/
https://whoami-labs.com/