



TECNOLÓGICO NACIONAL DE MÉXICO  
INSTITUTO TECNOLÓGICO DE TLAXIACO

---

## **SEGURIDAD Y VIRTUALIZACIÓN**

Practica 6: Unidad 2 Virtualización - Creación de un laboratorio de seguridad P1

---

### **Integrantes del Equipo:**

Arnol Jesus Cruz Ortiz

Amilkar Vladimir Reyes Reyes

Rael Gabriel Bautista

Sandra Gabriela Velasco Guzmán

### **Docente:**

Edward Osorio Salinas

### **Carrera:**

Ingeniera en Sistemas Computacionales

### **Grupo:** 7US

**Semestre:** Agosto – diciembre 2024

24/Octubre/2024

## PRACTICA 6: CREACIÓN DE UN LABORATORIO DE SEGURIDAD P1

### Índice

<b>OBJETIVO.....</b>	3
<b>INTRODUCCIÓN.....</b>	3
<b>INSTALAR OPNSENSE EN UNA MÁQUINA VIRTUAL Y CONFIGURAR UN FIREWALL.....</b>	4
<b>Creación de la máquina virtual para OpnSense.....</b>	4
<b>Instalación y configuración del OpnSense .....</b>	7
<b>Configuración de interfaces de red .....</b>	12
<b>INSTALAR KALI LINUX EN UNA MÁQUINA VIRTUAL Y CONFIGURAR UN SISTEMA DE DETECCIÓN DE INTRUSOS.....</b>	14
<b>Descarga de Kali Linux.....</b>	14
<b>Creación de maquina virtual para Kali Linux.....</b>	14
<b>Instalación y configuración de Kali Linux .....</b>	16
<b>INSTALAR UN SISTEMA DE DETECCIÓN DE INTRUSOS COMO SURICATA.....</b>	28
<b>Instalación de del sistema de detección de intrusos suricata.....</b>	28
<b>Configuración de reglas de detección de intrusos .....</b>	31
<b>Asignar una dirección IP estática a Kali Linux.....</b>	37
<b>Habilitar la configuración en NetworkManager .....</b>	39
<b>CREAR UNA MÁQUINA VIRTUAL VULNERABLE POR DISEÑO COMO METASPLOITABLE2.....</b>	40
<b>Descarga de MetaSploitable2.....</b>	40
<b>Creación de maquina virtual para MetaSploitable2 .....</b>	40
<b>Inicio de MetaPloitble2 y asignación de ip estatica.....</b>	40
<b>PING SATISFACTORIO ENTRE LAS MÁQUINAS VIRTUALES.....</b>	43
<b>Configuración de interfaz de red de MetaSploitable2.....</b>	43
<b>Configuración de reglas del firewall para permitir el trafico de ping .....</b>	44
<b>Configuración de interfaces de red De Kali Linux.....</b>	45
<b>Configuración de reglas para permitir el ping en Kali Linux.....</b>	46
<b>REALIZACIÓN DE PING ENTRE LAS MÁQUINAS VIRTUALES.....</b>	47
<b>Realización de ping desde MetaSplitables2 a Kali Linux .....</b>	47
<b>Realización de ping desde Kali Linux a MetaSplitables2 .....</b>	47
<b>CONCLUSIÓN.....</b>	48

## **OBJETIVO**

Implementar un laboratorio de seguridad en VirtualBox o VMWare con las siguientes características:

- Una máquina virtual con OpnSense o pfSense configurada como firewall.
- Una máquina virtual con Kali Linux configurada como sistema de detección de intrusos.
- Una máquina virtual vulnerable por diseño como MetaSploitable2.
- Ping satisfactorio entre las máquinas virtuales.

## **INTRODUCCIÓN**

Lo que realizaremos en esta práctica es crear un entorno controlado de seguridad de red mediante la instalación y configuración de diversas herramientas de red virtualizadas. Este entorno permitirá simular un sistema de red protegido y analizar el comportamiento ante amenazas ciberneticas utilizando soluciones de firewall, detección de intrusos, y simulación de máquinas vulnerables.

Para comenzar, se instalará una plataforma de virtualización como VirtualBox, lo que permitirá crear y gestionar máquinas virtuales (VM) de manera eficiente. Esto proporciona un entorno aislado y seguro donde se podrán probar diferentes configuraciones y servicios sin comprometer la infraestructura real. La instalación de estas plataformas es el primer paso para asegurar que todas las máquinas y servicios necesarios se ejecuten de manera independiente.

El siguiente paso será la instalación de Kali Linux, una distribución especializada en pruebas de penetración y análisis de seguridad. Kali Linux se instalará en una máquina virtual y se configurará para funcionar como un sistema de detección de intrusos (IDS). Esto se logrará mediante la instalación de herramientas de seguridad avanzadas como Snort o Suricata siguiendo pasos apropiados para su configuración.

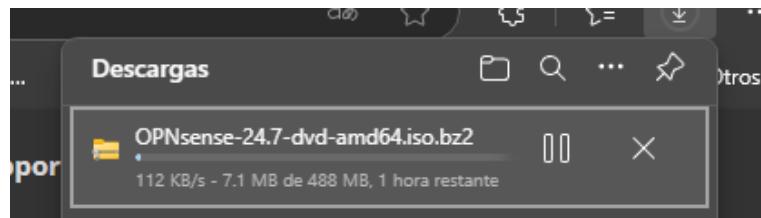
Después instalaremos y configuraremos MetaSploitable2 para simular un entorno vulnerable que permita realizar pruebas de penetración.

Y por último configuraremos la conectividad y realizaremos pruebas de seguridad así como ping entre dos máquinas virtuales.

## “INICIO DE PRACTICA”

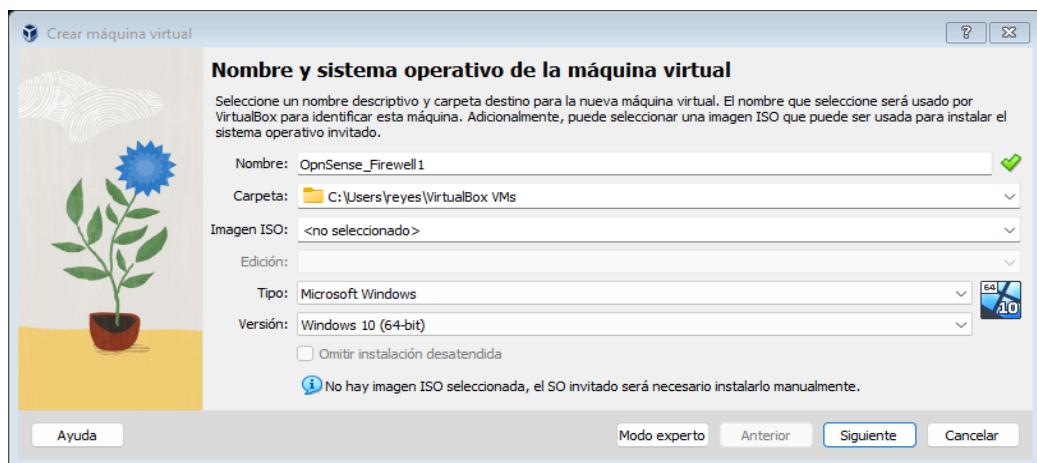
### INSTALAR OPNSENSE EN UNA MÁQUINA VIRTUAL Y CONFIGURAR UN FIREWALL.

Descargar la imagen iso de OpnSense

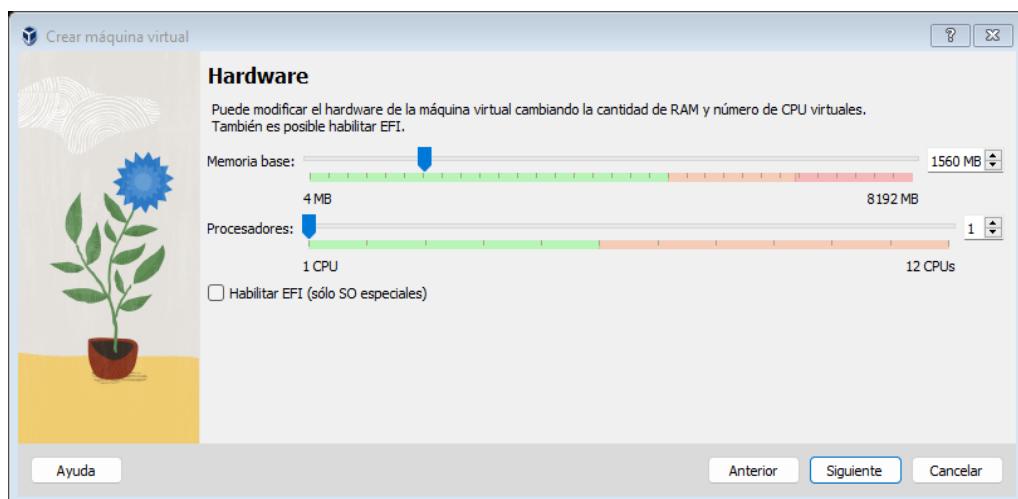


#### Creación de la máquina virtual para OpnSense

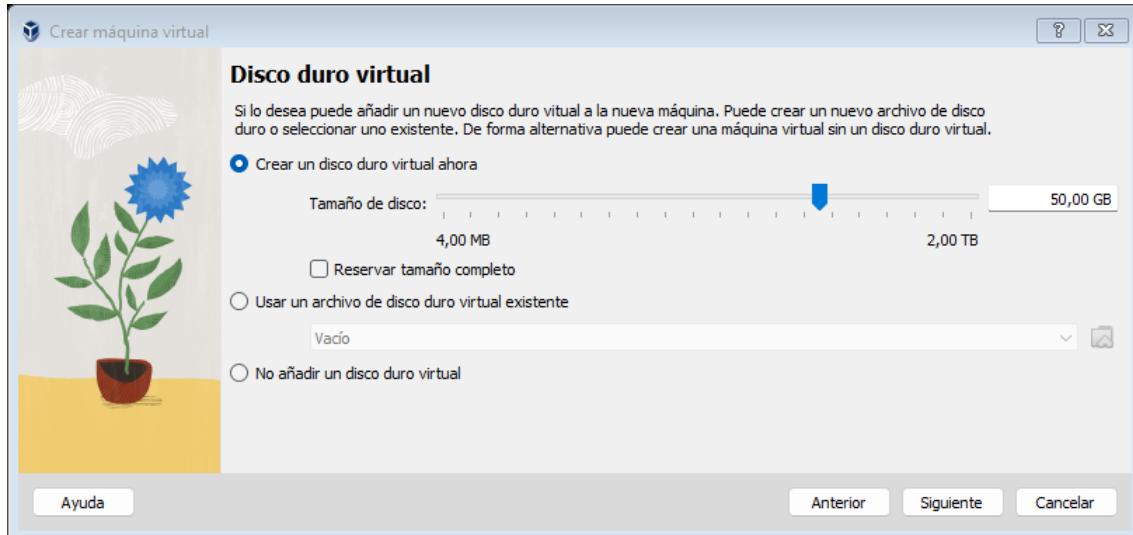
Para instalar OpnSense debemos de crear primero una nueva maquina virtual con el nombre de OpnSense firewall, ahí mismo dirigimos hacia la carpeta en donde deseamos que se guarde.



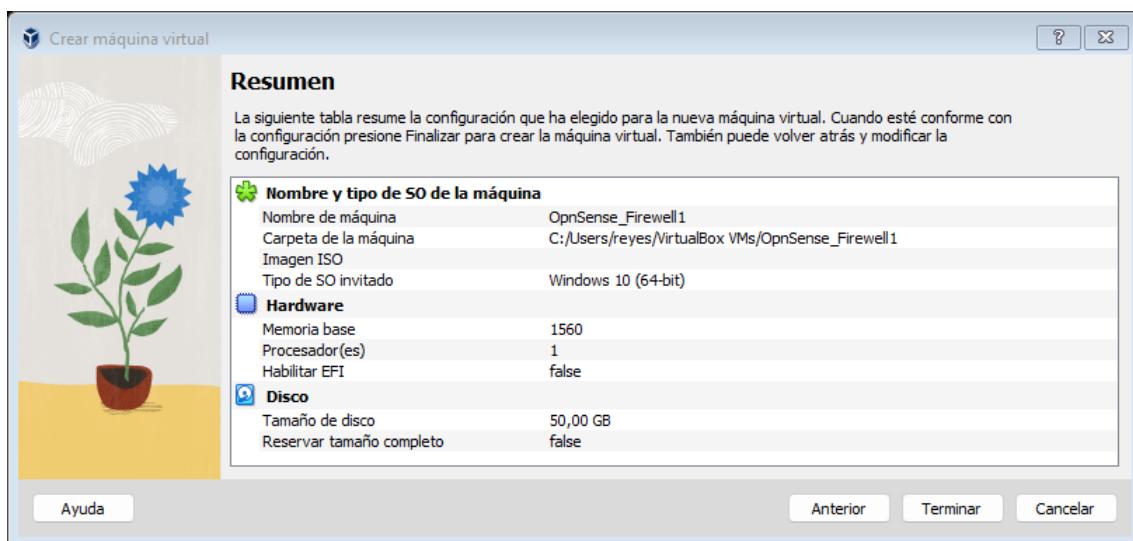
Le damos 4 gb de RAM para que dé un mejor funcionamiento



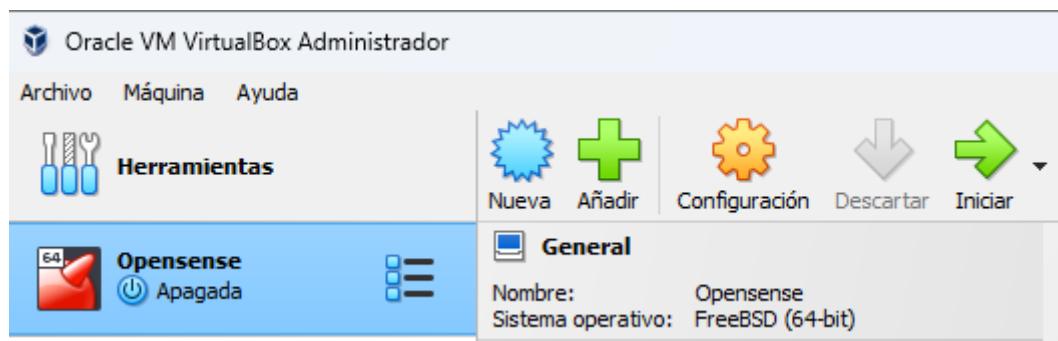
En esta parte se deja por defecto ya que lo que buscamos es crear un disco virtual por ahora.



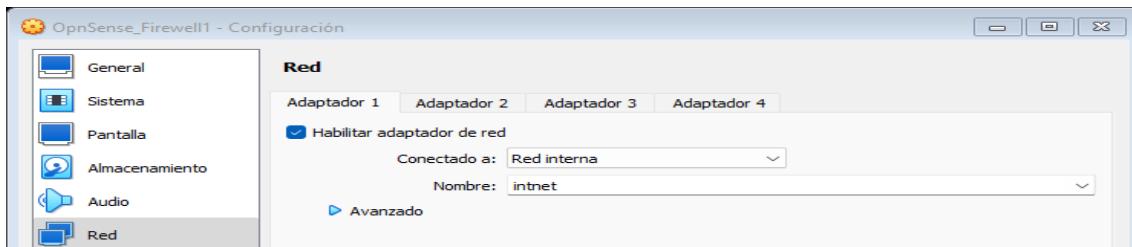
En esta parte nos muestra el nombre y tipos de así que utilizara la máquina virtual, lo único que hacemos es dar clic en terminar.



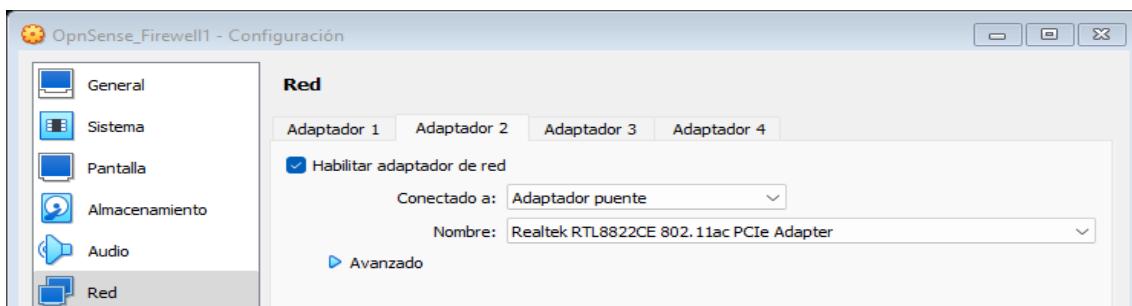
Y así es como se crearía nuestra nueva maquina virtual Opensense.



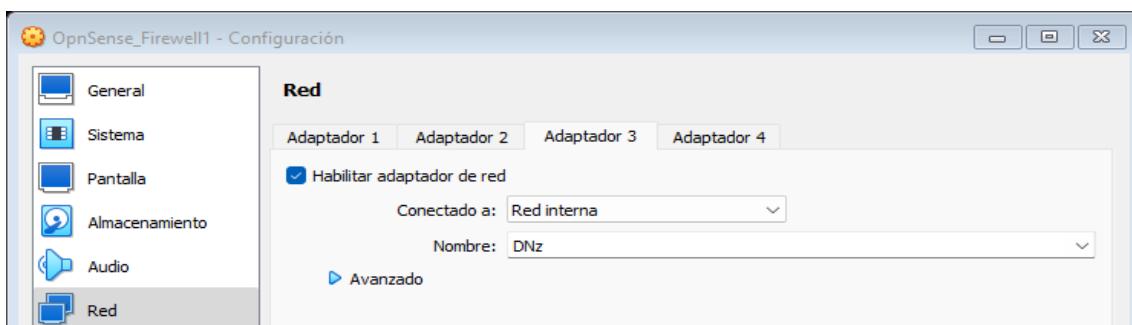
Ahora habilitaremos la tarjeta de red 1 va a ser la interna, la LAN en este caso y luego se pone en modo puente el adaptador para que tome la ip y si queremos tener las cuatro tarjetas de red vamos a una red interna y ahí vemos otra red interna que será la intermedia.



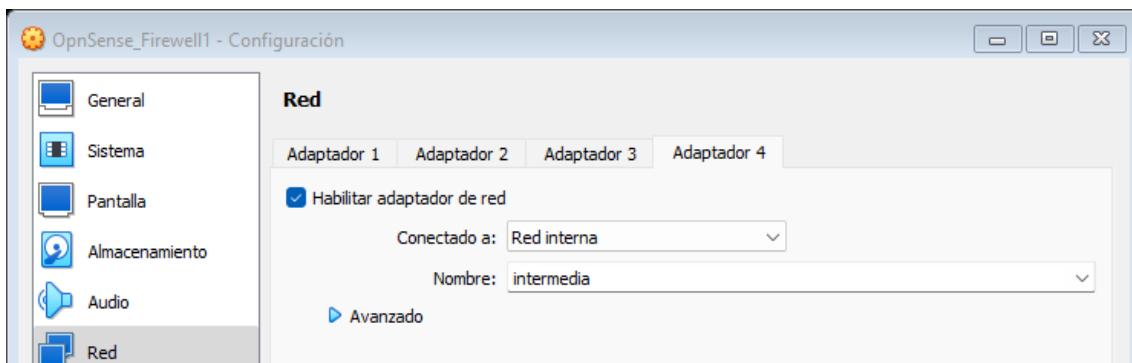
Adaptador puente número dos.



Adaptador Red interna con nombre DNZ número tres

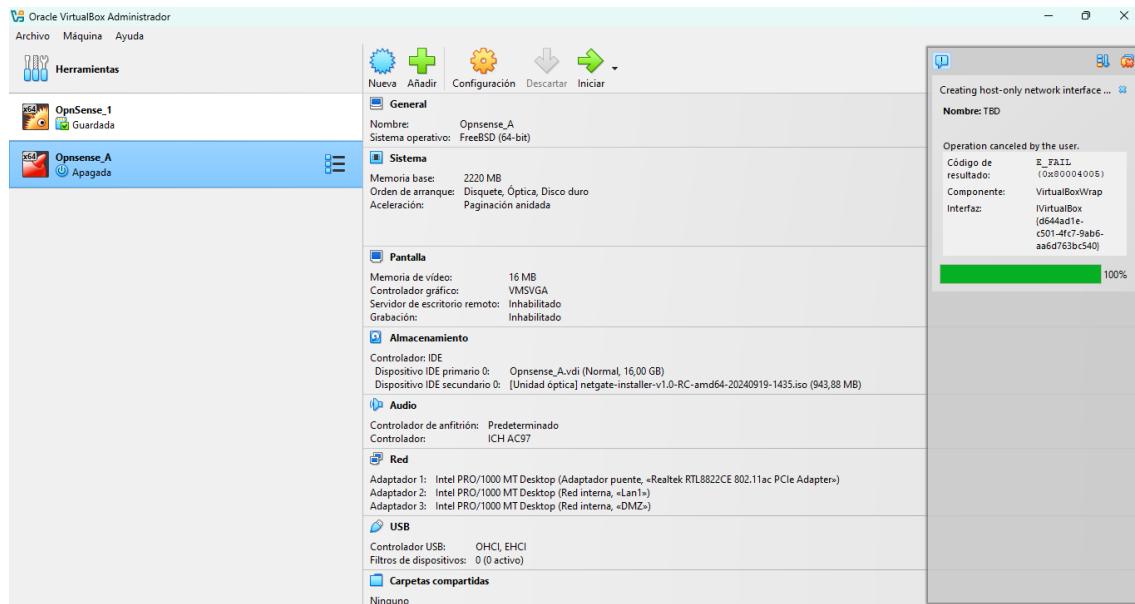


Adaptador red interna con nombre intermedia numero cuatro.

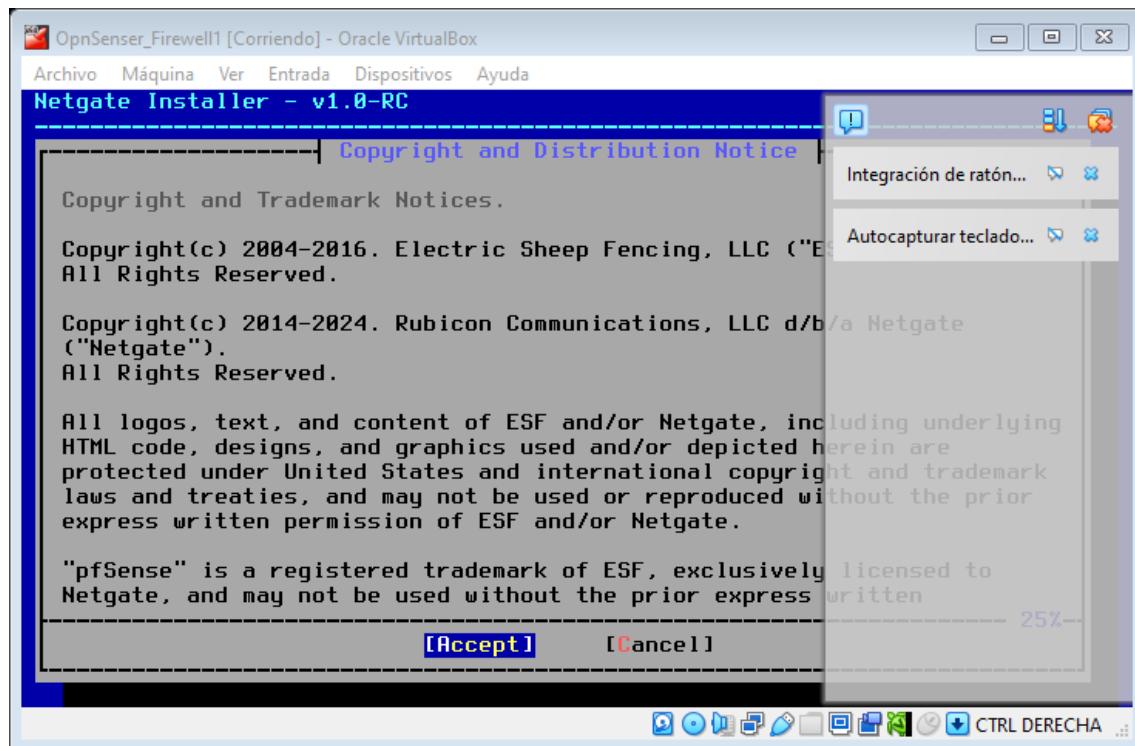


## Instalación y configuración del OpnSense

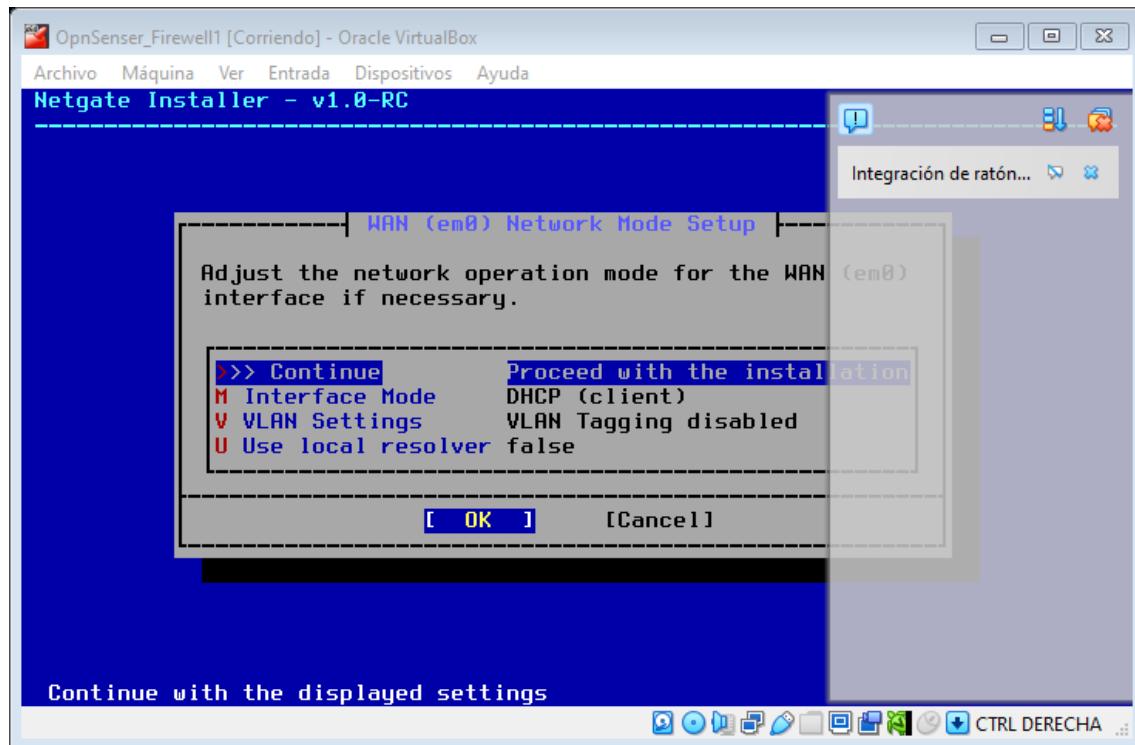
Aquí ya esta creada la maquina virtual con su respectiva configuración de red.



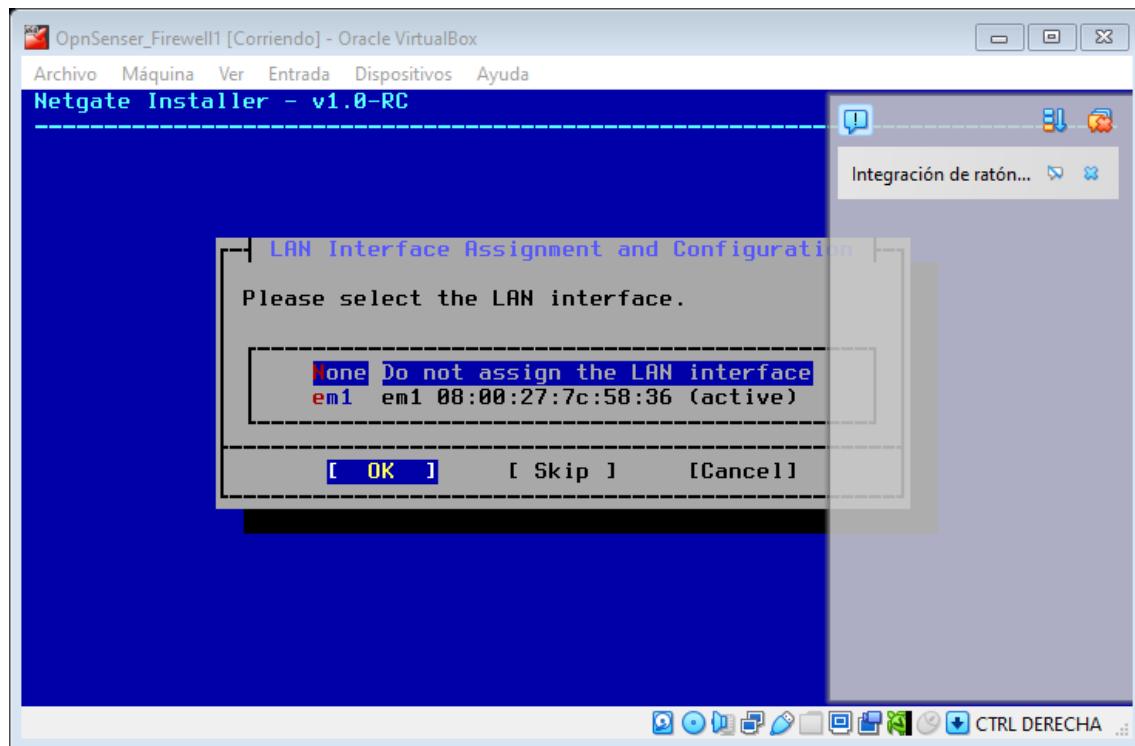
Ya que terminamos lo anterior iniciamos la maquina y nos pedirá que pongamos el disco que se utilizara se le da en aceptar.



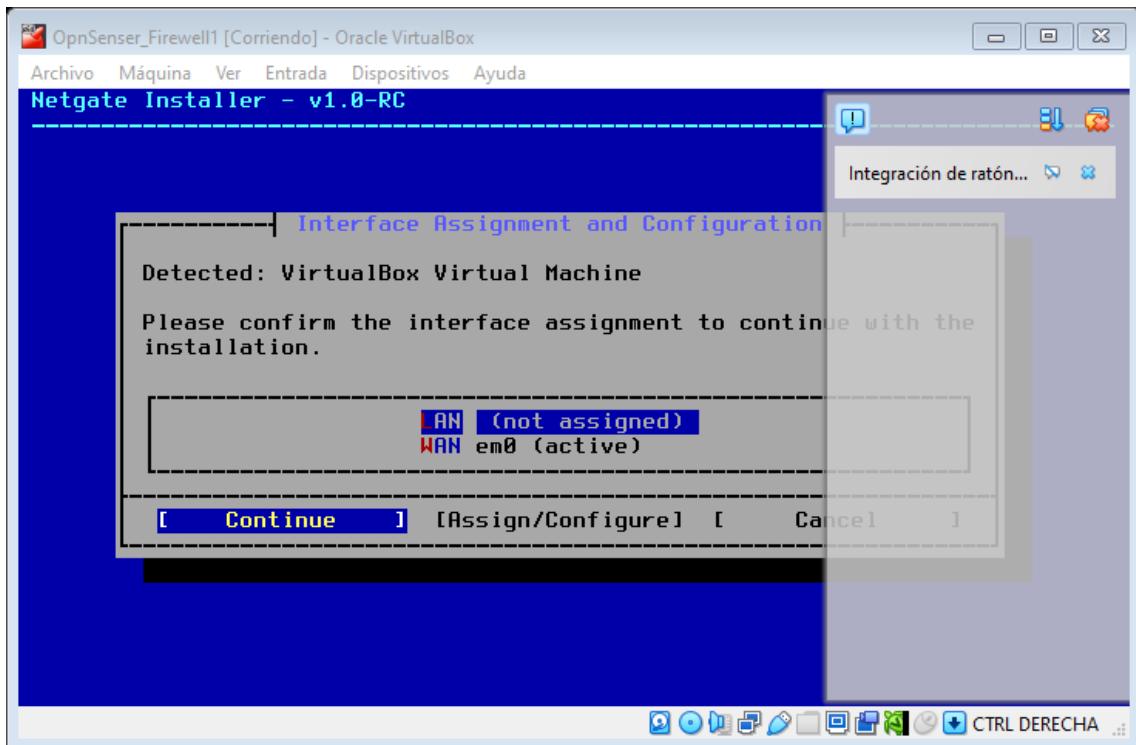
Después le damos en instalar PFSENSE Y CONTINUAR



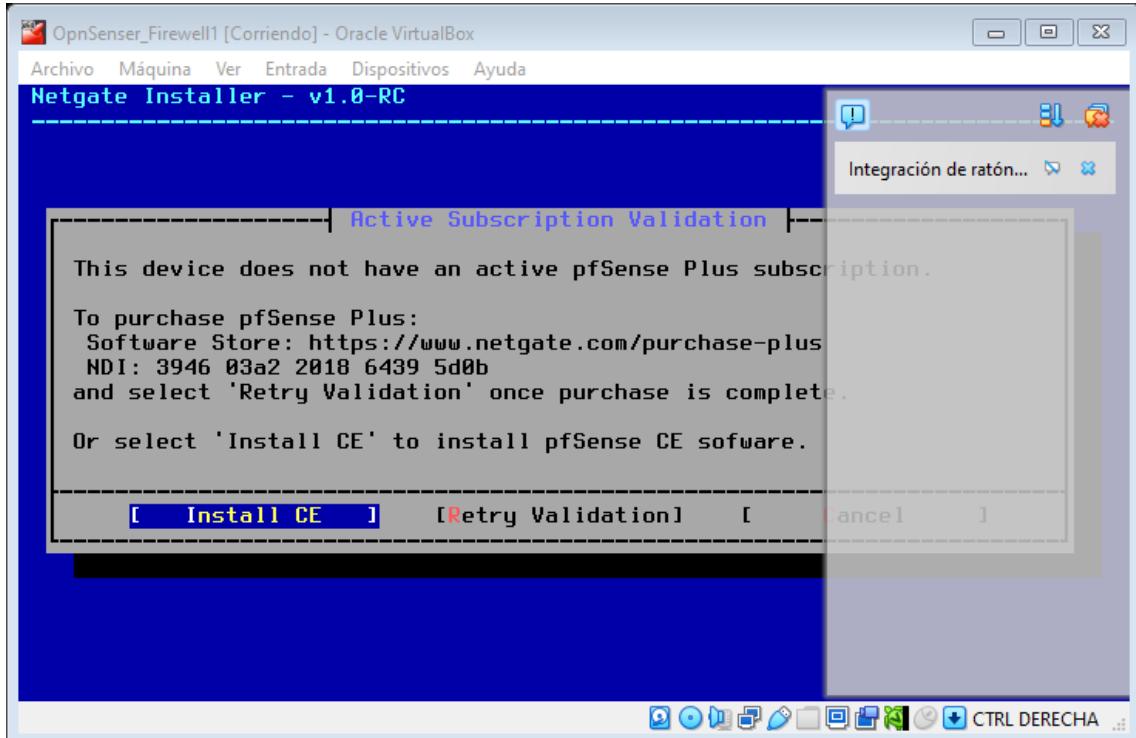
Aquí debemos de seleccionar la interfaz de la LAN



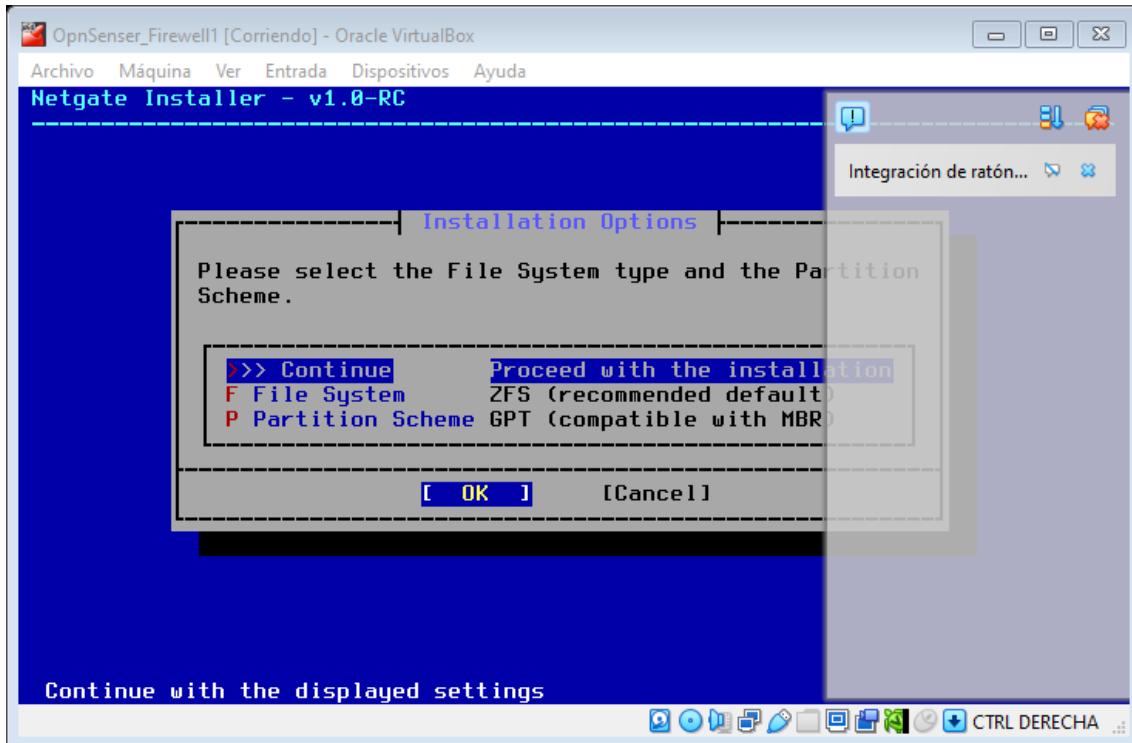
Volvemos a confirmar la interfaz que elegimos y continuamos con la instalación



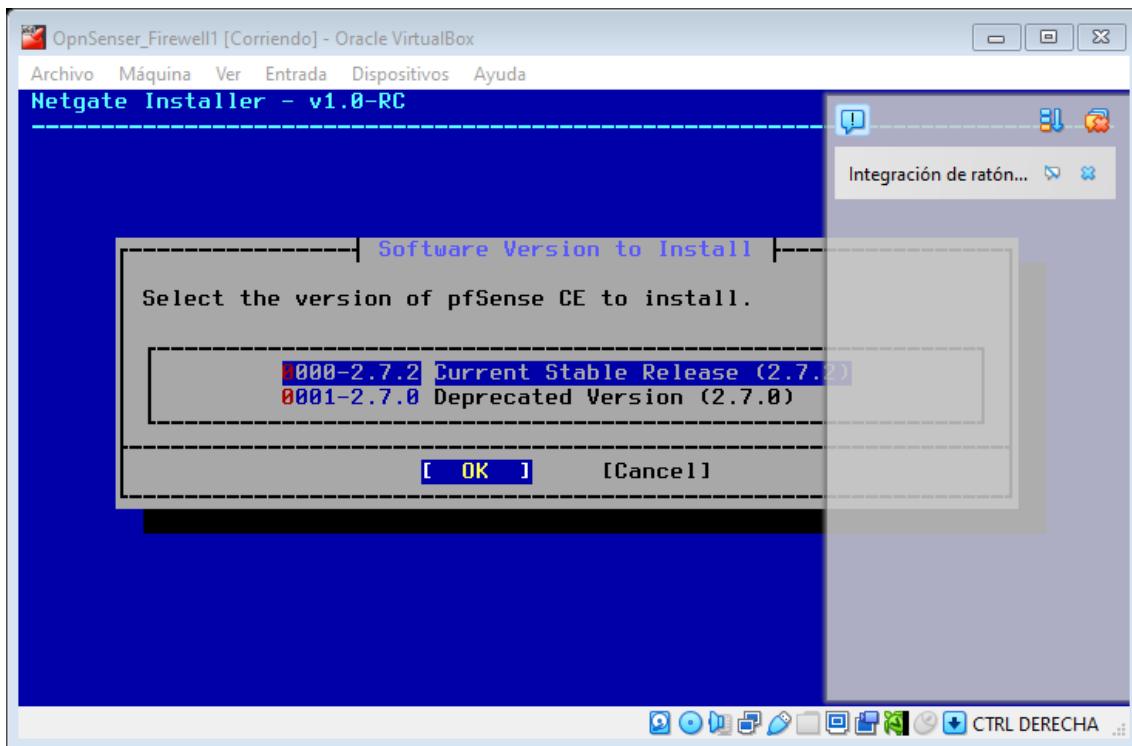
Le damos clic en Install CE



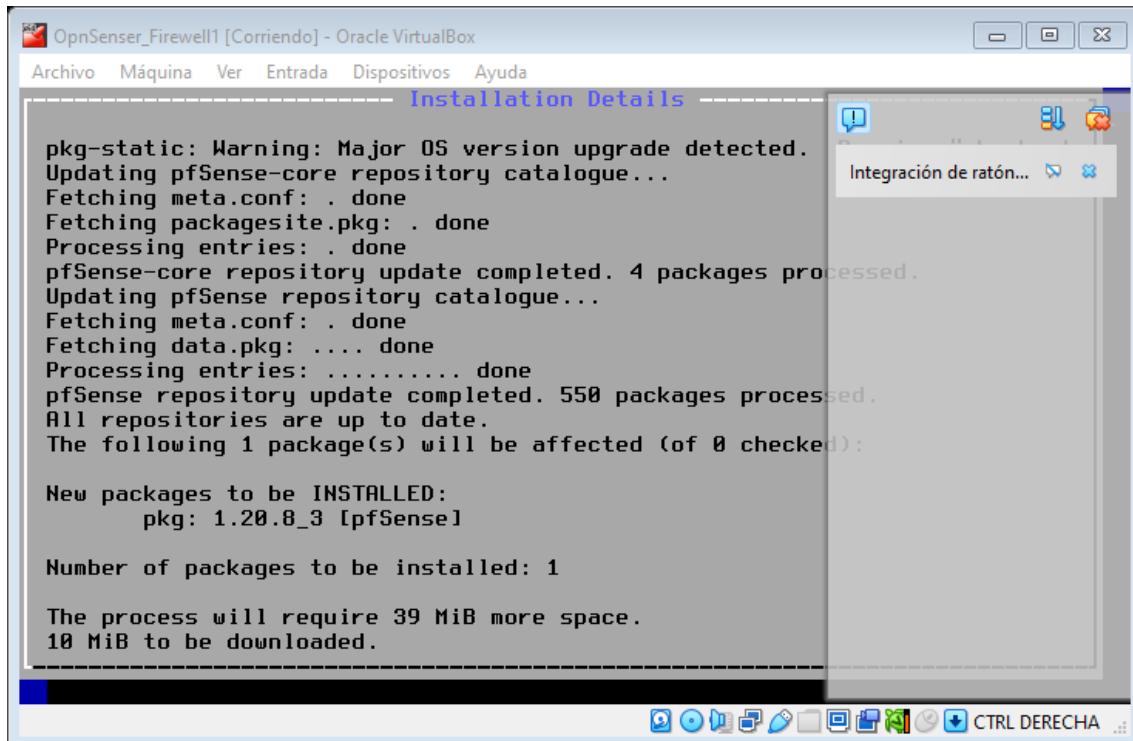
Seleccionamos el proceso para poder instalarlo



Seleccionamos la versión del OpenSense y le damos en ok



Aquí está en proceso el Will ya que requiere de mib



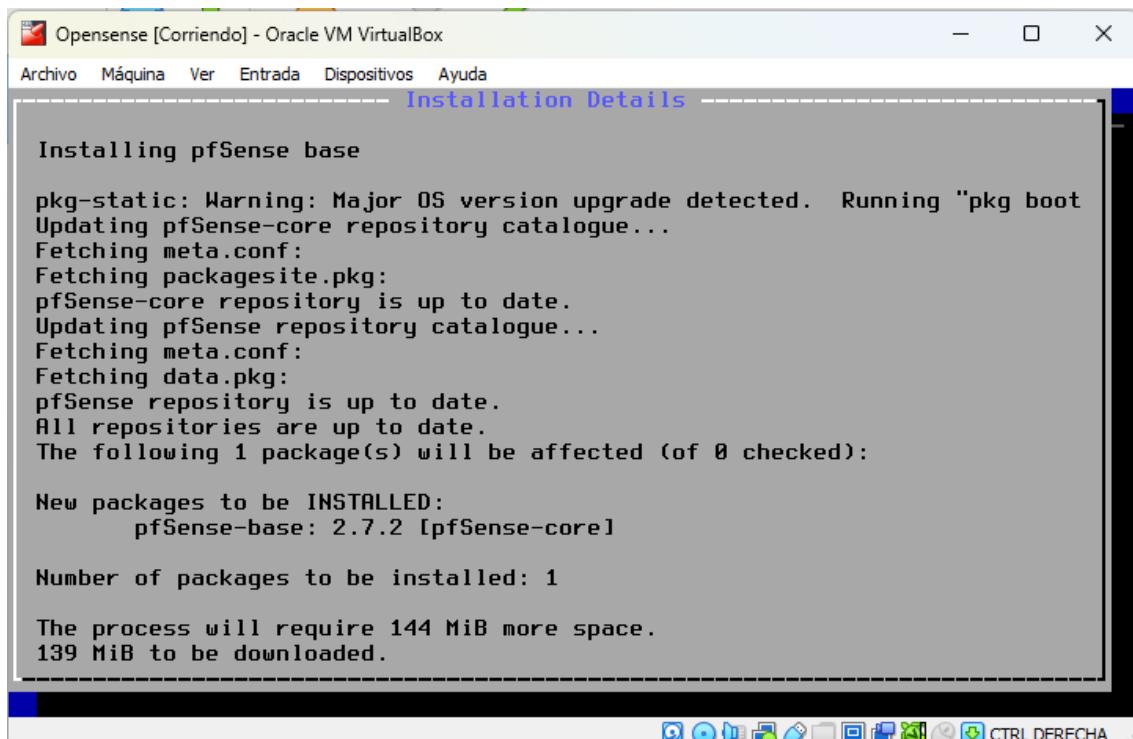
```
pkg-static: Warning: Major OS version upgrade detected.
Updating pfSense-core repository catalogue...
Fetching meta.conf: . done
Fetching packagesite.pkg: . done
Processing entries: . done
pfSense-core repository update completed. 4 packages processed.
Updating pfSense repository catalogue...
Fetching meta.conf: . done
Fetching data.pkg: .... done
Processing entries: ..... done
pfSense repository update completed. 550 packages processed.
All repositories are up to date.
The following 1 package(s) will be affected (of 0 checked):

New packages to be INSTALLED:
    pkg: 1.20.8_3 [pfSense]

Number of packages to be installed: 1

The process will require 39 MiB more space.
10 MiB to be downloaded.
```

Instalación de la fase de opensense



```
Installing pfSense base

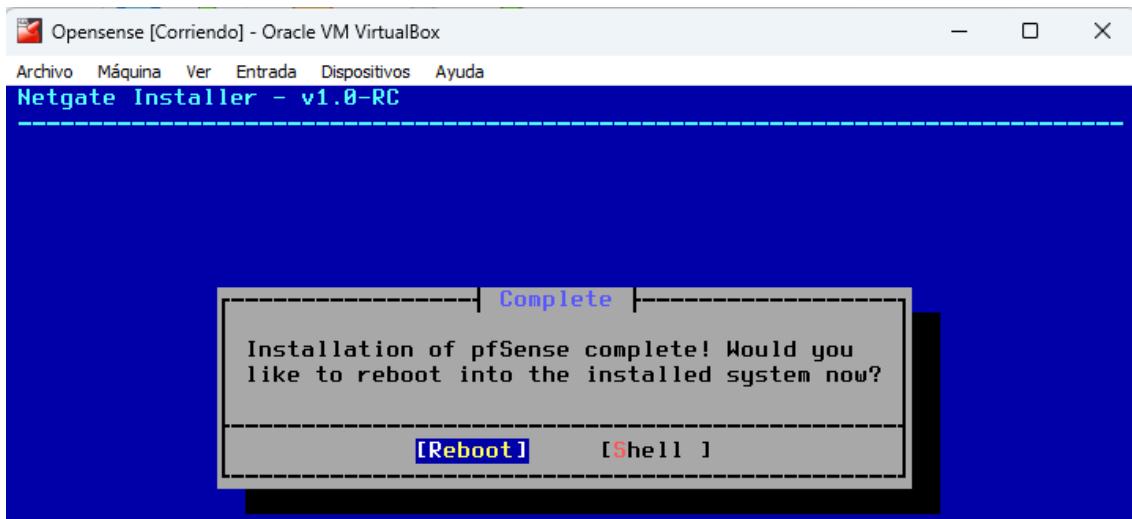
pkg-static: Warning: Major OS version upgrade detected. Running "pkg boot"
Updating pfSense-core repository catalogue...
Fetching meta.conf:
Fetching packagesite.pkg:
pfSense-core repository is up to date.
Updating pfSense repository catalogue...
Fetching meta.conf:
Fetching data.pkg:
pfSense repository is up to date.
All repositories are up to date.
The following 1 package(s) will be affected (of 0 checked):

New packages to be INSTALLED:
    pfSense-base: 2.7.2 [pfSense-core]

Number of packages to be installed: 1

The process will require 144 MiB more space.
139 MiB to be downloaded.
```

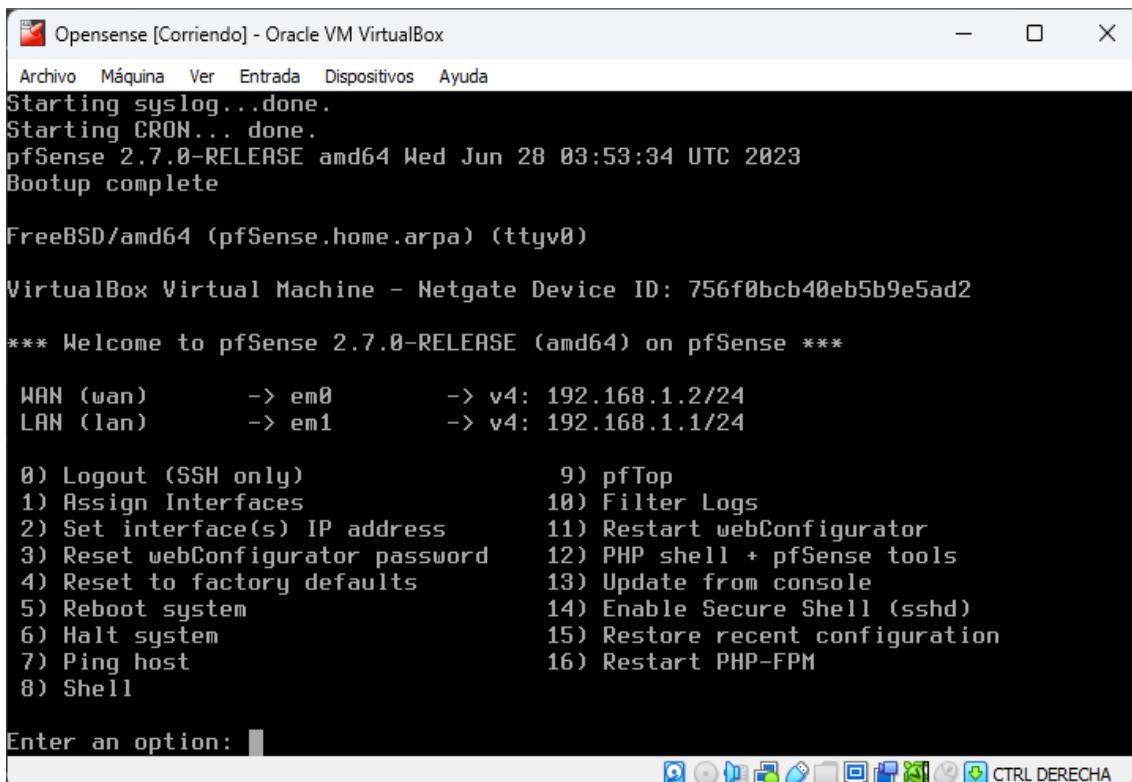
Aquí nos dice que ya se ha instalado correctamente



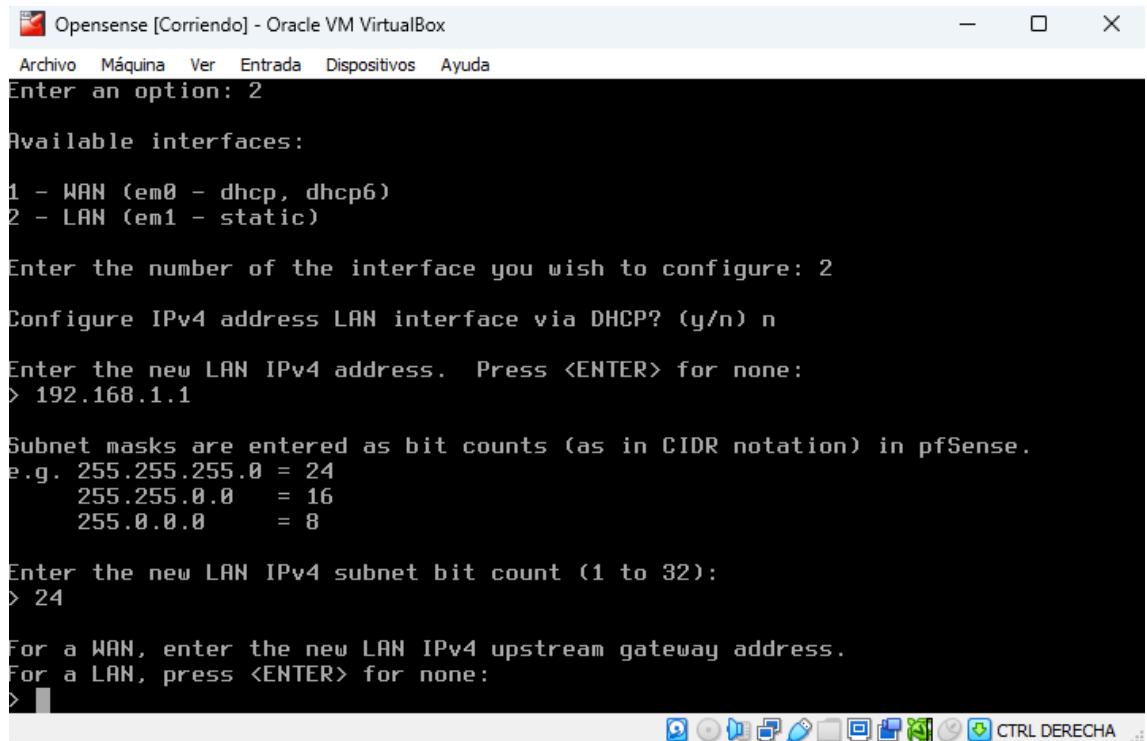
## Configuración de interfaces de red

Ahora quitamos la imagen iso y después lo reiniciamos

Y nos muestra el menú de la consola y muestra 13 opciones.



Elegimos la opción de asignar interfaces y las modificamos dependiendo a nuestra ip de la red que estemos utilizando.



```
Opensense [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Enter an option: 2

Available interfaces:
1 - WAN (em0 - dhcp, dhcp6)
2 - LAN (em1 - static)

Enter the number of the interface you wish to configure: 2

Configure IPv4 address LAN interface via DHCP? (y/n) n

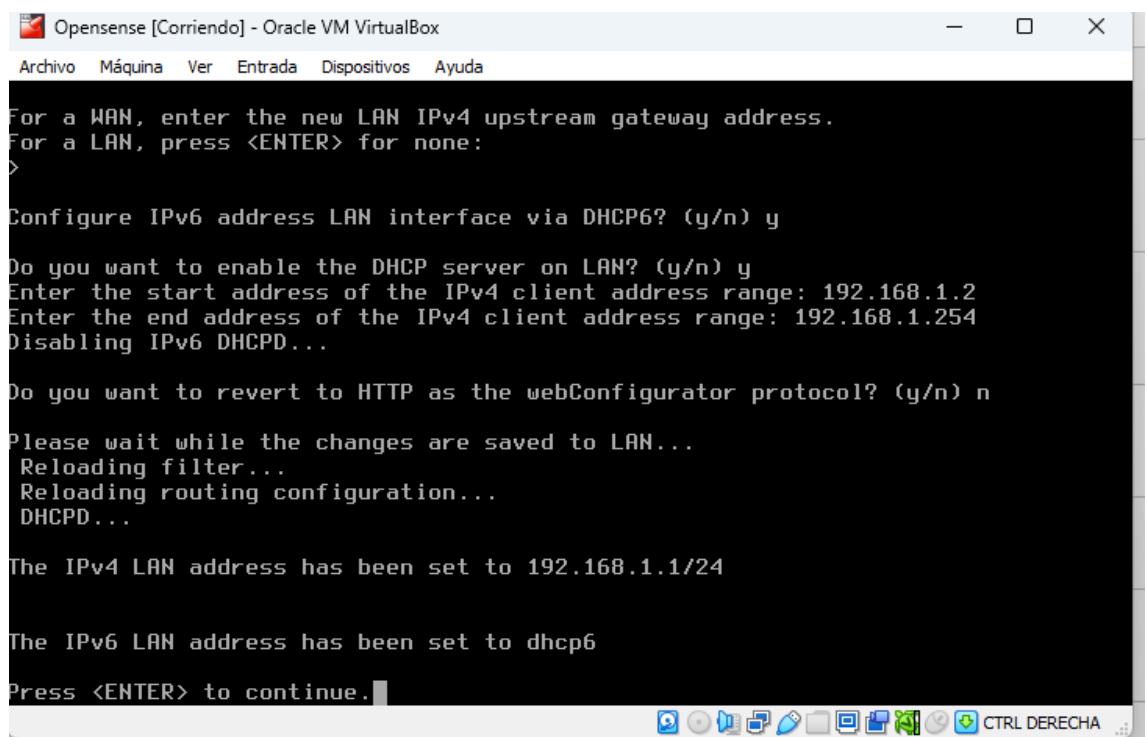
Enter the new LAN IPv4 address. Press <ENTER> for none:
> 192.168.1.1

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0     = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24

For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
> 
```

Aquí nos muestra que ya asignamos las direcciones ip



```
Opensense [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
>

Configure IPv6 address LAN interface via DHCP6? (y/n) y

Do you want to enable the DHCP server on LAN? (y/n) y
Enter the start address of the IPv4 client address range: 192.168.1.2
Enter the end address of the IPv4 client address range: 192.168.1.254
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) n

Please wait while the changes are saved to LAN...
  Reloading filter...
  Reloading routing configuration...
  DHCPD...

The IPv4 LAN address has been set to 192.168.1.1/24

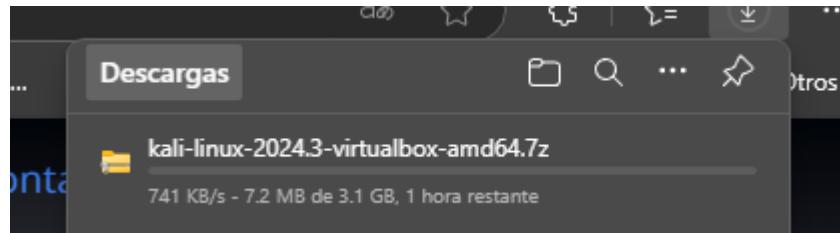
The IPv6 LAN address has been set to dhcp6

Press <ENTER> to continue. 
```

# INSTALAR KALI LINUX EN UNA MÁQUINA VIRTUAL Y CONFIGURAR UN SISTEMA DE DETECCIÓN DE INTRUSOS.

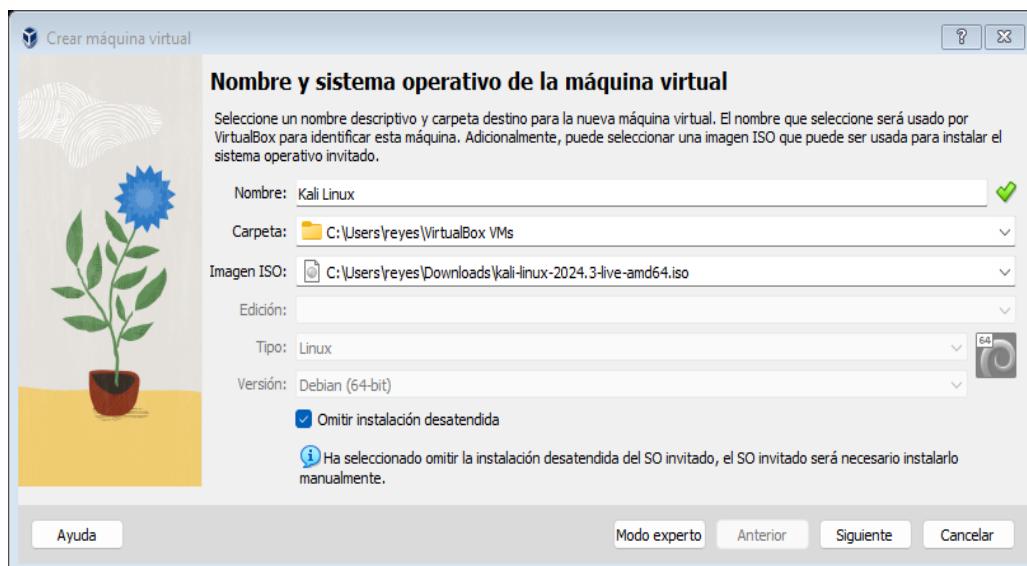
## Descarga de Kali Linux

Ahora realizaremos la descarga de Kali, para eso nos vamos al navegador de nuestra preferencia y buscamos Kali Linux.

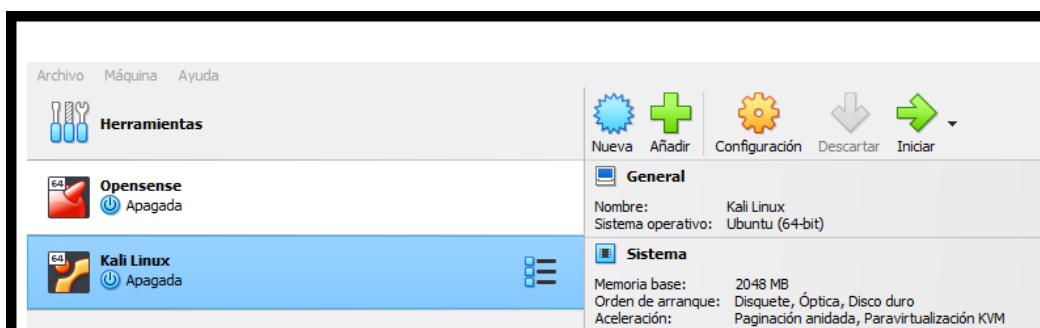


## Creación de maquina virtual para Kali Linux

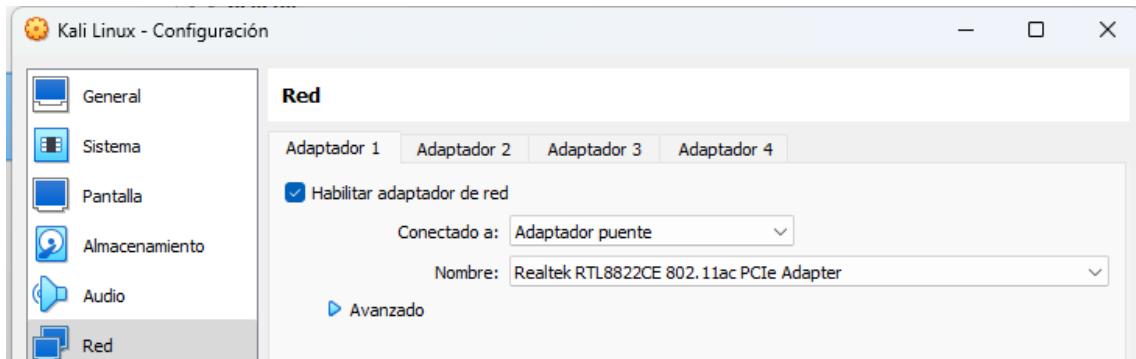
Ahora instalaremos Kali Linux en VirtualBox para ello realizamos una nueva maquina virtual llamada Kali Linux.



Aquí ya esta creada para poder iniciar la maquina virtual



Después nos vamos a su configuración y en la parte de red vamos asignar el adaptador puente

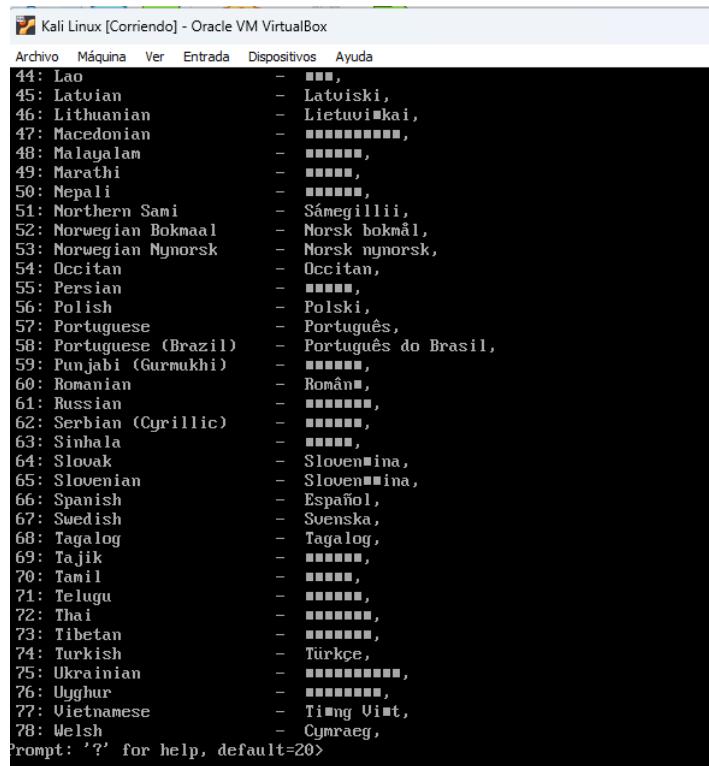


Después en almacenamiento validamos que este el disco con la ISO de Kali y aplicamos cambios

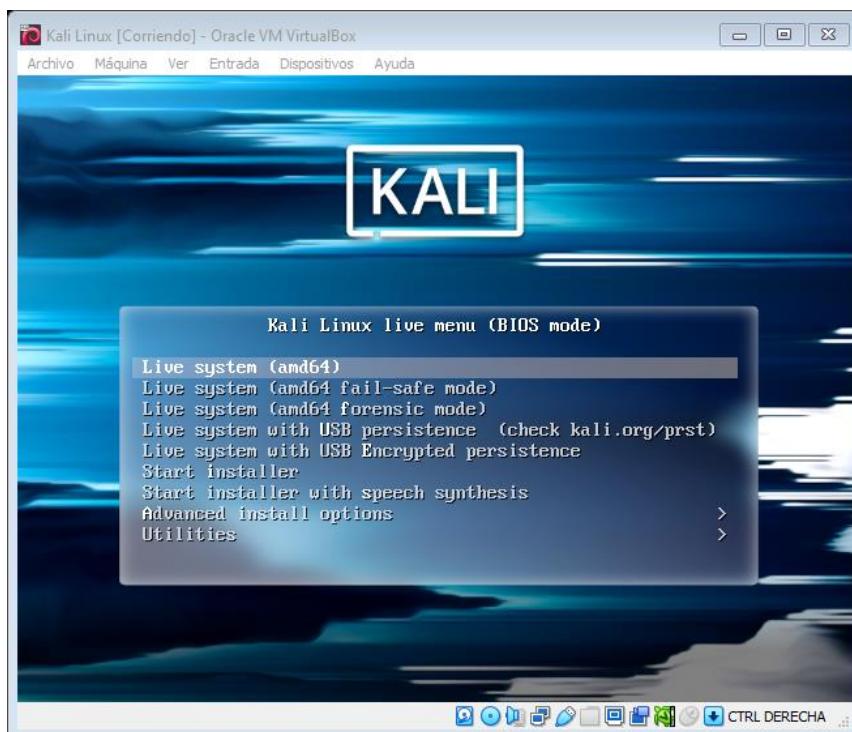


# Instalación y configuración de Kali Linux

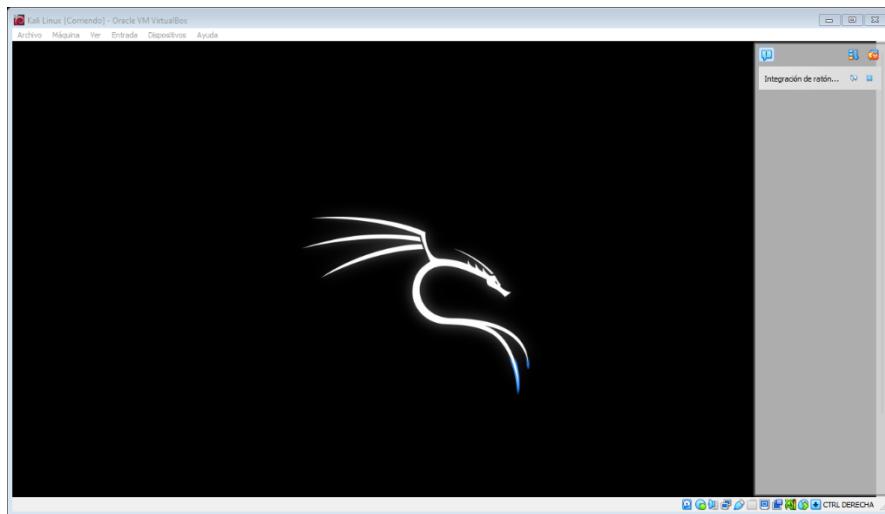
Después le damos en iniciar para arrancar la máquina virtual de Kali Linux



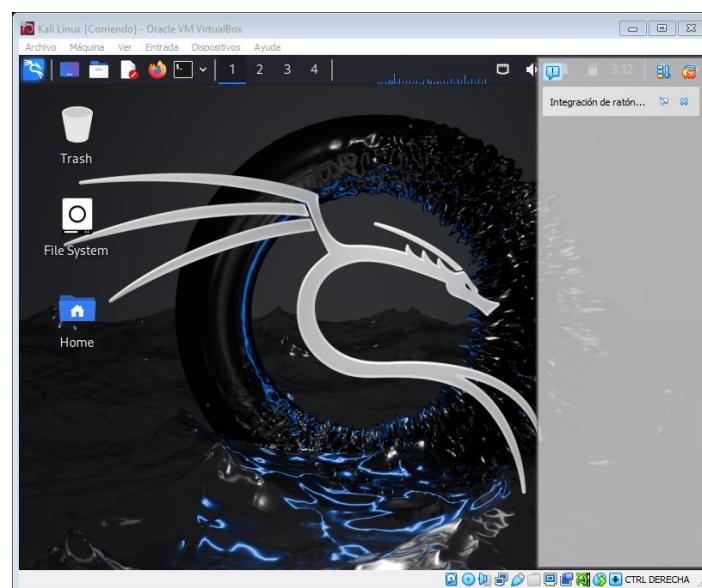
De ahí nos aparecerá una pantalla en donde viene un menú y le damos en donde dice Live System para conocer el sistema antes de instalarlo,



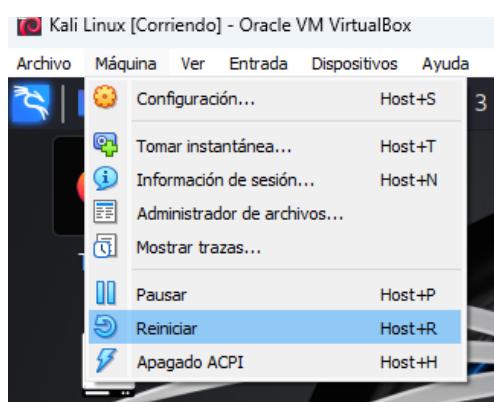
Esperamos a que cargue



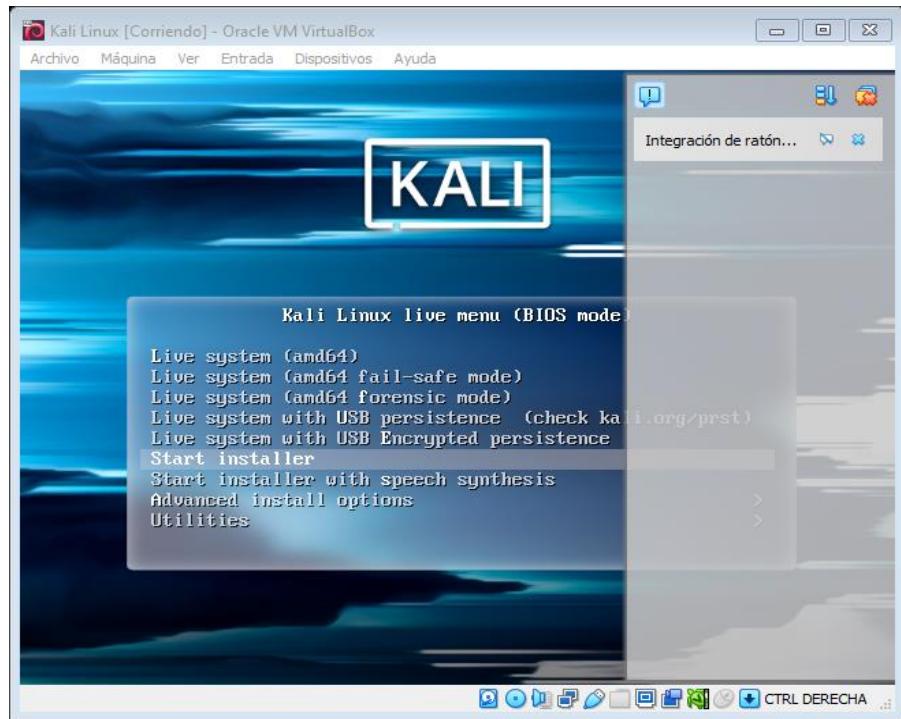
Ahí es posible usar el sistema sin ser instalado, pero todos los cambios se perderán, entonces le damos clic en maquina y reiniciar



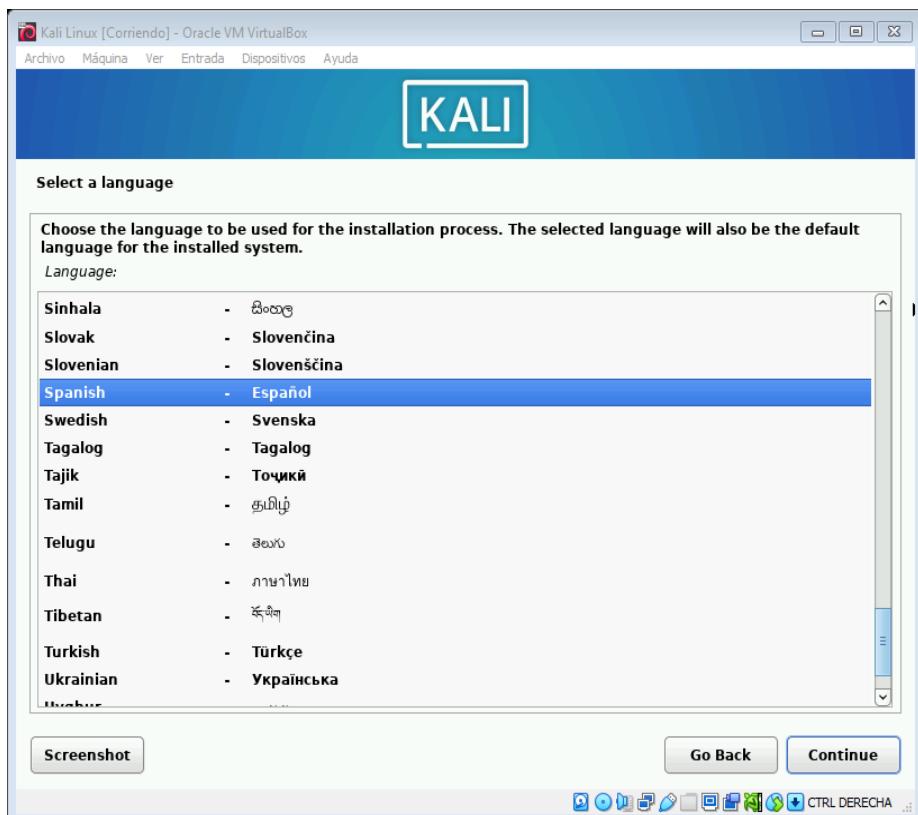
Le damos en maquina



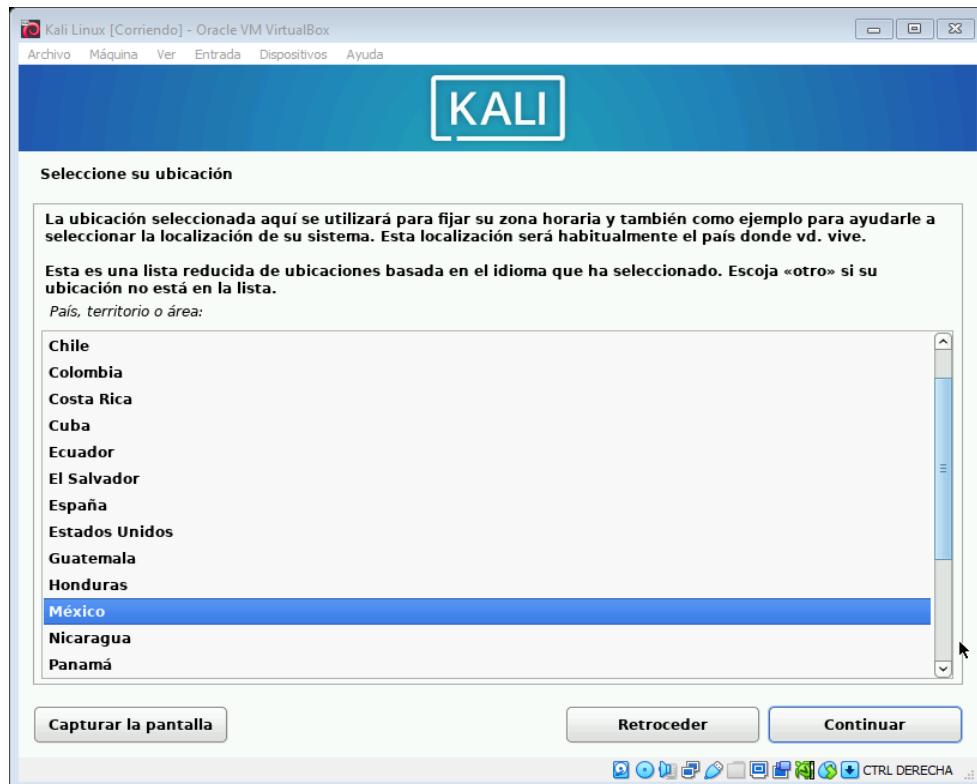
Ya que se reinició en el menú que nos aparece le damos en Start installer para instalar Kali Linux en virtual box



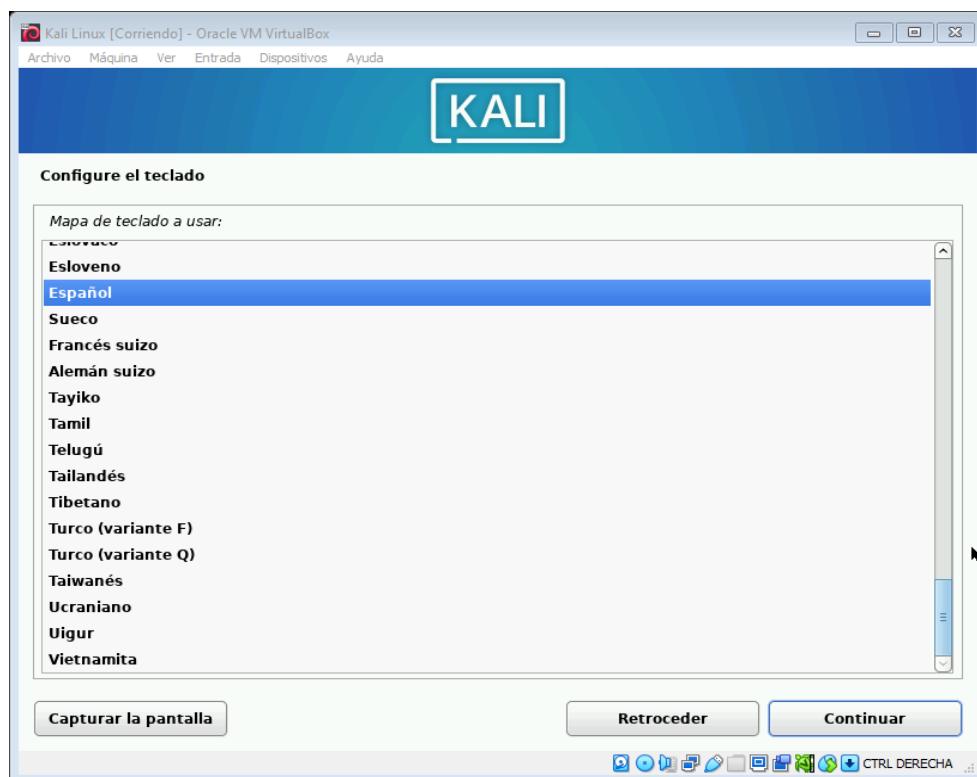
Ahora seleccionamos el idioma Spanish Español de Kali Linux y le damos en continuar.



Igual seleccionamos la ubicación y le damos en México



Y también tendríamos que seleccionamos el idioma del teclado que seria en español y le damos continuar.



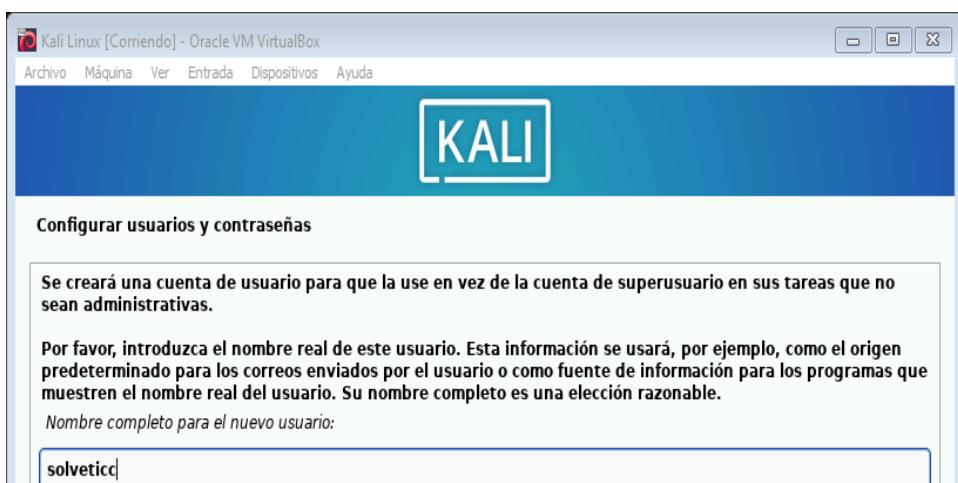
## Empieza a cargar componentes adicionales



Ya que cargo asignamos un nombre a la maquina en este caso será Kali



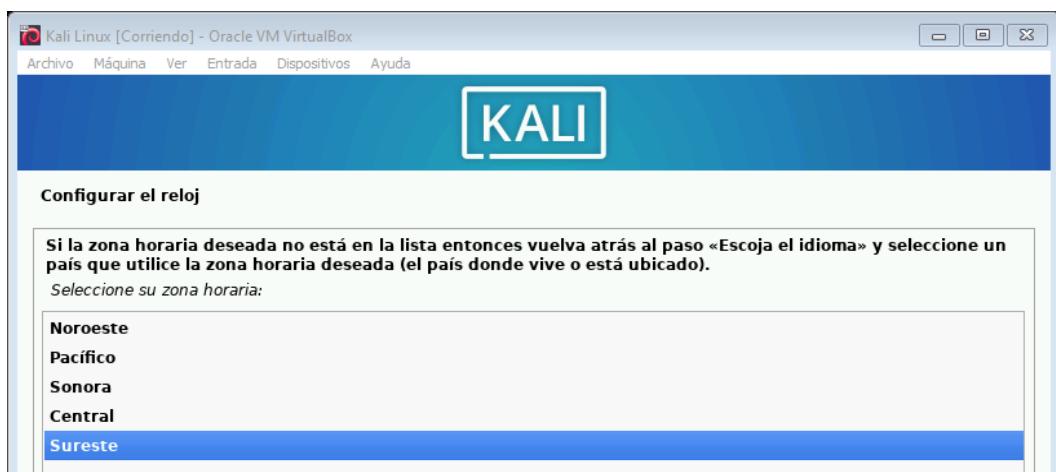
Ahora vamos asignar un usuario administrador y contraseña la cual le pondré solveticc.



Ahora vamos a poner una contraseña que no pueda ser delibrada tan fácilmente, en este caso le pondremos hQpLtYrZwMxDaU.



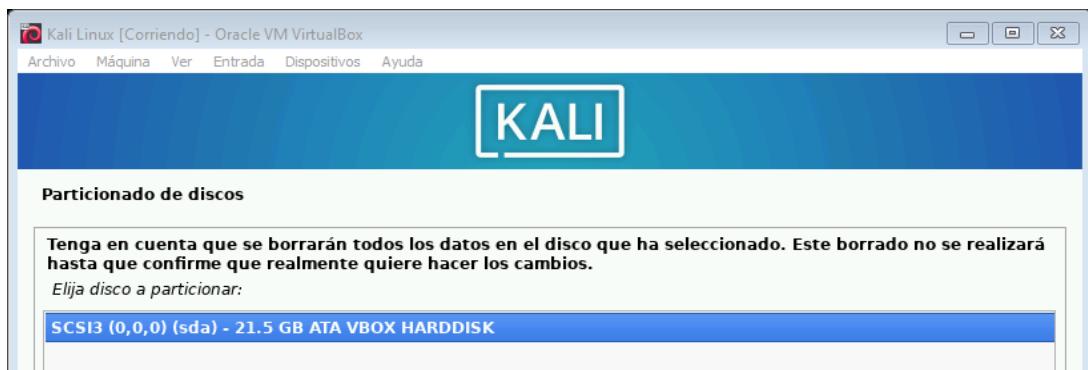
Después aparece que se esta configurando el reloj y seleccionamos la zona horaria y le damos en sureste



Ahora vamos a definir el particionado de los discos y le damos en el primero que es en el guiado



## Confirmamos el proceso



Después le damos en todos los ficheros en partición



Se le da en finalizar la partición



Después le damos en la opción donde dice si



Se reiniciará la instalación de Kali Linux en virtual box



Ya que termine debemos de confirmar si se usara una replica en red la cual le decimos que no.



Después nos aparecerá otra pestaña en donde debemos de activar la casilla si para instalar GRUB



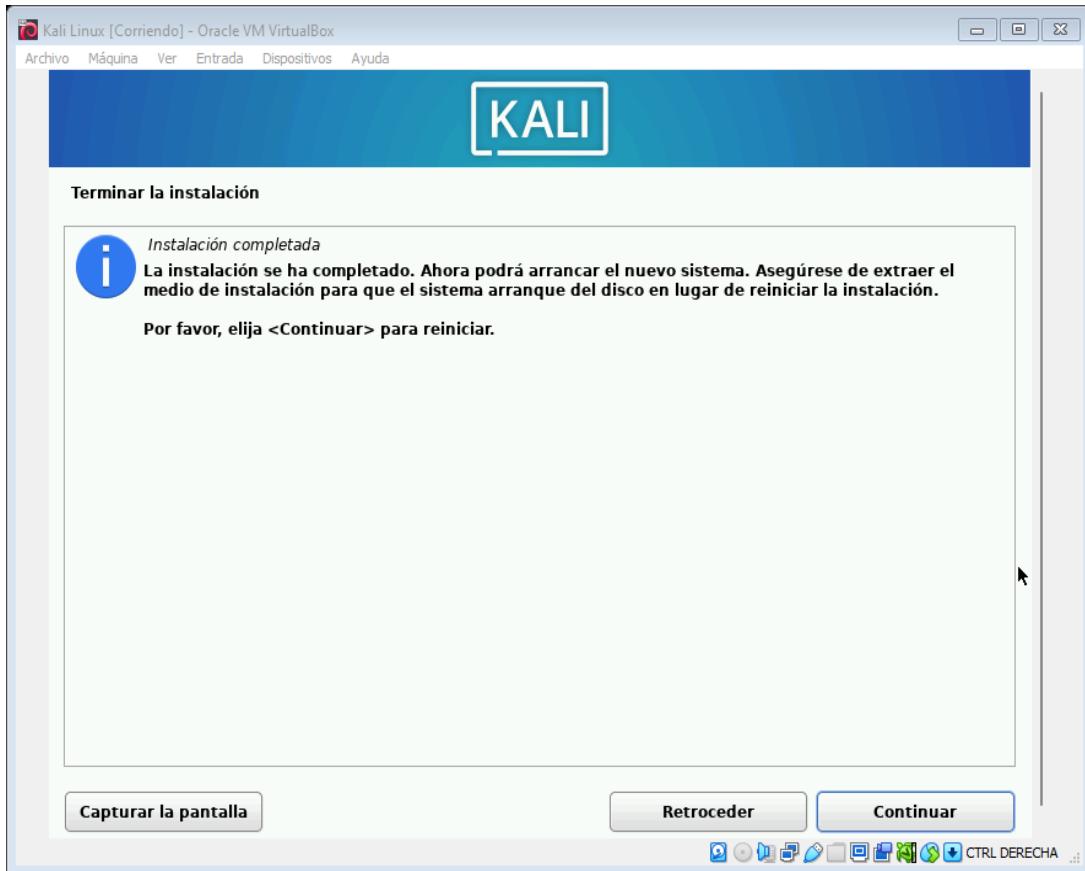
Después seleccionamos la partición en la cual se instalará.



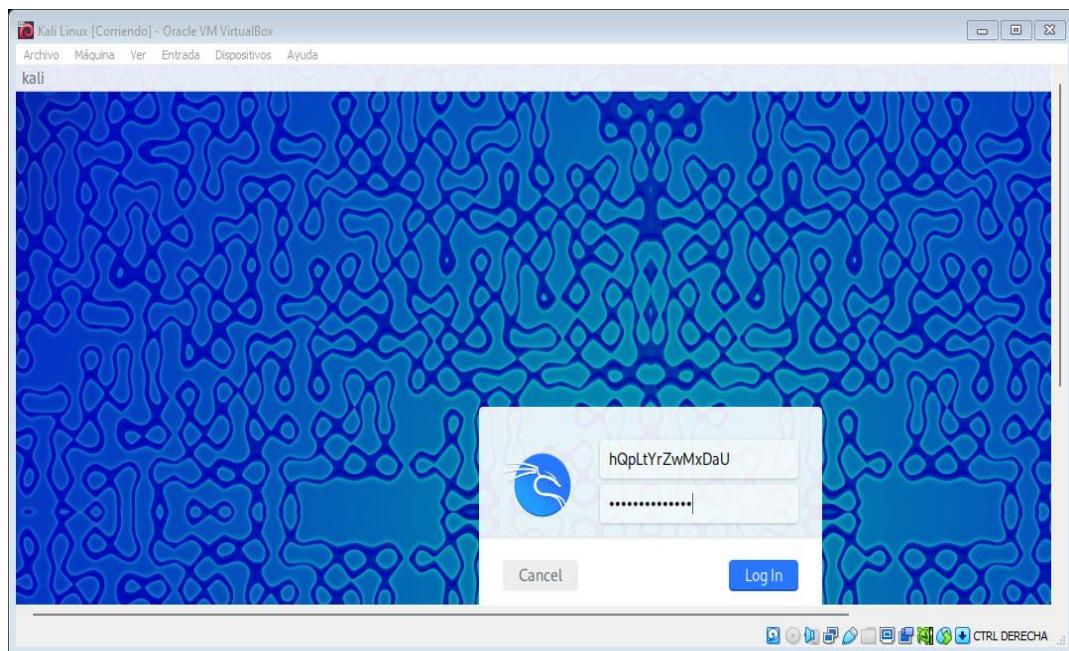
Se completa el proceso de instalación



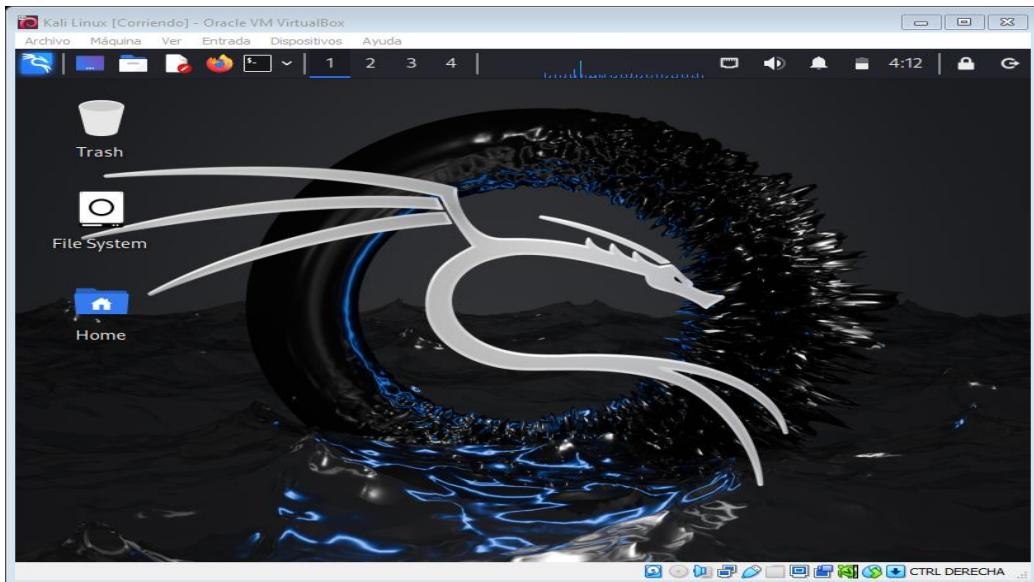
Después dirá que la instalación esta completa y le damos en continuar



La máquina será reiniciada y nos pedirá la contraseña que le pusimos y le damos en iniciar sesión



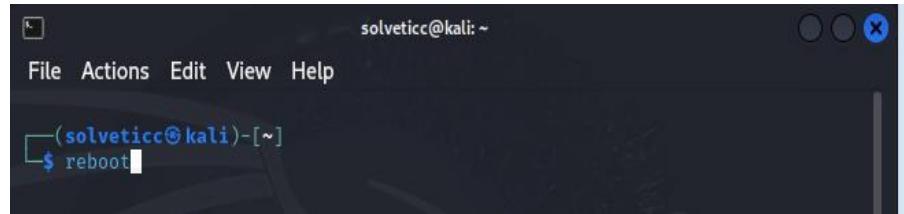
Al poner el nombre del usuario y contraseña ya podemos ingresar en Kali Linux correctamente como se ve en la captura de pantalla.



Ahora para instalar las virtualBOx Guest Additions abrimos la terminal y en actualizar el sistema ponemos sudo apt update y ingresamos la contraseña.

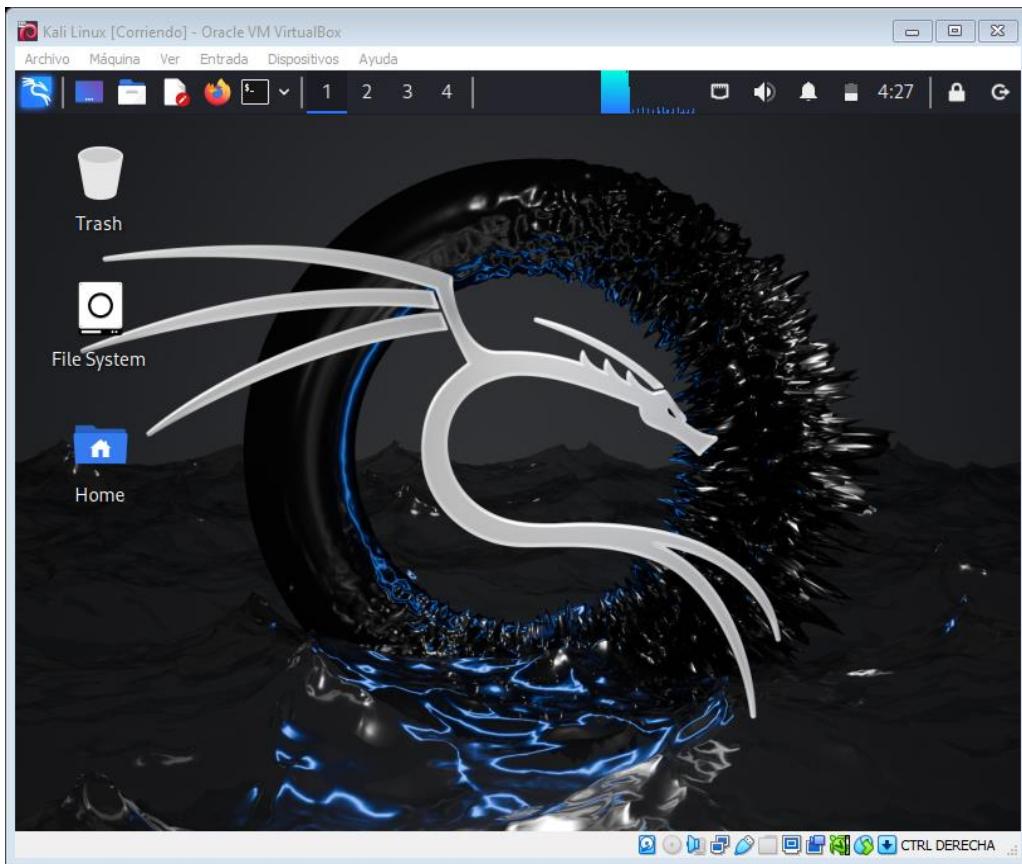
A screenshot of a terminal window titled 'solveticc@kali: ~'. The window shows the user's prompt '(solveticc@kali)-[~]\$'. The user has typed '\$ sudo apt update' and is awaiting the password. The text 'password for solveticc:' is visible, followed by 'Sorry, try again.' and '[sudo] password for solveticc:'. The progress bar at the bottom indicates '0% [Working]'.A screenshot of a terminal window titled 'solveticc@kali: ~'. The window shows the user's prompt '(solveticc@kali)-[~]\$'. The user has completed the command '\$ sudo apt update'. The output includes several error messages: 'Ign:1 http://http.kali.org/kali kali-rolling InRelease', 'Err:1 http://http.kali.org/kali kali-rolling InRelease', 'Temporary failure resolving 'http.kali.org'', 'All packages are up to date.', 'Warning: Failed to fetch http://http.kali.org/kali/dists/kali-rolling/InRelease', 'Temporary failure resolving 'http.kali.org'', 'Warning: Some index files failed to download. They have been ignored, or old ones used instead.', 'Notice: Repository 'Kali Linux' changed its 'firmware component' value from 'non-free' to 'non-free-firmware'', and 'Notice: More information about this can be found online at: https://www.kali.org/blog/non-free-firmware-transition/'. The terminal prompt '(solveticc@kali)-[~]\$' is visible at the bottom.

Ahora ejecutamos el comando sudo apt install virtualbox-x11, confirmamos la operación y reiniciamos la maquina



```
solveticc@kali: ~
File Actions Edit View Help
(solveticc@kali)-[~]
$ reboot
```

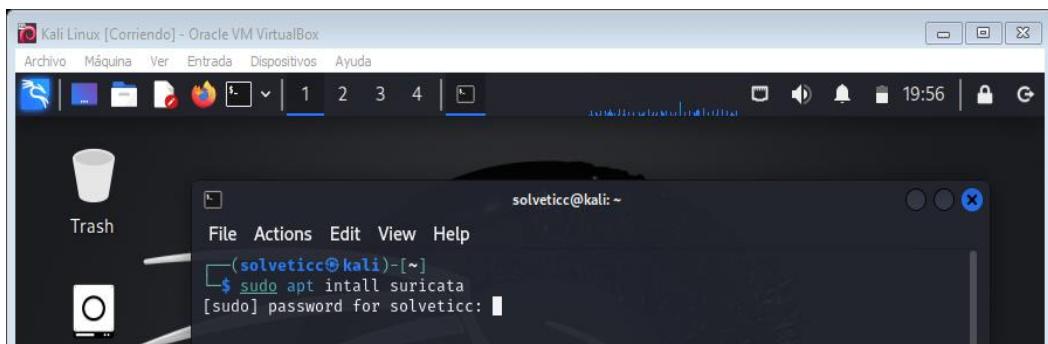
Ahora iniciamos sesión y ya podremos usar al máximo Kali Linux en virtual Box.



## **INSTALAR UN SISTEMA DE DETECCIÓN DE INTRUSOS COMO SURICATA.**

## **Instalación de del sistema de detección de intrusos suricata.**

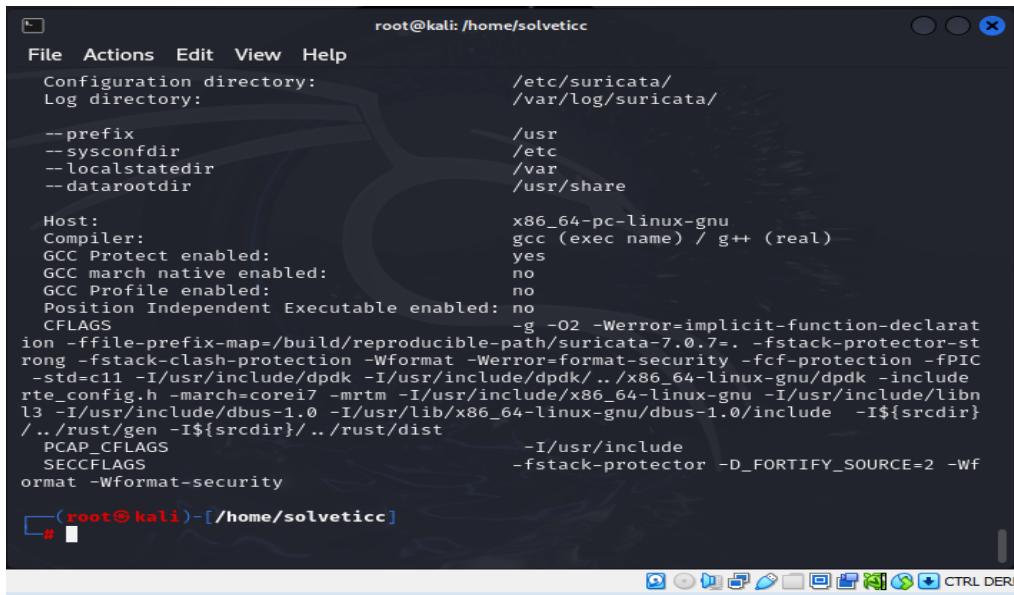
Primero nos dirigimos al Kali linus y nos vamos a la terminal para poder instalar suricata. Para poder instalar Suricata vamos a utilizar el comando sudo su y entramos al modo administrador para poder instalar suricata.



Nos pide la contraseña, se la ponemos y empieza el proceso. Y bueno al terminar ya tenemos instalado suricata.

```
[root@kali: /home/solveticcc] File Actions Edit View Help  
Reading package lists ... Done  
Building dependency tree ... Done  
Reading state information... Done  
The following additional packages will be installed:  
  isa-support libc-bin libc-dev-bin libc-l10n libc6 libc6-dev libc6-i386  
  libfdt1 libhypert2 libhyperscan5 libnetfilter-log1 librte-bus-pci24  
  librte-bus-vdev24 librte-eal24 librte-ethdev24 librte-hash24 librte-ip-frag24  
  librte-kvargs24 librte-log24 librte-mbuf24 librte-mempool24 librte-meter24  
  librte-net-bond24 librte-net24 librte-pci24 librte-rcu24 librte-ring24  
  librte-sched24 librte-telemetry24 libxdp1 locales locales-all sse3-support  
  sse4.2-support suricata-update  
Suggested packages:  
  libtcmalloc-minimal4  
The following NEW packages will be installed:  
  isa-support libfdt1 libhypert2 libhyperscan5 libnetfilter-log1 librte-bus-pci24  
  librte-bus-vdev24 librte-eal24 librte-ethdev24 librte-hash24 librte-ip-frag24  
  librte-kvargs24 librte-log24 librte-mbuf24 librte-mempool24 librte-meter24  
  librte-net-bond24 librte-net24 librte-pci24 librte-rcu24 librte-ring24  
  librte-sched24 librte-telemetry24 libxdp1 sse3-support sse4.2-support  
  suricata suricata-update  
The following packages will be upgraded:  
  libc-bin libc-dev-bin libc-l10n libc6 libc6-dev libc6-i386 locales  
  locales-all  
8 upgraded, 28 newly installed, 0 to remove and 1805 not upgraded.  
Need to get 30.2 MB of archives.  
After this operation, 38.5 MB of additional disk space will be used.  
Do you want to continue? [Y/n] █
```

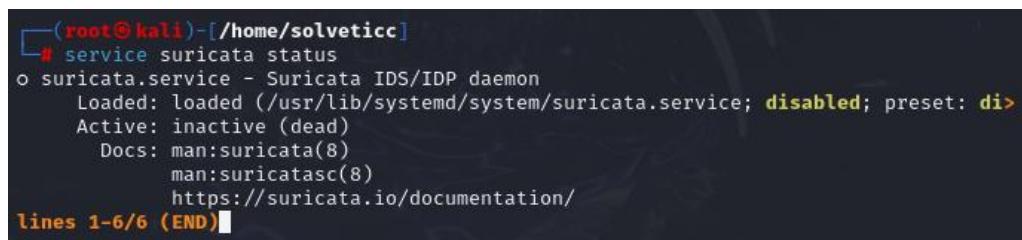
Ahora checamos que si se instaló correctamente y efectivamente ya está , ahí podemos ver los datos de la instalación como la configuración del directorio suricata



```
root@kali: /home/solveticc
File Actions Edit View Help
Configuration directory: /etc/suricata/
Log directory: /var/log/suricata/
--prefix /usr
--sysconfdir /etc
--localstatedir /var
--datarootdir /usr/share
Host: x86_64-pc-linux-gnu
Compiler: gcc (exec name) / g++ (real)
GCC Protect enabled: yes
GCC march native enabled: no
GCC Profile enabled: no
Position Independent Executable enabled: no
CFLAGS -g -O2 -Werror=implicit-function-declaration -fstack-protector-strong -fstack-clash-protection -Wformat -Werror=format-security -fcf-protection -fPIC -std=c11 -I/usr/include/dpdk -I/usr/include/dpdk/..x86_64-linux-gnu/dpdk -include rte_config.h -march=corei7 -mrtm -I/usr/include/x86_64-linux-gnu -I/usr/include/libnl3 -I/usr/include/dbus-1.0 -I/usr/lib/x86_64-linux-gnu/dbus-1.0/include -I${srcdir} /..rust/gen -I${srcdir}/..rust/dist
PCAP_CFLAGS -I/usr/include
SECCFLAGS -fstack-protector -D_FORTIFY_SOURCE=2 -Wformat -Wformat-security
format -Wformat-security

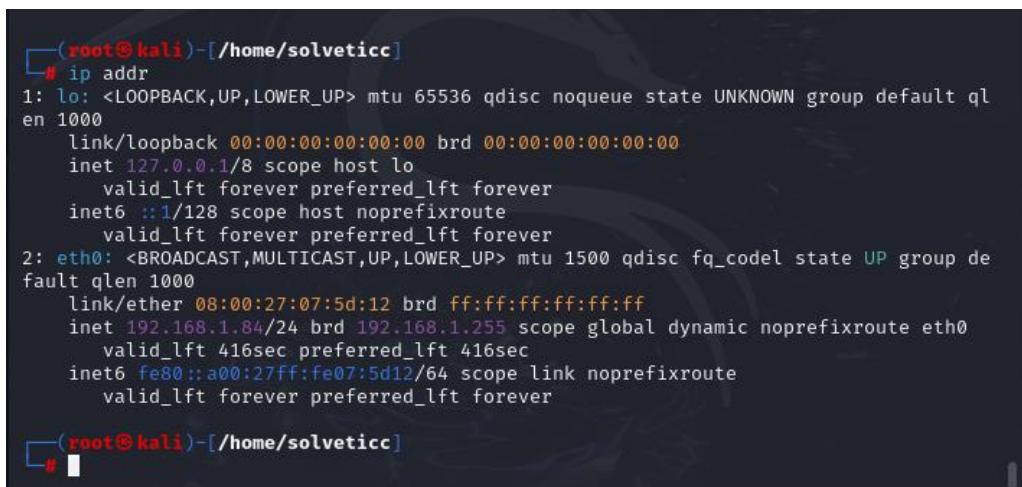
[root@kali]#
```

Ahora vamos a validar el estatus de suricata con el siguiente comando service suricata status



```
(root@kali)-[~/home/solveticc]
# service suricata status
● suricata.service - Suricata IDS/IDP daemon
  Loaded: loaded (/usr/lib/systemd/system/suricata.service; disabled; preset: disabled)
  Active: inactive (dead)
    Docs: man:suricata(8)
          man:suricatasc(8)
          https://suricata.io/documentation/
lines 1-6/6 (END)
```

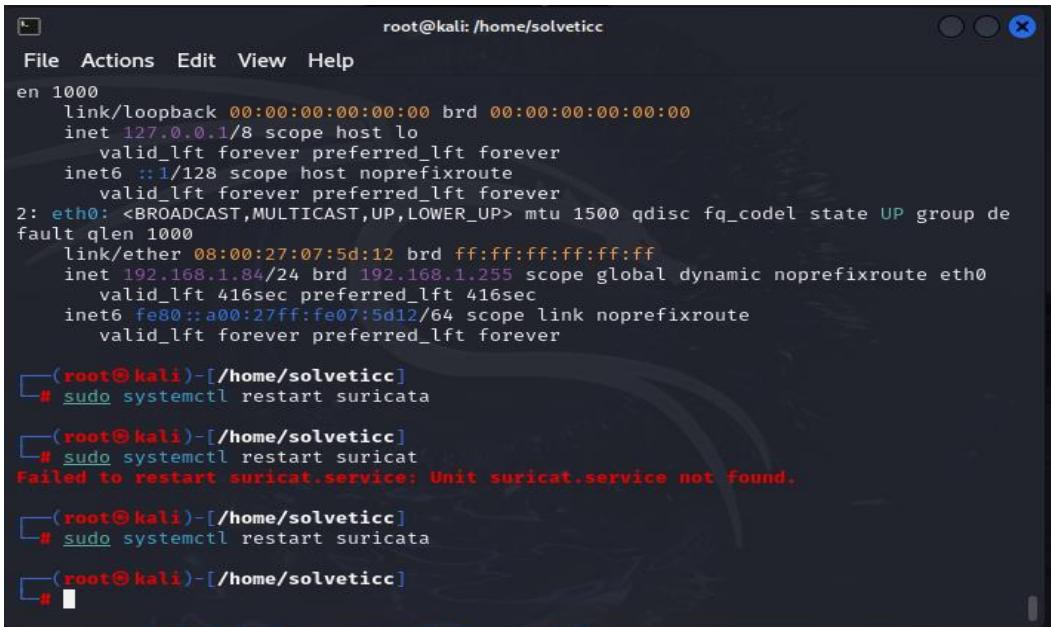
Lo siguiente lo que vamos hacer es validar nuestra ip addr, aquí lo importante es verificar la interfaz de red



```
(root@kali)-[~/home/solveticc]
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:07:5d:12 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.84/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
        valid_lft 416sec preferred_lft 416sec
        inet6 fe80::a00:27ff:fe07:5d12/64 scope link noprefixroute
            valid_lft forever preferred_lft forever

[root@kali]-[~/home/solveticc]
#
```

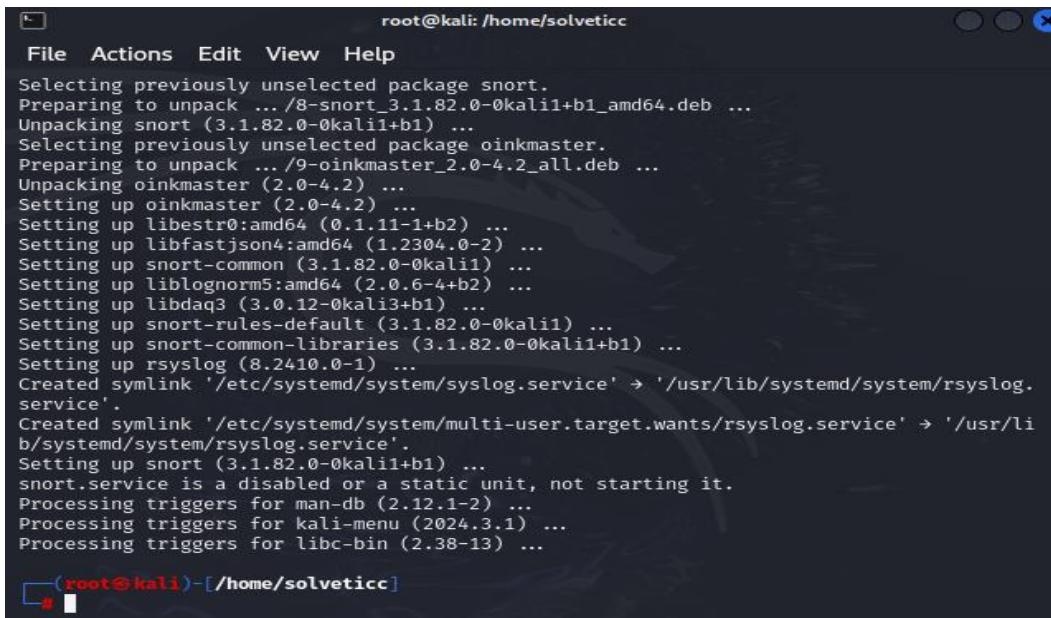
Luego con el comando sudo systemctl restart suricata se va a reiniciar el suricata.



```
root@kali: /home/solveticc
File Actions Edit View Help
en 1000
    Link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:07:5d:12 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.84/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
        valid_lft 416sec preferred_lft 416sec
    inet6 fe80::a00:27ff:fe07:5d12/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

[root@kali ~]# sudo systemctl restart suricata
[root@kali ~]# sudo systemctl restart suricata
Failed to restart suricat.service: Unit suricat.service not found.
[root@kali ~]# sudo systemctl restart suricata
[root@kali ~]#
```

Aquí ya se reinicio correctamente

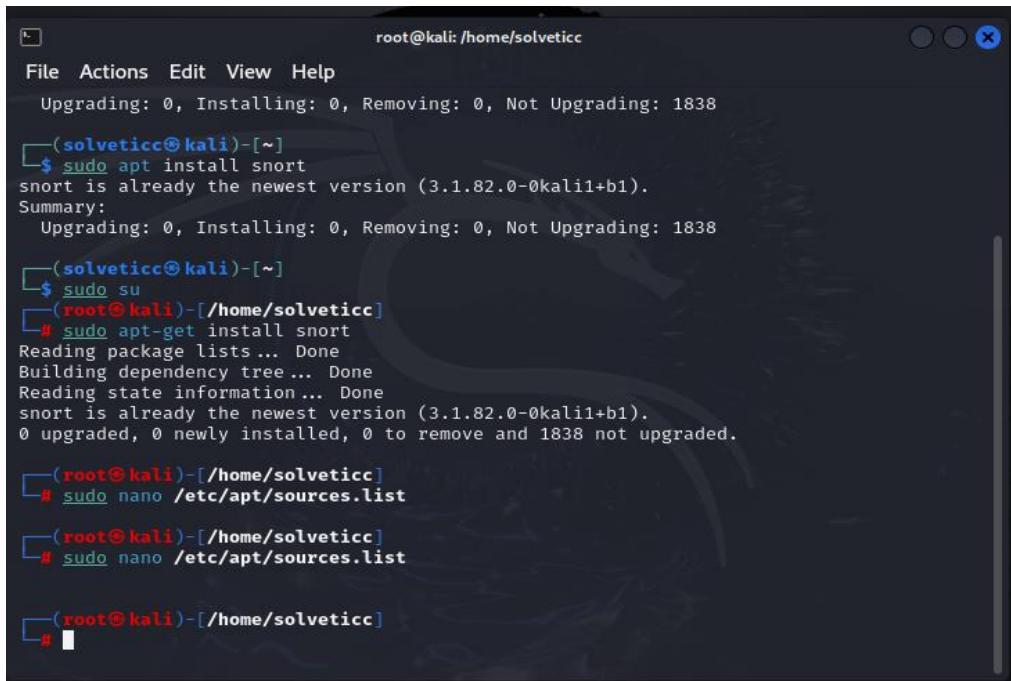


```
root@kali: /home/solveticc
File Actions Edit View Help
Selecting previously unselected package snort.
Preparing to unpack .../8-snort_3.1.82.0-0kali1+b1_amd64.deb ...
Unpacking snort (3.1.82.0-0kali1+b1) ...
Selecting previously unselected package oinkmaster.
Preparing to unpack .../9-oinkmaster_2.0-4.2_all.deb ...
Unpacking oinkmaster (2.0-4.2) ...
Setting up oinkmaster (2.0-4.2) ...
Setting up libestr0:amd64 (0.1.11-1+b2) ...
Setting up libfastjson4:amd64 (1.2304.0-2) ...
Setting up snort-common (3.1.82.0-0kali1) ...
Setting up liblognorm5:amd64 (2.0.6-4+b2) ...
Setting up libdaq3 (3.0.12-0kali3+b1) ...
Setting up snort-rules-default (3.1.82.0-0kali1) ...
Setting up snort-common-libraries (3.1.82.0-0kali1+b1) ...
Setting up rsyslog (8.2410.0-1) ...
Created symlink '/etc/systemd/system/syslog.service' → '/usr/lib/systemd/system/rsyslog.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/rsyslog.service' → '/usr/lib/systemd/system/rsyslog.service'.
Setting up snort (3.1.82.0-0kali1+b1) ...
snort.service is a disabled or a static unit, not starting it.
Processing triggers for man-db (2.12.1-2) ...
Processing triggers for kali-menu (2024.3.1) ...
Processing triggers for libc-bin (2.38-13) ...

[root@kali ~]#
```

## Configuración de reglas de detección de intrusos

Ahora vamos a crear nuestro archivo de regla con el comando touch /etc/snort/rules/custom.rules



```
root@kali: /home/solveticc
File Actions Edit View Help
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1838

[(solveticc㉿kali)-[~]
$ sudo apt install snort
snort is already the newest version (3.1.82.0-0kali1+b1).
Summary:
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1838

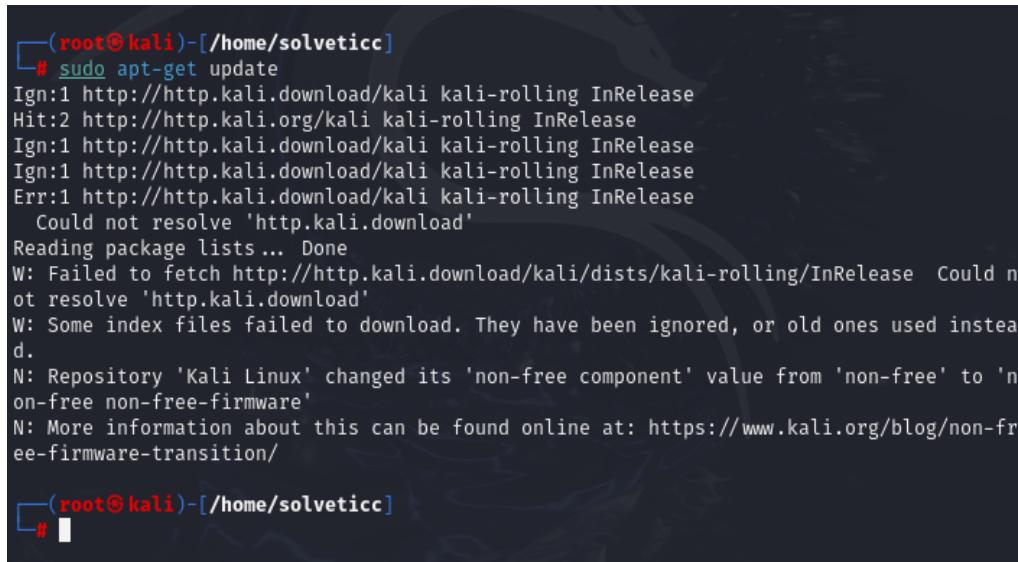
[(solveticc㉿kali)-[~]
$ sudo su
[(root㉿kali)-[/home/solveticc]
# sudo apt-get install snort
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
snort is already the newest version (3.1.82.0-0kali1+b1).
0 upgraded, 0 newly installed, 0 to remove and 1838 not upgraded.

[(root㉿kali)-[/home/solveticc]
# sudo nano /etc/apt/sources.list

[(root㉿kali)-[/home/solveticc]
# sudo nano /etc/apt/sources.list

[(root㉿kali)-[/home/solveticc]
# ]
```

Ahora vamos hacer sudo apt- get update para realizar las get



```
(root㉿kali)-[/home/solveticc]
# sudo apt-get update
Ign:1 http://http.kali.download/kali kali-rolling InRelease
Hit:2 http://http.kali.org/kali kali-rolling InRelease
Ign:1 http://http.kali.download/kali kali-rolling InRelease
Ign:1 http://http.kali.download/kali kali-rolling InRelease
Err:1 http://http.kali.download/kali kali-rolling InRelease
  Could not resolve 'http.kali.download'
Reading package lists... Done
W: Failed to fetch http://http.kali.download/kali/dists/kali-rolling/InRelease  Could n
ot resolve 'http.kali.download'
W: Some index files failed to download. They have been ignored, or old ones used instea
d.
N: Repository 'Kali Linux' changed its 'non-free component' value from 'non-free' to 'n
on-free non-free-firmware'
N: More information about this can be found online at: https://www.kali.org/blog/non-fr
ee-firmware-transition/

[(root㉿kali)-[/home/solveticc]
# ]
```

Una vez que ya este reiniciado con el comando sudo systemctl status suricata vamos a ver el estado y aquí nos dice que ya está activo y corriendo.

```
[root@kali]~[/home/solveticc]
# sudo systemctl status suricata
● suricata.service - Suricata IDS/IDP daemon
  Loaded: loaded (/usr/lib/systemd/system/suricata.service; disabled; preset: di>
    Active: active (running) since Mon 2024-11-18 20:56:31 UTC; 1min 7s ago
  Invocation: 6dc03c3593e745dea8c1f8e0b41eafe4
    Docs: man:suricata(8)
          man:suricatasc(8)
          https://suricata.io/documentation/
  Process: 17536 ExecStart=/usr/bin/suricata -D --af-packet -c /etc/suricata/suri>
 Main PID: 17546 (Suricata-Main)
   Tasks: 7 (limit: 2260)
  Memory: 40.1M (peak: 40.3M)
     CPU: 1.402s
    CGroup: /system.slice/suricata.service
              └─17546 /usr/bin/suricata -D --af-packet -c /etc/suricata/suricata.yam>

Nov 18 20:56:31 kali systemd[1]: Starting suricata.service - Suricata IDS/IDP daemo>
Nov 18 20:56:31 kali suricata[17536]: i: suricata: This is Suricata version 7.0.7 R>
Nov 18 20:56:31 kali systemd[1]: Started suricata.service - Suricata IDS/IDP daemon.
lines 1-18/18 (END)
```

Ahora vamos a utilizar el comando sudo suricata-update update—sources

```
[root@kali]~[/home/solveticc]
# sudo suricata-update update-sources
18/11/2024 -- 21:03:30 - <Info> -- Using data-directory /var/lib/suricata.
18/11/2024 -- 21:03:30 - <Info> -- Using Suricata configuration /etc/suricata/surica>
ta.yaml
18/11/2024 -- 21:03:30 - <Info> -- Using /etc/suricata/rules for Suricata provided r
ules.
18/11/2024 -- 21:03:30 - <Info> -- Found Suricata version 7.0.7 at /usr/bin/suricata
.
18/11/2024 -- 21:03:30 - <Info> -- Downloading https://www.openinfosecfoundation.org
/rules/index.yaml
18/11/2024 -- 21:04:00 - <Error> -- Failed to download index: https://www.openinfose
cfoundation.org/rules/index.yaml: curlopen error _ssl.c:989: The handshake operation
timed out

[root@kali]~[/home/solveticc]
#
```

Y Por último vamos a utilizar el comando sudo suricata-update list-sources para validar que está en la lista sours.

```
File Actions Edit View Help
root@kali: /home/solveticc
License: Commercial
Parameters: secret-code
Subscription: https://www.stamus-networks.com/stamus-labs/subscribe-to-threat-inte
l-feed
Name: stamus/nrd-entropy-30-open
Vendor: Stamus Networks
Summary: Newly Registered Domains Open only - 30 day list, high entropy
License: Commercial
Parameters: secret-code
Subscription: https://www.stamus-networks.com/stamus-labs/subscribe-to-threat-inte
l-feed
Name: stamus/nrd-phishing-14-open
Vendor: Stamus Networks
Summary: Newly Registered Domains Open only - 14 day list, phishing
License: Commercial
Parameters: secret-code
Subscription: https://www.stamus-networks.com/stamus-labs/subscribe-to-threat-inte
l-feed
Name: stamus/nrd-phishing-30-open
Vendor: Stamus Networks
Summary: Newly Registered Domains Open only - 30 day list, phishing
License: Commercial
Parameters: secret-code
Subscription: https://www.stamus-networks.com/stamus-labs/subscribe-to-threat-inte
l-feed
Name: tgreen/hunting
```

## Configuración de reglas y alertas de detección de intrusos.

Ya con suricata instalado ahora vamos a descargar las reglas de la comunidad suricata

```
[root@kali)-[/home/solveticc]
# wget http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz
--2024-11-18 21:34:07--  http://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz
Resolving rules.emergingthreats.net (rules.emergingthreats.net) ... 54.173.5.230, 34.193.218.6, 3.226.221.201, ...
Connecting to rules.emergingthreats.net (rules.emergingthreats.net)|54.173.5.230|:80...
... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 4599733 (4.4M) [application/octet-stream]
Saving to: 'emerging.rules.tar.gz'

emerging.rules.tar.g 100%[=====] 4.39M 555KB/s in 10s

2024-11-18 21:34:19 (437 KB/s) - 'emerging.rules.tar.gz' saved [4599733/4599733]

[root@kali)-[/home/solveticc]
#
```

Ahora descomprimimos el archivo de las reglas que descargamos

```
[root@kali)-[/home/solveticc]
# tar zxvf emerging.rules.tar.gz
rules/
rules/3coresec.rules
rules/BSD-License.txt
rules/LICENSE
rules/botcc.portgrouped.rules
rules/botcc.rules
rules/ciarmy.rules
rules/classification.config
rules/compromised-ips.txt
rules/compromised.rules
rules/drop.rules
rules/dshield.rules
rules/emerging-activex.rules
rules/emerging-adware_pup.rules
rules/emerging-attack_response.rules
rules/emerging-chat.rules
rules/emerging-coinminer.rules
rules/emerging-current_events.rules
rules/emerging-deleted.rules
rules/emerging-dns.rules
rules/emerging-dos.rules
rules/emerging-exploit.rules
rules/emerging-exploit_kit.rules
rules/emerging-ftp.rules
```

Ahora vamos a mover las reglas al archivo que ya está creado ósea el directorio, ya abrimos la reglas y abrimos el directorio y vamos a abrir el editor de reglas.

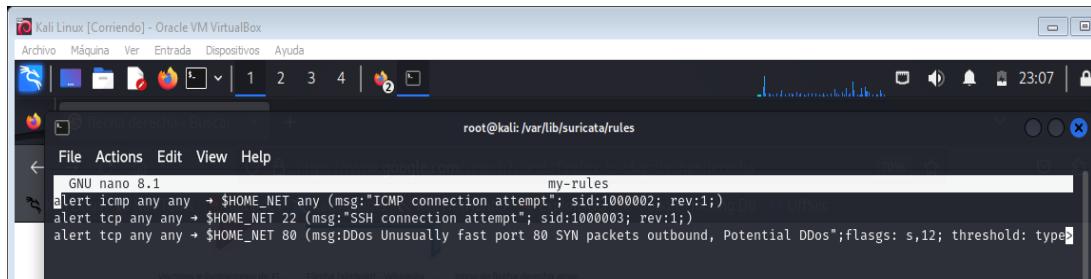
```
(root@kali)-[~/home/solveticc]
# sudo mkdir /var/lib/suricata
mkdir: cannot create directory '/var/lib/suricata': File exists

(root@kali)-[~/home/solveticc]
# sudo mv rules /var/lib/suricata/

(root@kali)-[~/home/solveticc]
# cd /var/lib/suricata/rules

(root@kali)-[/var/lib/suricata/rules]
# !
```

Ahora vamos a crear tres reglas, la primera regla es de penetrante, la segunda regla es para las conexiones que irán al puerto 22 Y la tercera regla va hacer para ataques del puerto 80



Ahora con el comando sudo nano /etc/suricata/suricata.yaml vamos a hacer la definición del par de reglas.

```
root@kali:/var/lib/suricata/rules
File Actions Edit View Help
GNU nano 8.1          /etc/suricata/suricata.yaml
YAML 1.1  Kali Tutorials  Kali Docs  Kali Forums  Kali NetHunter  Exploit-DB  Google Hacking Database  OffSec

# Suricata configuration file. In addition to the comments describing all
# options in this file, full documentation can be found at:
# https://docs.suricata.io/en/latest/configuration/suricata-yaml.html

# This configuration file generated by Suricata 7.0.7.
suricata-version: "7.0"

##
## Step 1: Inform Suricata about your network
##
# Flechas, Símbolos de flechas y direcciones Unicode → ...
vars:
    # more specific is better for alert accuracy and performance
address-groups:
    HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]"
    #HOME_NET: "[192.168.0.0/16]"
    #HOME_NET: "[10.0.0.0/8]"  BillApp
    #HOME_NET: "[172.16.0.0/12]"  BillApp
    #HOME_NET: "any"  BillApp

EXTERNAL_NET: "!$HOME_NET"
[ Read 2188 lines ]
```

Realizamos la búsqueda y remplazamos la ruta de las librerías y la ruta de rules.

```
# hashmode: hash5tuplesorted; Flecha (símbolo) - Wikipedia, ... icono de flecha
## Configure Suricata to load Suricata-Update managed rules.
## default-rule-path: /var/lib/suricata/rules
rule-files: Flechas. Símbolos de flechas y direcciones Unicode → ► ...
- emerging-exploit.rules
- my-rules [ nombre de usuario o estado en (@_@)SYMBL
## PiliApp
## Auxiliary configuration files.
## Símbolos de flecha - - PiliApp
classification-file: /etc/suricata/classification.config
reference-config-file: /etc/suricata/reference.config
# threshold-file: /etc/suricata/threshold.config
```

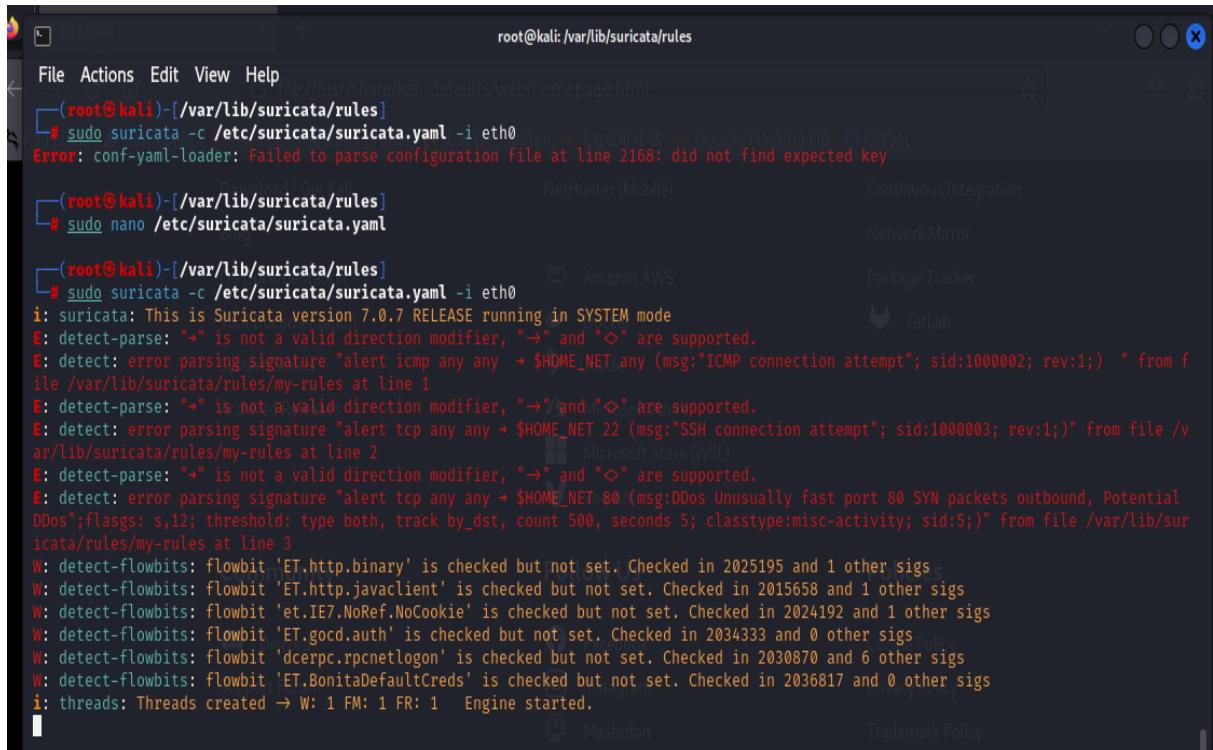
Ahora realizamos la ejecución de suricata con el comando sudo suricata -c /etc/suricata/suricata.yaml -1 eth0

```
(root㉿kali)-[~/var/lib/suricata/rules]
# sudo suricata -c /etc/suricata/suricata.yaml -1 eth0
```

El ethernet cero sale de nuestra red lo que debemos de hacer es buscar el ip confid es donde vamos apuntar las alertas.

```
[sudo] password for solveticcc:
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.1.84 netmask 255.255.255.0 broadcast 192.168.1.255
      inet6 fe80::a00:27ff:fe07:5d12 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:07:5d:12 txqueuelen 1000 (Ethernet)
          RX packets 22527 bytes 15575655 (14.8 MiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 8220 bytes 1104851 (1.0 MiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
          RX packets 25 bytes 2852 (2.7 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 25 bytes 2852 (2.7 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Y ahí queda la ejecución de suricata



```
root@kali: /var/lib/suricata/rules
File Actions Edit View Help
[root@kali ~]# sudo suricata -c /etc/suricata/suricata.yaml -i eth0
Error: conf-yaml-loader: Failed to parse configuration file at line 2168: did not find expected key

[root@kali ~]# sudo nano /etc/suricata/suricata.yaml

[root@kali ~]# sudo suricata -c /etc/suricata/suricata.yaml -i eth0
i: suricata: This is Suricata version 7.0.7 RELEASE running in SYSTEM mode
E: detect-parse: "→" is not a valid direction modifier, "→" and "▷" are supported.
E: detect: error parsing signature "alert icmp any any → $HOME_NET.any (msg:"ICMP connection attempt"; sid:1000002; rev:1;)" from file /var/lib/suricata/rules/my-rules at line 1
E: detect-parse: "→" is not a valid direction modifier, "→" and "▷" are supported.
E: detect: error parsing signature "alert tcp any any + $HOME_NET 22 (msg:"SSH connection attempt"; sid:1000003; rev:1;)" from file /var/lib/suricata/rules/my-rules at line 2
E: detect-parse: "→" is not a valid direction modifier, "→" and "▷" are supported.
E: detect: error parsing signature "alert tcp any any + $HOME_NET 80 (msg:DDos Unusually fast port 80 SYN packets outbound, Potential DDos';flags: s,l2; threshold: type both, track by_dst, count 500, seconds 5; classtype:misc-activity; sid:5;)" from file /var/lib/suricata/rules/my-rules at line 3
W: detect-flowbits: flowbit 'ET.http.binary' is checked but not set. Checked in 2025195 and 1 other sigs
W: detect-flowbits: flowbit 'ET.http.javaclient' is checked but not set. Checked in 2015658 and 1 other sigs
W: detect-flowbits: flowbit 'et.IE7.NoRef.NoCookie' is checked but not set. Checked in 2024192 and 1 other sigs
W: detect-flowbits: flowbit 'ET.gocd.auth' is checked but not set. Checked in 2034333 and 0 other sigs
W: detect-flowbits: flowbit 'dcerpc.rpcnetlogon' is checked but not set. Checked in 2030870 and 6 other sigs
W: detect-flowbits: flowbit 'ET.BonitaDefaultCreds' is checked but not set. Checked in 2036817 and 0 other sigs
i: threads: Threads created → W: 1 FM: 1 FR: 1 Engine started.
```

Como saber si esta funcionando podemos hacer ping la cual la voy hacer desde mi maquina normal.

```
C:\Users\reyes>ping 192.168.1.84

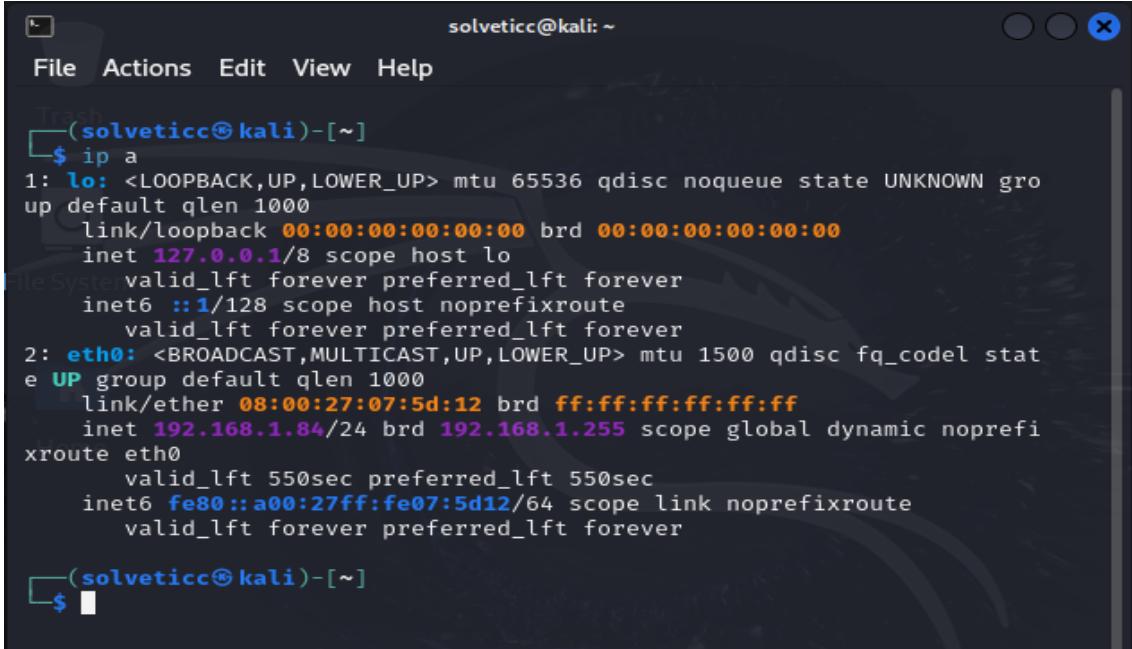
Haciendo ping a 192.168.1.84 con 32 bytes de datos:
Respuesta desde 192.168.1.84: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.84: bytes=32 tiempo=1ms TTL=64
Respuesta desde 192.168.1.84: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.84: bytes=32 tiempo=1ms TTL=64
```

```
Estadísticas de ping para 192.168.1.84:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
                (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 3ms, Media = 1ms
```

```
C:\Users\reyes>
```

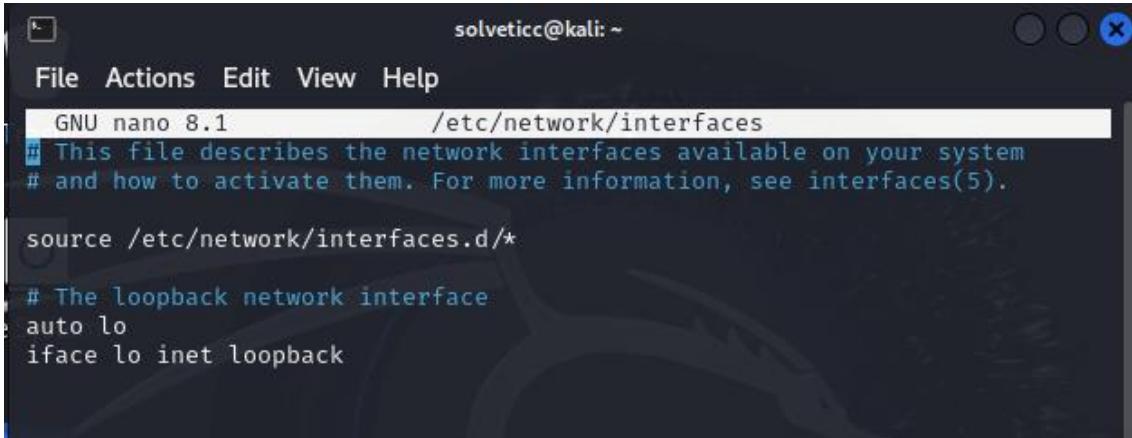
## Asignar una dirección IP estática a Kali Linux.

Primero abrimos la terminal de Kali Linux y ponemos el comando ip a y ejecutamos el comando para listar las interfaces de red disponibles.



```
solveticc@kali: ~
File Actions Edit View Help
Trash
(solveticc@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:07:5d:12 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.84/24 brd 192.168.1.255 scope global dynamic noprefixroute
        valid_lft 550sec preferred_lft 550sec
        inet6 fe80::a00:27ff:fe07:5d12/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
(solveticc@kali)-[~]
$
```

Ahora vamos a editar el archivo de configuración de la red para ello vamos abrir el archivo de configuración de red con un editor de texto como nano sudo nano /etc/network/interfaces



```
solveticc@kali: ~
File Actions Edit View Help
GNU nano 8.1          /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback
```

Buscaremos y agregaremos la configuración de la interfaz correspondiente. Si deseas asignar una IP estática a eth0, por ejemplo, nosotros le pondremos.

```
auto eth0
```

```
iface eth0 inet static
```

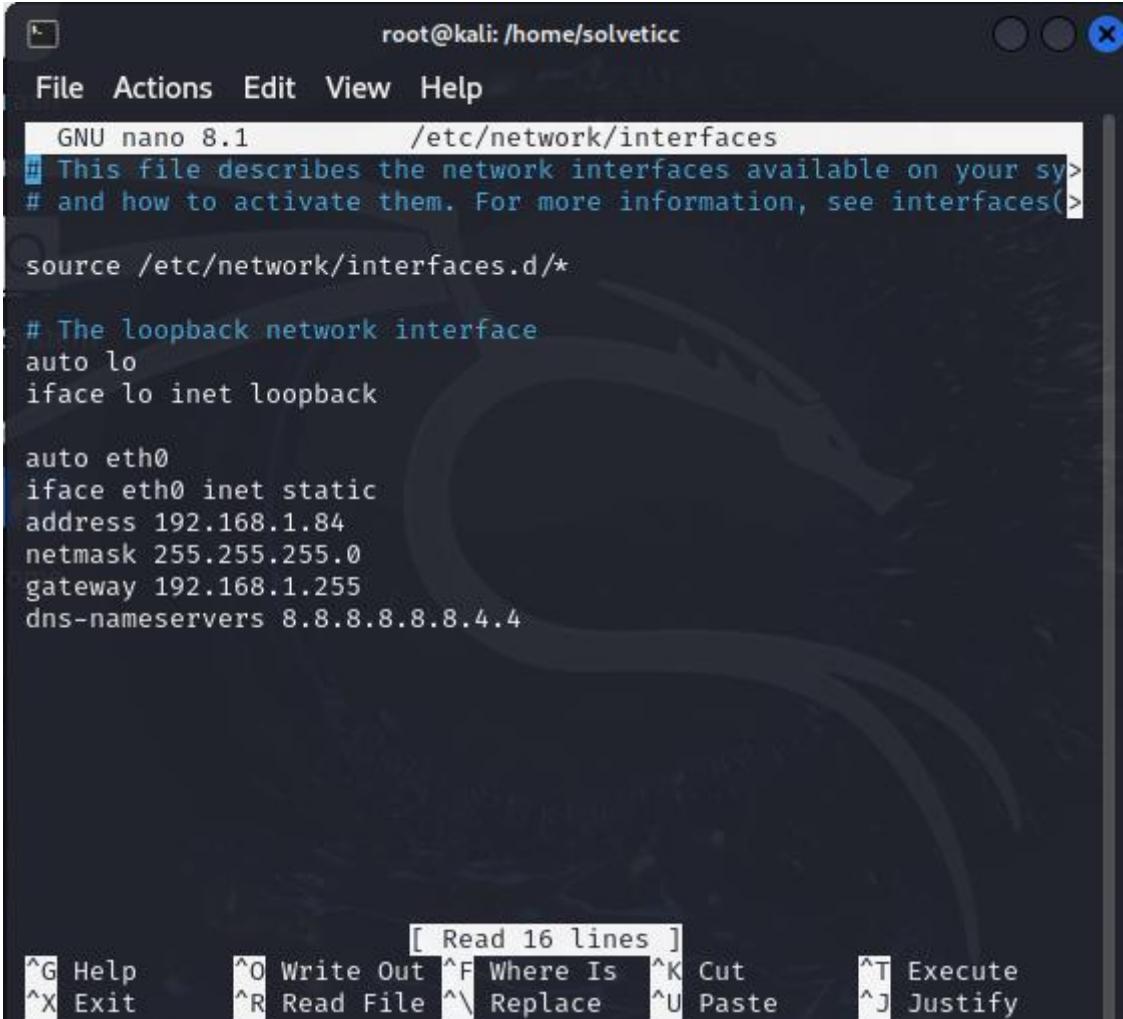
```
address 192.168.1.84
```

```
netmask 255.255.255.0
```

```
gateway 192.168.1.1
```

```
dns-nameservers
```

Entonces vamos a editar las líneas y en la primera le ponemos allow-hotplug eth0, en la segunda le ponemos iface eth0 int static y en la tercera línea se le pone la instrucción de address con la dirección 192.168.1.84 y también modificamos el Gateway con 192.168.1.255.

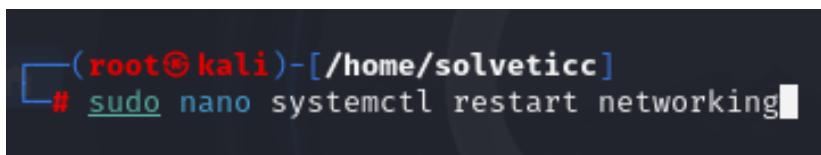


```
root@kali: /home/solveticc
File Actions Edit View Help
GNU nano 8.1      /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
source /etc/network/interfaces.d/*
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.84
netmask 255.255.255.0
gateway 192.168.1.255
dns-nameservers 8.8.8.8.8.4.4

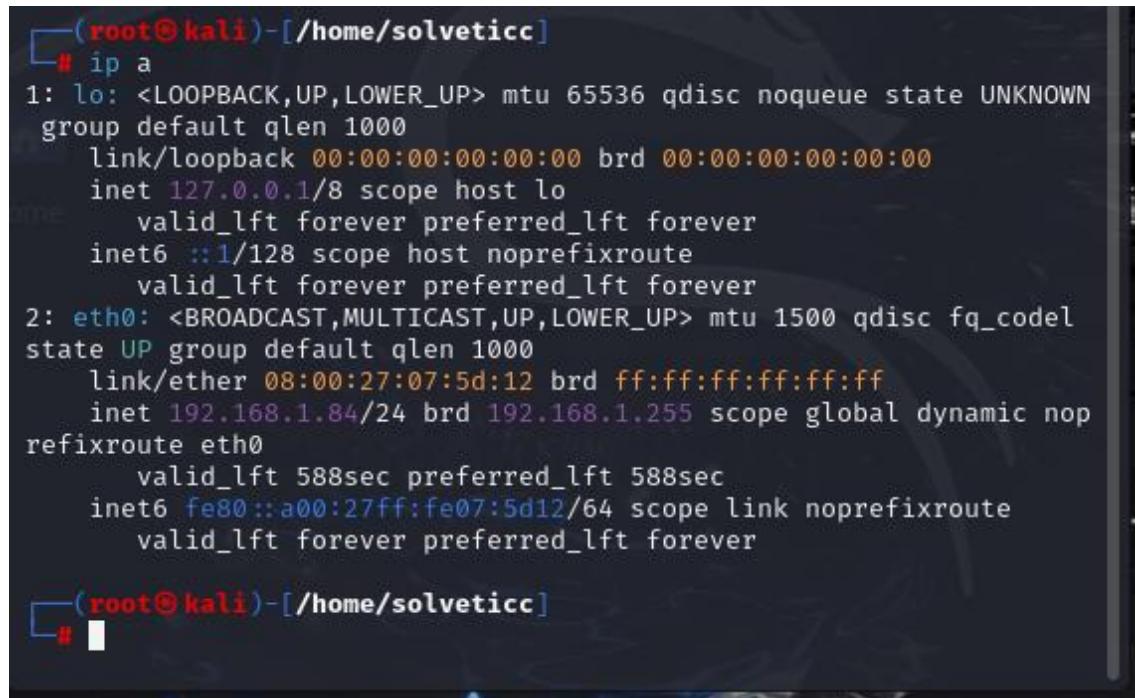
[ Read 16 lines ]
^G Help      ^O Write Out  ^F Where Is  ^K Cut      ^T Execute
^X Exit      ^R Read File  ^\ Replace   ^U Paste    ^J Justify
```

Ahora iniciamos el adaptador y con eso ya tendría nuevas direcciones



```
(root@kali)-[/home/solveticc]
# sudo nano systemctl restart networking
```

Hora checamos que la ip estática este asignada con el comando ip a y efectivamente ya esta

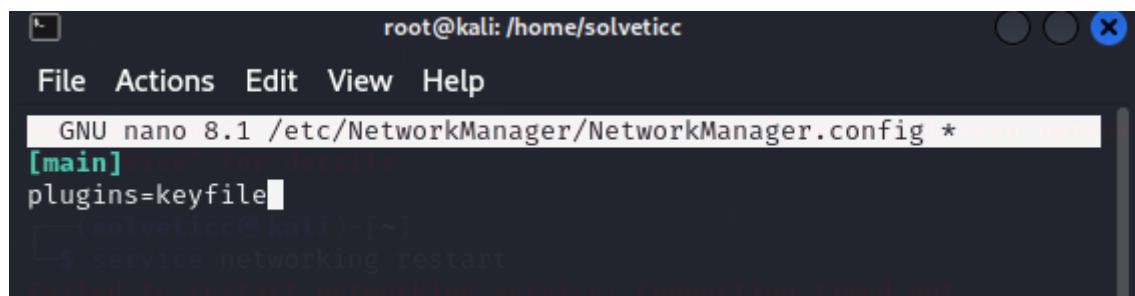


```
(root@kali)-[~/home/solveticc]
# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    group default qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    state UP group default qlen 1000
        link/ether 08:00:27:07:5d:12 brd ff:ff:ff:ff:ff:ff
        inet 192.168.1.84/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
            valid_lft 588sec preferred_lft 588sec
            inet6 fe80::a00:27ff:fe07:5d12/64 scope link noprefixroute
                valid_lft forever preferred_lft forever

(root@kali)-[~/home/solveticc]
#
```

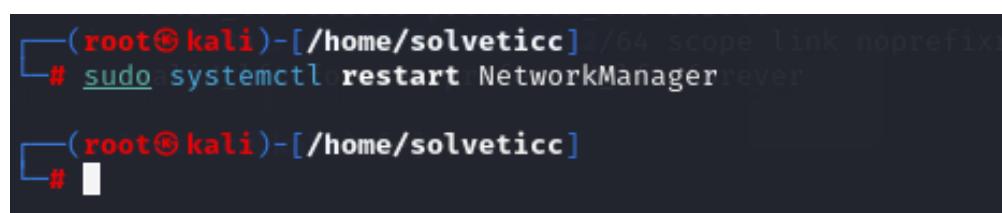
### Habilitar la configuración en NetworkManager

Si NetworkManager gestiona nuestras conexiones, desactivaremos su gestión automática para esta interfaz



```
root@kali: /home/solveticc
File Actions Edit View Help
  GNU nano 8.1 /etc/NetworkManager/NetworkManager.config *
[main] plugins=keyfile
[solveticc@kali] ~
$ service networking restart
Failed to restart networking.service: Connection timed out
```

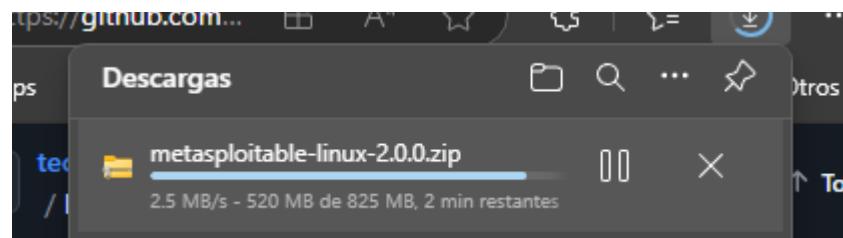
Y nada mas reiniciamos y listo



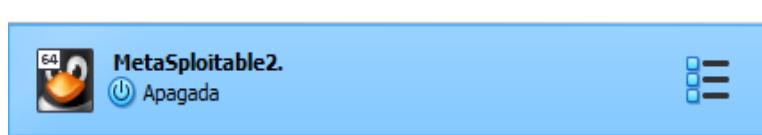
```
(root@kali)-[~/home/solveticc] /64 scope link noprefix
# sudo systemctl restart NetworkManager
[root@kali] ~
```

## CREAR UNA MÁQUINA VIRTUAL VULNERABLE POR DISEÑO COMO METASPLOITABLE2.

### Descarga de MetaSploitable2

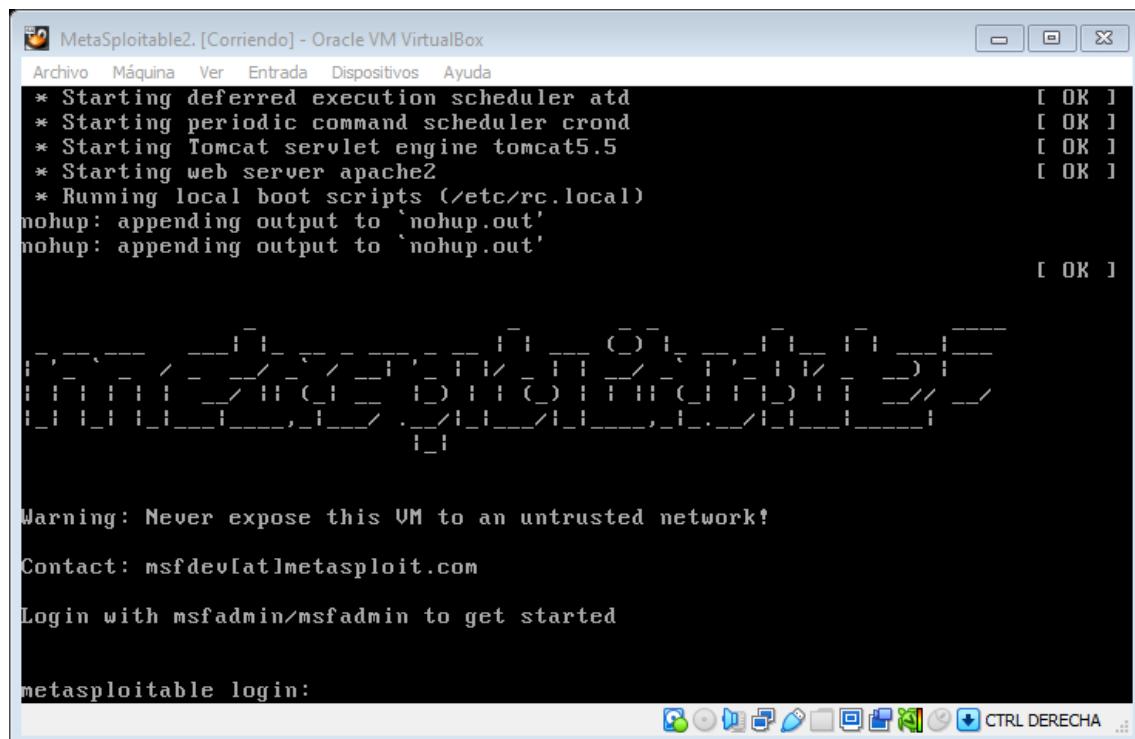


### Creación de maquina virtual para MetaSploitable2



### Inicio de MetaPloitable2 y asignación de ip estatica

Le damos en iniciar y esperamos a que la maquina arranque



Ponemos el usuario y la contraseña



Al poner el usuario y contraseña ingresamos, aquí ya estamos dentro.

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ _
```

Ahora vamos a Identificar el nombre del adaptador de red

Usando el siguiente comando para verificar el nombre del adaptador de red:  
ifconfig

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ifconfig  
eth0      Link encap:Ethernet HWaddr 08:00:27:b7:12:6a  
          inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0  
          inet6 addr: fe80::a00:27ff:feb7:126a/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
             RX packets:35 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:69 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:4641 (4.5 KB) TX bytes:7556 (7.3 KB)  
             Base address:0xd020 Memory:f0200000-f0220000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING MTU:16436 Metric:1  
             RX packets:131 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:131 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:38065 (37.1 KB) TX bytes:38065 (37.1 KB)  
  
msfadmin@metasploitable:~$
```

Ahora Editar el archivo de configuración de red

Abre el archivo de configuración de red para el adaptador identificado (eth0):  
sudo nano /etc/network/interfaces

```
GNU nano 2.0.7           File: /etc/network/interface           Modified  
auto eth0  
  
[ New File ]  
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit     ^J Justify   ^W Where Is  ^U Next Page ^U UnCut Text ^T To Spell
```

Ahora Buscamos la línea que define eth0 y cambia de dhcp a static. Y modificamos lo siguiente.

```
auto eth0
```

```
iface eth0 inet static
```

```
    address 192.168.1.74, netmask 255.255.255.0, gateway 192.168.1.1 y dns-nameservers 8.8.8.8 8.8.4.4
```

```
GNU nano 2.0.7          File: /etc/network/interface

auto eth0
iface eth0 inet static
address 192.168.1.74
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 8.8.8.8 8.8.4.4
```

Ahora vamos a Reiniciar el servicio de red

Guarda los cambios y reinicia el servicio de red para aplicar la nueva configuración: sudo /etc/init.d/networking restart

```
msfadmin@metasploitable:~$ sudo /etc/init.d/networking restart
```

Ahora vamos a verificar la nueva dirección IP

Usa el comando ifconfig o ip addr para verificar si se asignó la dirección IP estática.

```
Copyright 2004-2007 Internet Systems Consortium.
All rights reserved.
For info, please visit http://www.isc.org/sw/dhcp/

Listening on LPF/eth0/08:00:27:b7:12:6a
Sending on LPF/eth0/08:00:27:b7:12:6a
Sending on Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 6
DHCPoffer of 10.0.2.15 from 10.0.2.2
DHCPREQUEST of 10.0.2.15 on eth0 to 255.255.255.255 port 67
DHCPACK of 10.0.2.15 from 10.0.2.2
bound to 10.0.2.15 -- renewal in 32882 seconds.                                [ OK ]
msfadmin@metasploitable:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:b7:12:6a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global eth0
        inet6 fe80::a00:27ff:feb7:126a/64 scope link
            valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ ping 192
```

Reiniciamos el servicio de red para aplicar la nueva configuración:

```
sudo /etc/init.d/networking restart
```

Y para comprobar que se hizo correctamente hacemos ping

```
64 bytes from 192.168.1.78: icmp_seq=7 ttl=127 time=1.34 ms
64 bytes from 192.168.1.78: icmp_seq=8 ttl=127 time=1.79 ms
64 bytes from 192.168.1.78: icmp_seq=9 ttl=127 time=1.71 ms
64 bytes from 192.168.1.78: icmp_seq=10 ttl=127 time=1.44 ms
64 bytes from 192.168.1.78: icmp_seq=11 ttl=127 time=2.56 ms
64 bytes from 192.168.1.78: icmp_seq=12 ttl=127 time=1.60 ms
64 bytes from 192.168.1.78: icmp_seq=13 ttl=127 time=1.81 ms
64 bytes from 192.168.1.78: icmp_seq=14 ttl=127 time=1.14 ms
64 bytes from 192.168.1.78: icmp_seq=15 ttl=127 time=2.77 ms
64 bytes from 192.168.1.78: icmp_seq=16 ttl=127 time=2.30 ms
64 bytes from 192.168.1.78: icmp_seq=17 ttl=127 time=1.33 ms
64 bytes from 192.168.1.78: icmp_seq=18 ttl=127 time=1.66 ms
64 bytes from 192.168.1.78: icmp_seq=19 ttl=127 time=1.30 ms
64 bytes from 192.168.1.78: icmp_seq=20 ttl=127 time=2.24 ms
64 bytes from 192.168.1.78: icmp_seq=21 ttl=127 time=1.25 ms
64 bytes from 192.168.1.78: icmp_seq=22 ttl=127 time=1.50 ms
64 bytes from 192.168.1.78: icmp_seq=23 ttl=127 time=1.41 ms
64 bytes from 192.168.1.78: icmp_seq=24 ttl=127 time=2.15 ms
64 bytes from 192.168.1.78: icmp_seq=25 ttl=127 time=2.11 ms
64 bytes from 192.168.1.78: icmp_seq=26 ttl=127 time=1.71 ms
64 bytes from 192.168.1.78: icmp_seq=27 ttl=127 time=1.43 ms
64 bytes from 192.168.1.78: icmp_seq=28 ttl=127 time=1.45 ms
64 bytes from 192.168.1.78: icmp_seq=29 ttl=127 time=2.11 ms
64 bytes from 192.168.1.78: icmp_seq=30 ttl=127 time=1.51 ms
```

## PING SATISFACTORIO ENTRE LAS MÁQUINAS VIRTUALES.

### Configuración de interfaz de red de MetaSploitable2.

```
GNU nano 2.0.7          File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.1.70
netmask 255.255.255.0
gateway 192.168.1.1
```

[ Wrote 15 lines ]

```
msfadmin@metasploitable:~$ _
```

Una vez que se guardó vamos a reiniciar el servidor de red para aplicar los cambios. Para reiniciarlo ocupamos el comando sudo nano /etc/network/interfaces

```
[ Wrote 15 lines ]
msfadmin@metasploitable:~$ sudo systemctl restart networking
```

Le vamos asignar una nueva

Ya al reiniciar podemos ver que ya se cambió correctamente

```
fadmin@metasploitable:~$ ifconfig
h0      Link encap:Ethernet HWaddr 08:00:27:b7:12:6a
        inet addr:192.168.1.70 Bcast:192.168.1.255 Mask:255.255.255.0
        inet6 addr: fe80::a00:27ff:feb7:126a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:36 errors:0 dropped:0 overruns:0 frame:0
          TX packets:128 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4705 (4.5 KB) TX bytes:14823 (14.4 KB)
          Base address:0xd020 Memory:f0200000-f0220000

        Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:193 errors:0 dropped:0 overruns:0 frame:0
          TX packets:193 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:62085 (60.6 KB) TX bytes:62085 (60.6 KB)
```

### Configuración de reglas del firewall para permitir el tráfico de ping

Luego vamos a configurar las reglas de firewall para permitir el tráfico de ping.

```
GNU nano 2.0.7           File: /etc/iptables/rules.v4           Modified

sudo iptables -A INPUT -p icmp --icmp-type 8 -j ACCEPT
sudo iptables -A OUTPUT -p icmp --icmp-type 0 -j ACCEPT
```

Checamos en la maquina física si podemos hacer ping, esto solo es para comprobar. Y efectivamente hace ping así que todo saldrá bien

```
Símbolo del sistema Microsoft Windows [Versión 10.0.22631.4317]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\reyes>ping 192.168.1.70

Haciendo ping a 192.168.1.70 con 32 bytes de datos:
Respuesta desde 192.168.1.70: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 192.168.1.70:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
        (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\Users\reyes>
```

Ahora realizaremos lo mismo con Kali Linux, le asignamos una nueva dirección.

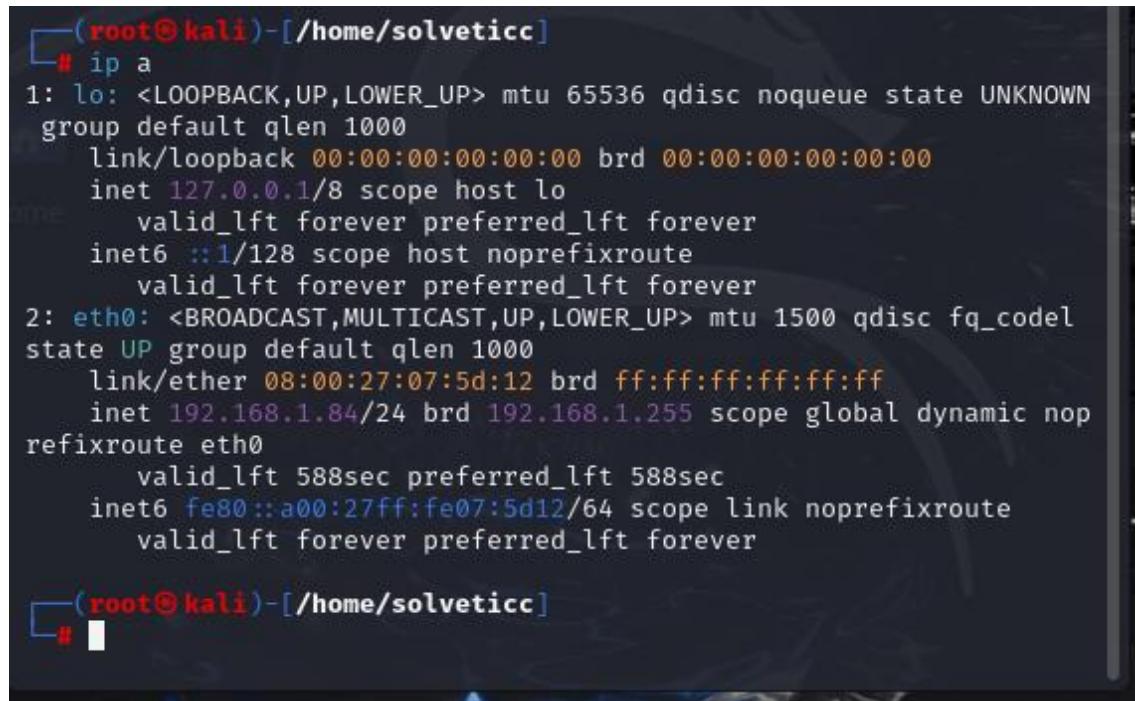
## Configuración de interfaces de red De Kali Linux

```
root@kali: /home/solveticc
File Actions Edit View Help
GNU nano 8.1          /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
#
source /etc/network/interfaces.d/*
#
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.1.84
    netmask 255.255.255.0
    gateway 192.168.1.255
    dns-nameservers 8.8.8.8.8.4.4

[ Read 16 lines ]
^G Help      ^O Write Out  ^F Where Is  ^K Cut      ^T Execute
^X Exit      ^R Read File  ^\ Replace   ^U Paste    ^J Justify
```

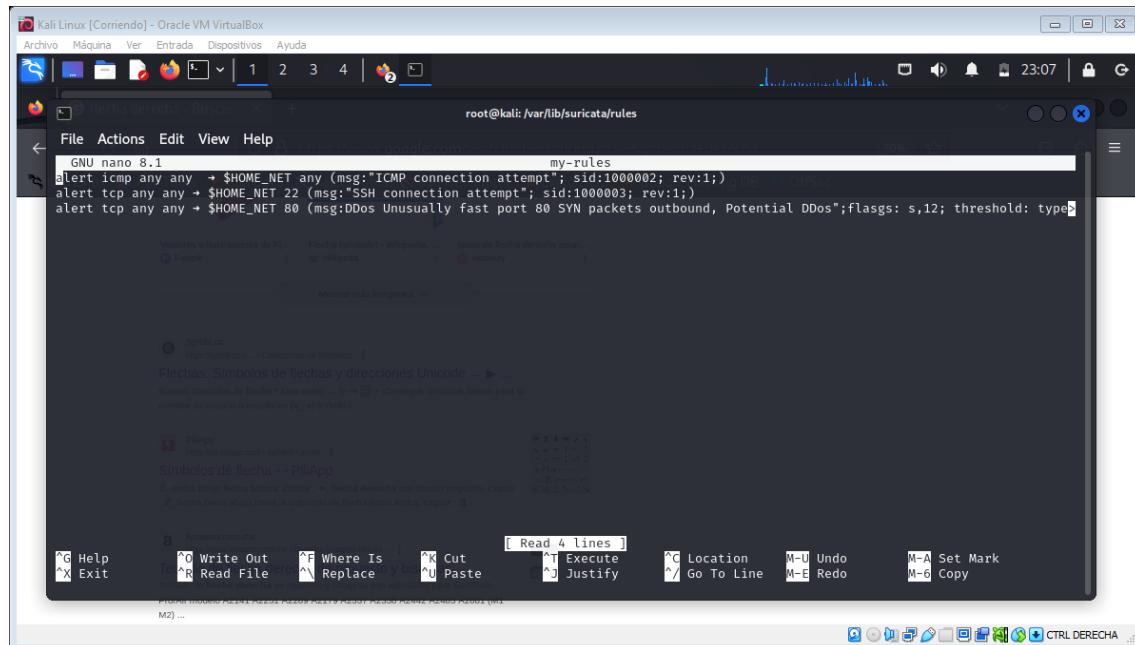
Al cambiar las direcciones IP lo guardamos y vemos que si se guarda correctamente.



```
(root@kali)-[~/home/solveticc]
# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    group default qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    state UP group default qlen 1000
        link/ether 08:00:27:07:5d:12 brd ff:ff:ff:ff:ff:ff
        inet 192.168.1.84/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
            valid_lft 588sec preferred_lft 588sec
            inet6 fe80::a00:27ff:fe07:5d12/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
```

## Configuración de reglas para permitir el ping en Kali Linux

Igualmente se configuran las reglas para permitir el tráfico de ping en Kali Linux



## REALIZACIÓN DE PING ENTRE LAS MÁQUINAS VIRTUALES.

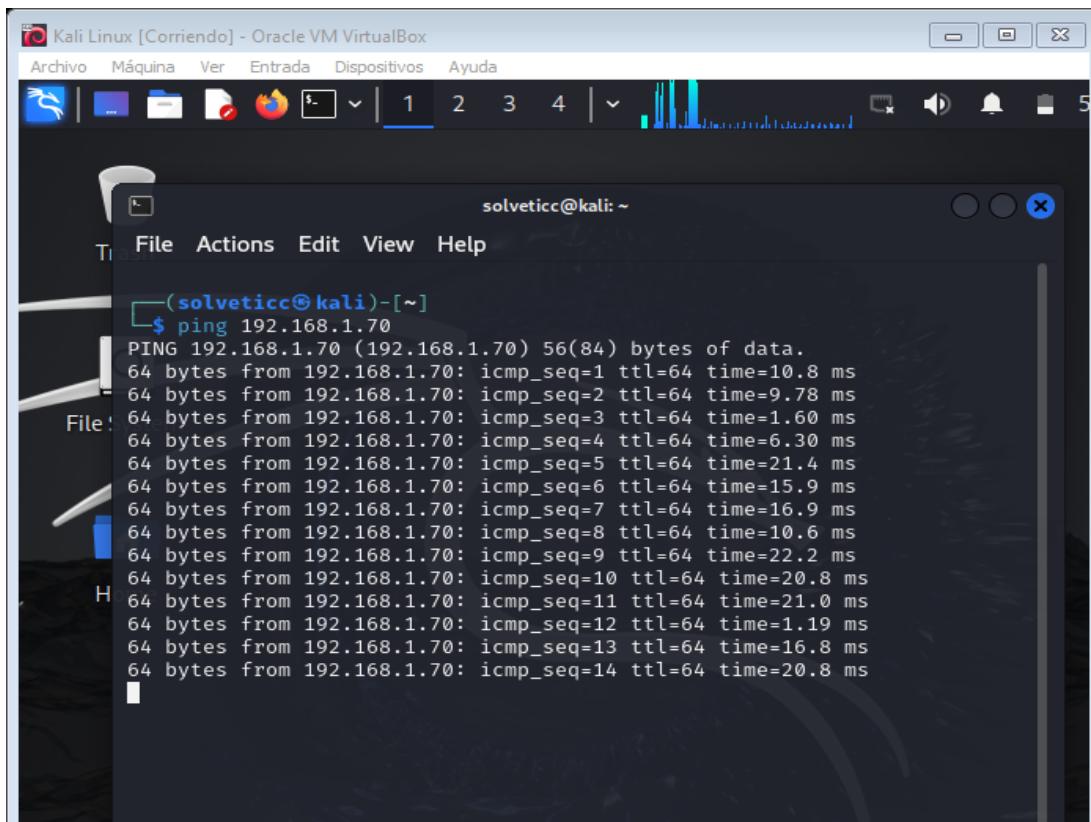
### Realización de ping desde MetaSplittables2 a Kali Linux

Ahora hacemos pin desde MetaSploitables a Kali Linux y la ip de Kali Linux es 192.168.1.84

```
--- 192.168.1.84 ping statistics ---
27 packets transmitted, 27 received, 0% packet loss, time 26032ms
rtt min/avg/max/mdev = 0.675/1.867/5.642/0.922 ms
msfadmin@metasploitable:~$ ping 192.168.1.84
PING 192.168.1.84 (192.168.1.84) 56(84) bytes of data.
64 bytes from 192.168.1.84: icmp_seq=1 ttl=64 time=1.59 ms
64 bytes from 192.168.1.84: icmp_seq=2 ttl=64 time=2.87 ms
64 bytes from 192.168.1.84: icmp_seq=3 ttl=64 time=1.59 ms
64 bytes from 192.168.1.84: icmp_seq=4 ttl=64 time=1.47 ms
64 bytes from 192.168.1.84: icmp_seq=5 ttl=64 time=2.10 ms
64 bytes from 192.168.1.84: icmp_seq=6 ttl=64 time=2.23 ms
```

### Realización de ping desde Kali Linux a MetaSplittables2

Ahora voy hacer lo mismo, pero de Kali Linux a MetaSploitables2 y la dirección de meta es 192.168.70, y efectivamente hace pin eso quiere decir que todo fue correctamente configurado.



## CONCLUSIÓN

Lo primero que se realizó fue la instalación de VirtualBox la cual la omitimos porque ya lo tenemos instalado.

Después realizamos la instalación de OpenSense en VirtualBox

La instalación de OpenSense en una máquina virtual permitió establecer una plataforma robusta y confiable para la gestión y control de redes mediante un firewall de código abierto. Durante el proceso, se configuró una máquina virtual con dos interfaces de red, lo que habilitó la segmentación entre las conexiones LAN y WAN. Posteriormente, se asignaron las interfaces de red y se definió una dirección IP estática para facilitar el acceso y la administración desde la interfaz web.

Igualmente se descargó Kali Linux y se instaló

En este caso pues logramos instalar y configurar Kali Linux en una máquina virtual, proporcionando un entorno seguro y controlado para realizar pruebas de seguridad informática. Durante el proceso, se asignó una dirección IP estática al sistema para facilitar la conectividad y administración de la red.

Ahí mismo implementamos un sistema de detección de intrusos utilizando herramientas como Suricata. Esto incluyó la configuración de reglas específicas para identificar actividades sospechosas en la red, así como la personalización de alertas para monitorear eventos de seguridad en tiempo real.

Con esta configuración, Kali Linux se convierte en una potente herramienta para el análisis y la protección de redes, permitiendo no solo identificar amenazas, sino también tomar medidas proactivas para mitigarlas. Este entorno es ideal para el aprendizaje práctico y el fortalecimiento de habilidades en ciberseguridad.

Después descargamos y instalamos MetaSploitable2

Se realizó la instalación y configuración de MetaSploitable2, descargando la máquina desde una fuente confiable, descomprimiéndola e importándola a un hipervisor como VirtualBox o VMware. Se ajustó la red para operar en un entorno controlado, permitiendo prácticas seguras de auditoría y pruebas de penetración.

## Configuración de Ping entre Máquinas Virtuales

En este procedimiento, configuramos las interfaces de red, ajustamos las reglas de firewall y realizamos pruebas de conectividad mediante el comando ping entre dos máquinas virtuales: Kali Linux y Metasploitable2. Este ejercicio nos permitió garantizar que ambas máquinas pudieran comunicarse correctamente dentro de una misma red virtual o física.