

02/06/2019

---

---

---

# Virtual Lab for Malware Analysis Set Up

*Introducción. “Good planning at the beginning always means less troubles in the long run”.*

---

---

---

Elvira Castillo Fernández

# Virtual Lab for Malware Analysis Set Up

**Introducción.** *“Good planning at the beginning always means less troubles in the long run”.*

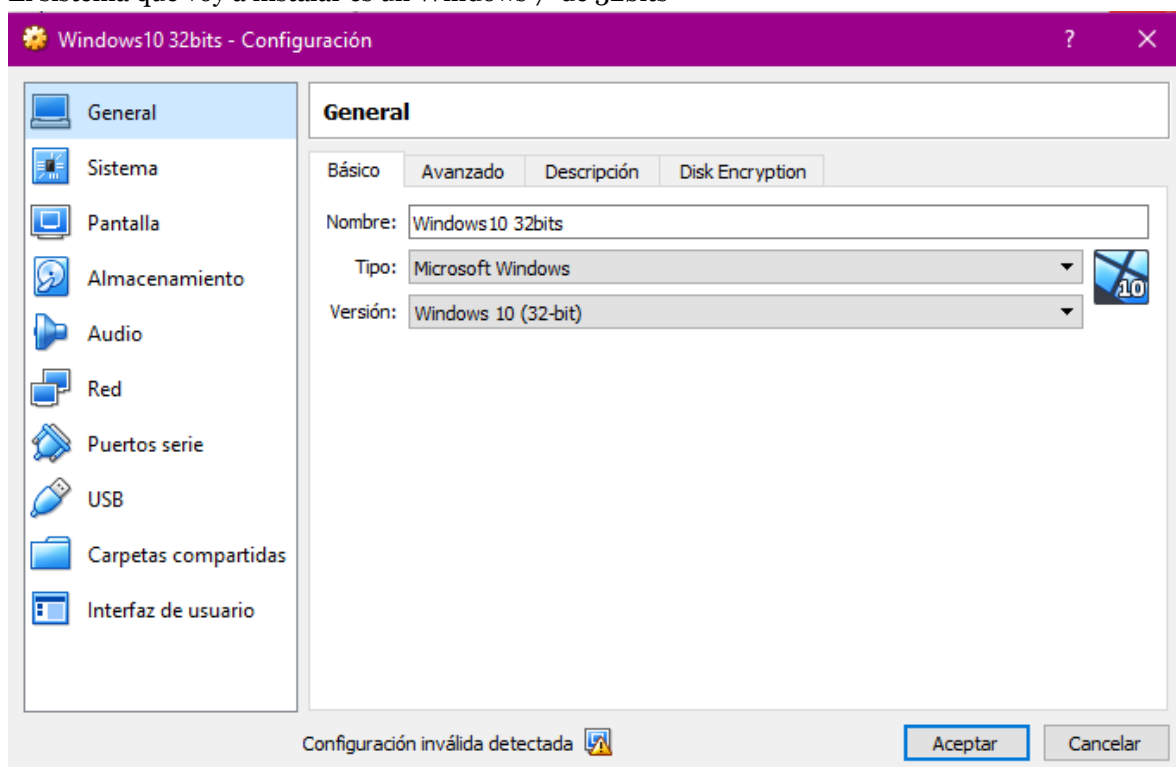
En este guión, vamos a explicar paso a paso como debemos configurar el entorno sobre el cual vamos a realizar los análisis de malware.

Dado que los ficheros que vamos a analizar son posibles archivos de malware para Windows, necesitaremos una máquina con Windows 10 32bits y 64bits. Los instalaremos sobre la plataforma de virtualización VMWare o bien VirtualBox. Y una segunda máquina con Remnux que estará en red con ambos equipos Windows.

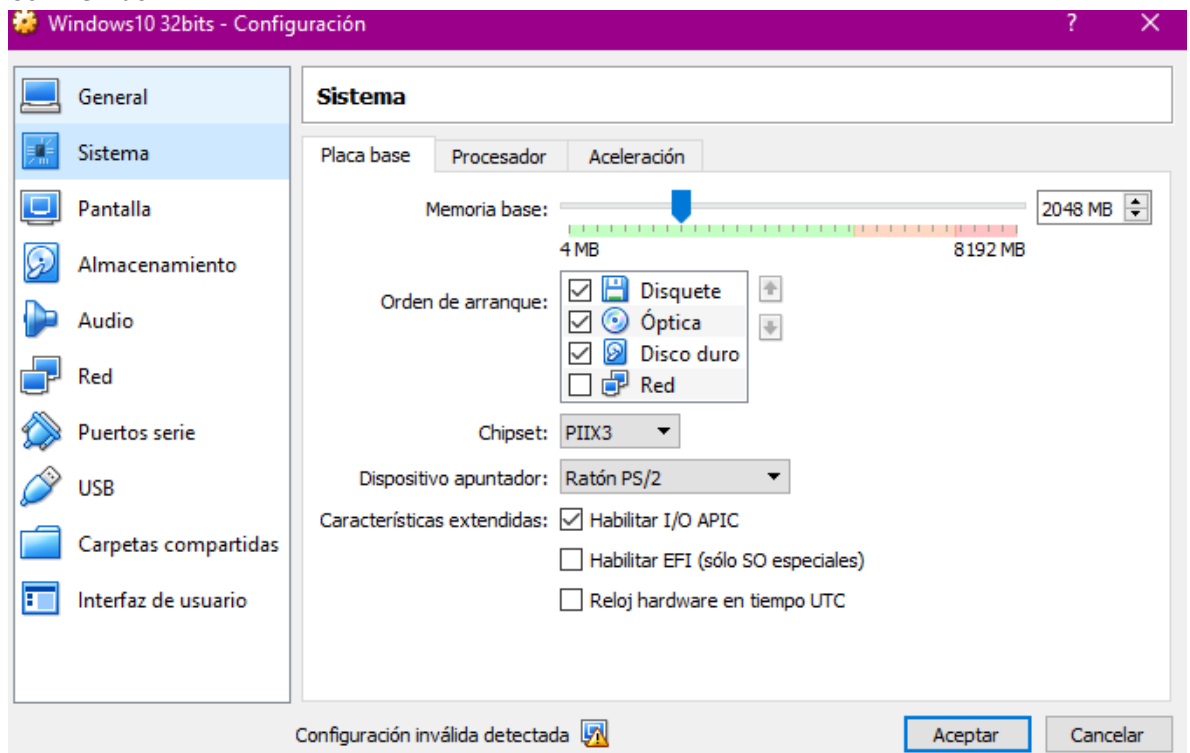
Dado que yo voy a usar la plataforma de virtualización VirtualBox, voy a explicar como hay que configurar la red entre las dos máquinas virtuales lo voy a hacer para Windows de 32bits pero se hace igual para el de 64bits.

Vamos a ver la configuración que hay que hacer para crear la maquina virtual:

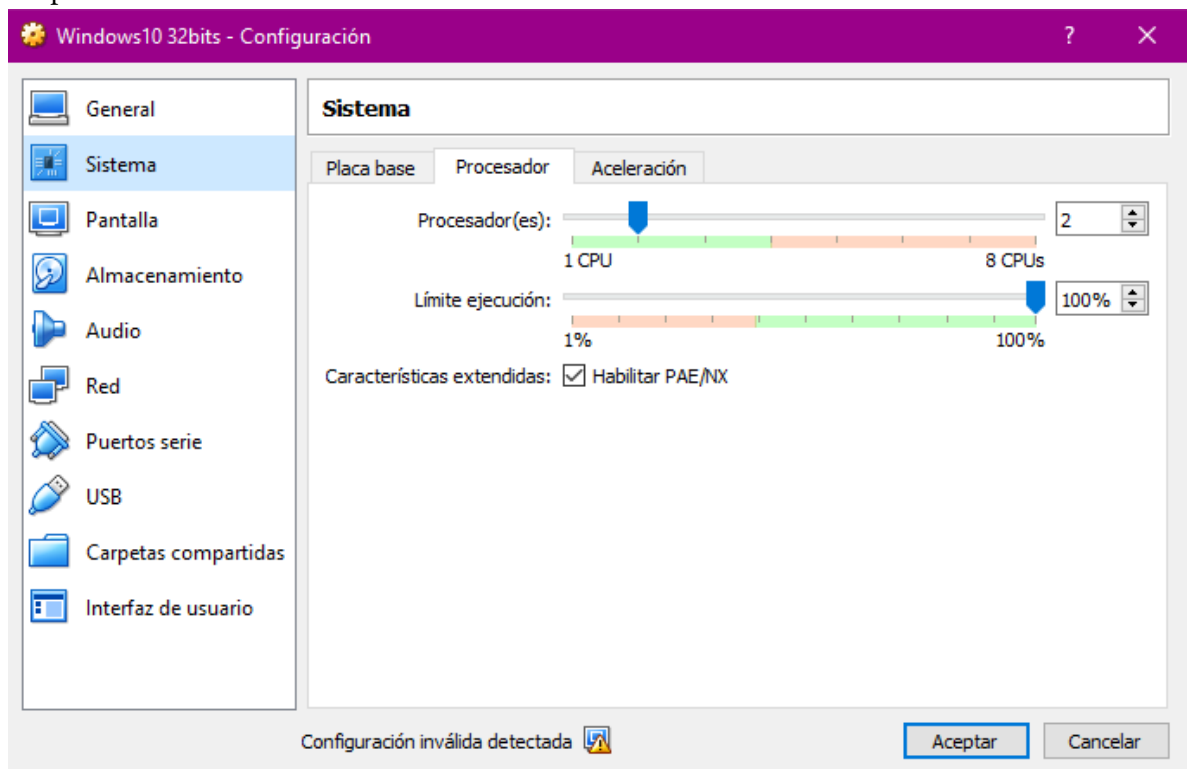
- El sistema que voy a instalar es un Windows 7 de 32bits



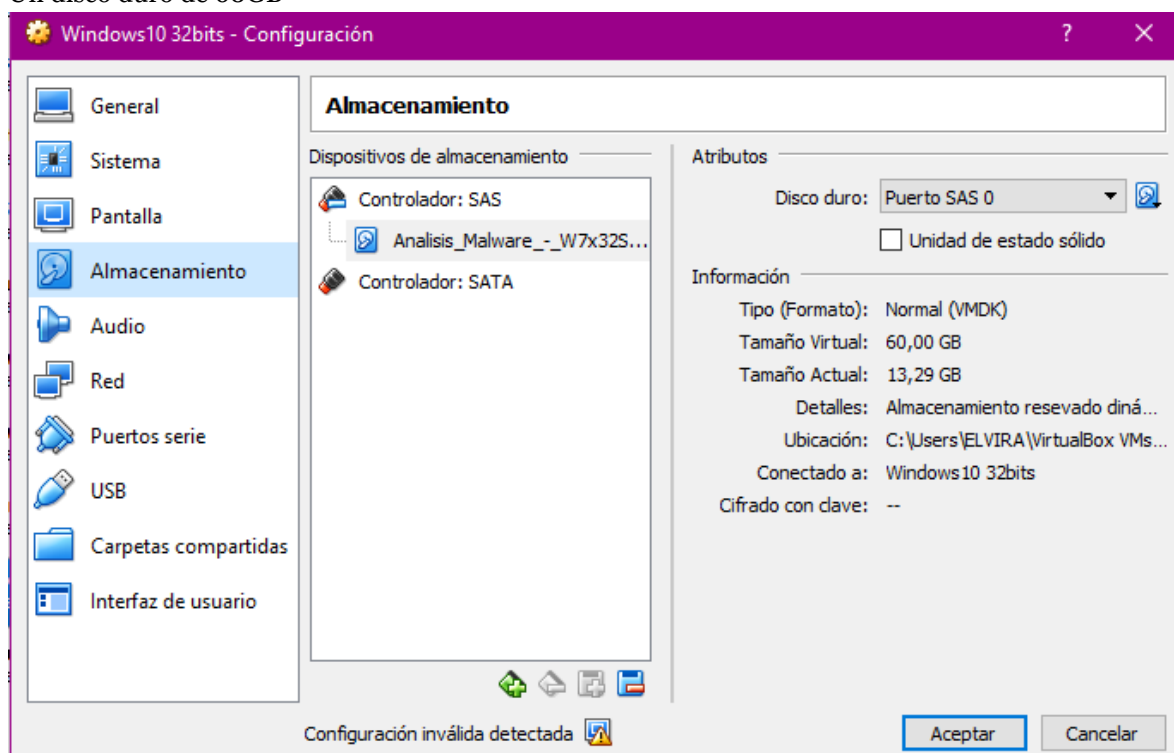
- Con 2GB de RAM



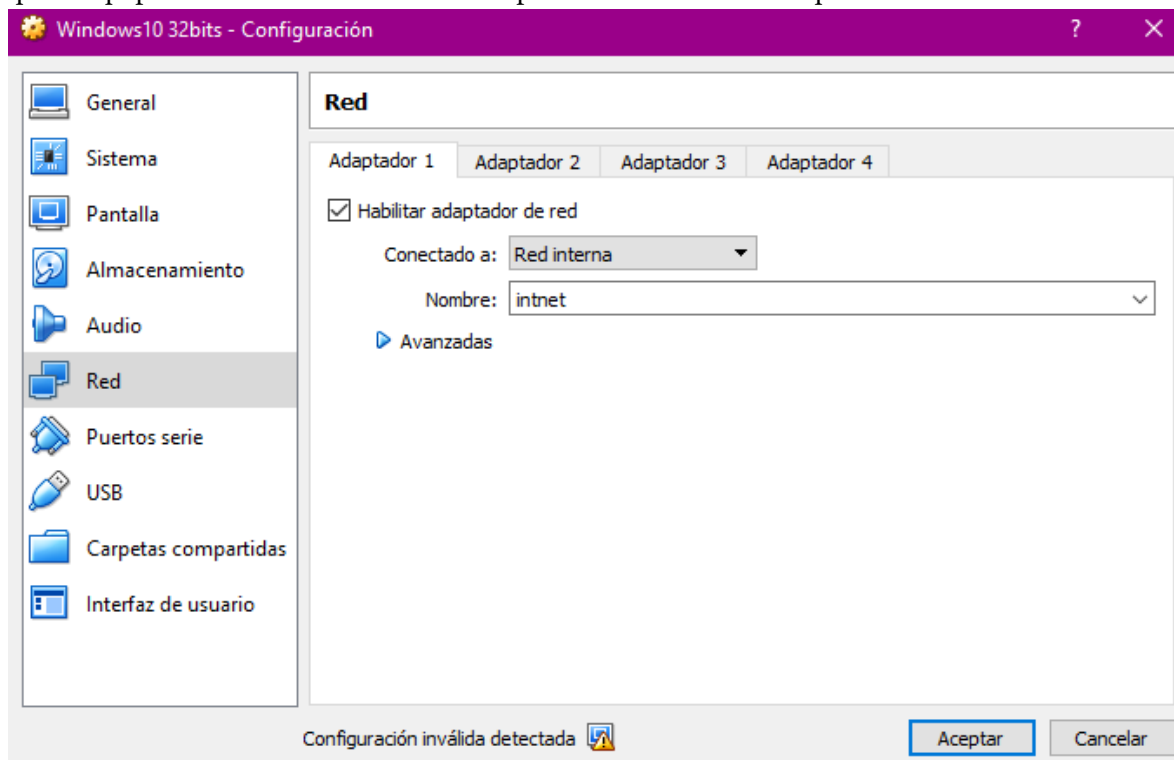
- Un procesador con 2 cores



- Un disco duro de 60GB

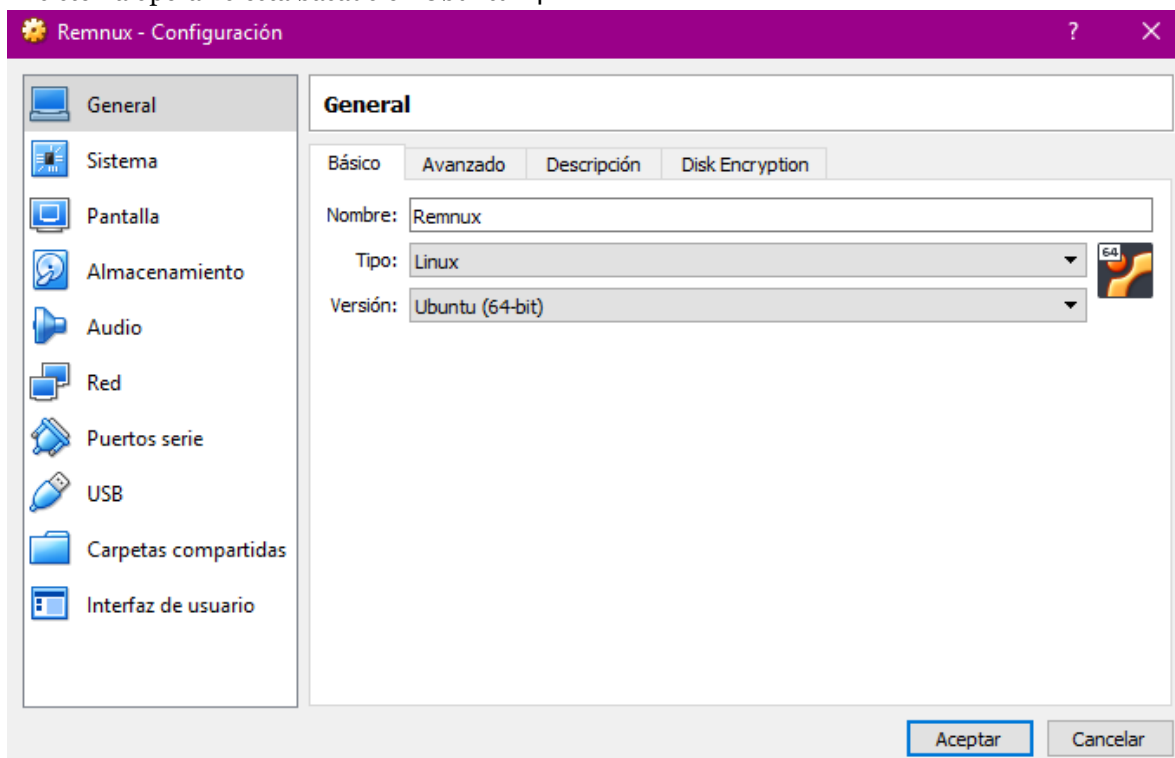


- La configuración de la tarjeta de red es imprescindible que sea **Red Interna** para que la máquina que puede llegar a estar comprometida tras la ejecución del malware no comprometa al equipo anfitrión ni mucho menos los posibles equipos que haya conectados en la misma red que el equipo anfitrión. Sólo tendré un adaptador de red en esta máquina.

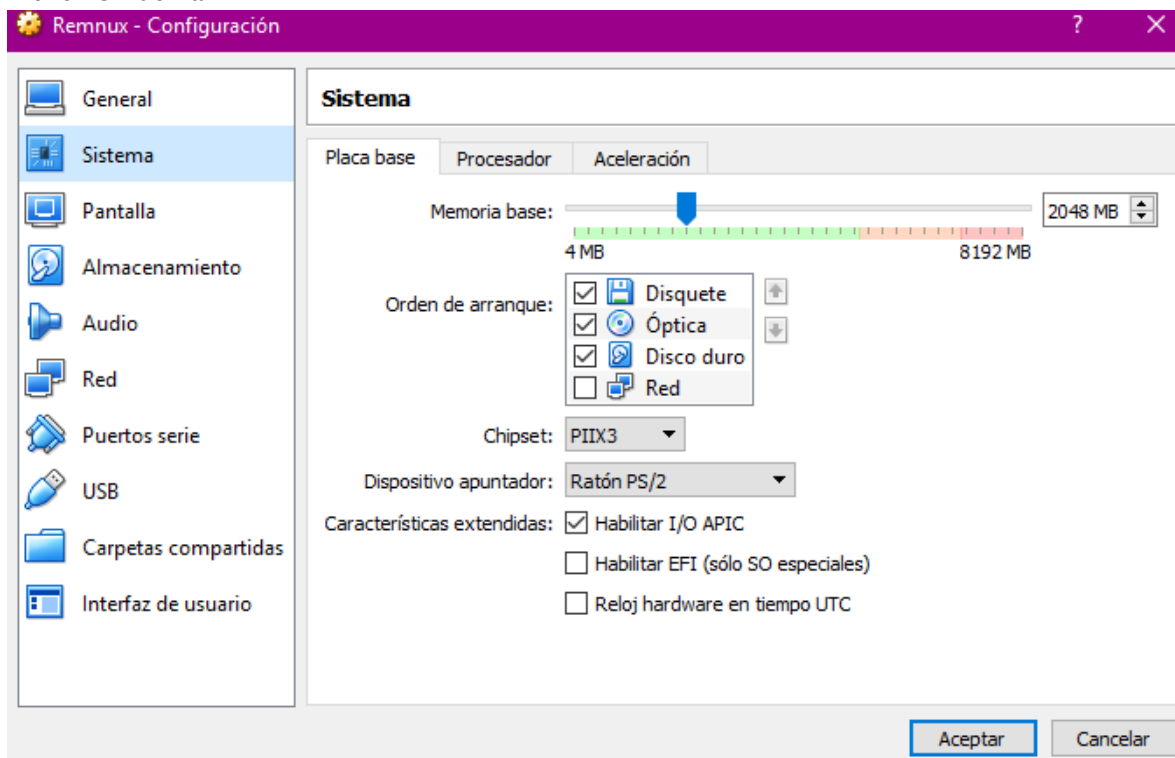


La otra máquina que crearemos será la máquina donde correrá Remnux debe tener la siguiente configuración:

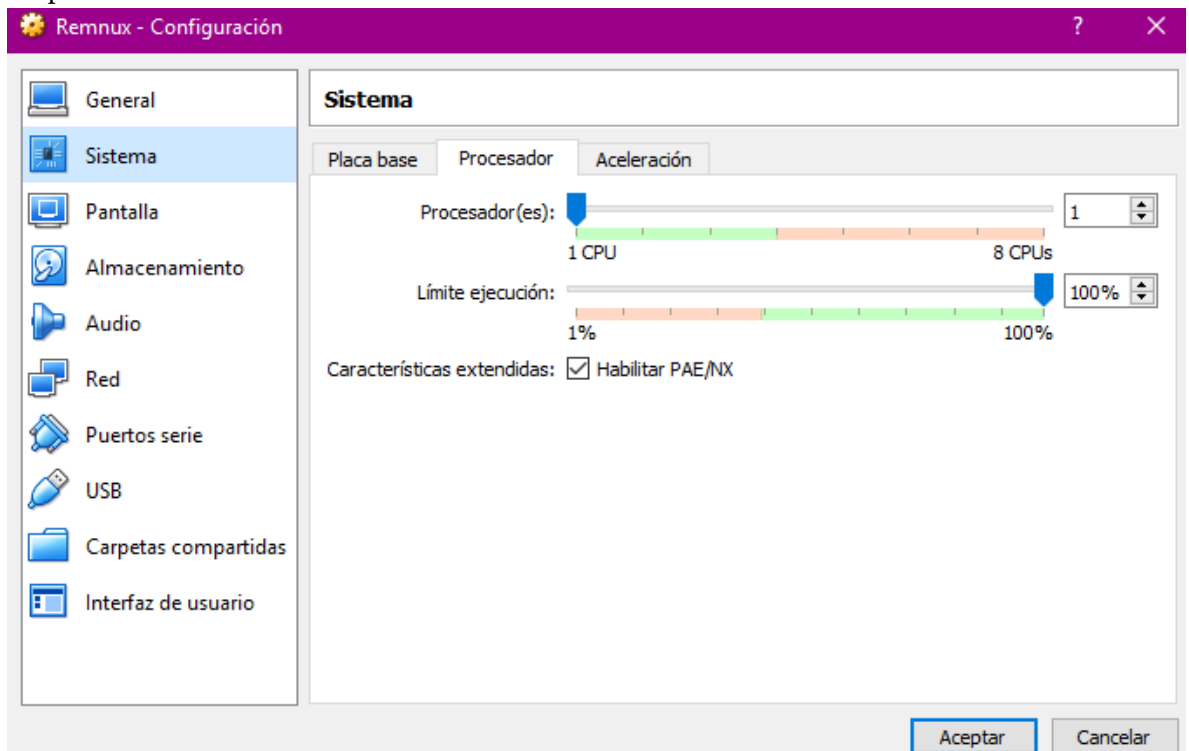
- El sistema operativo está basado en Ubuntu 14



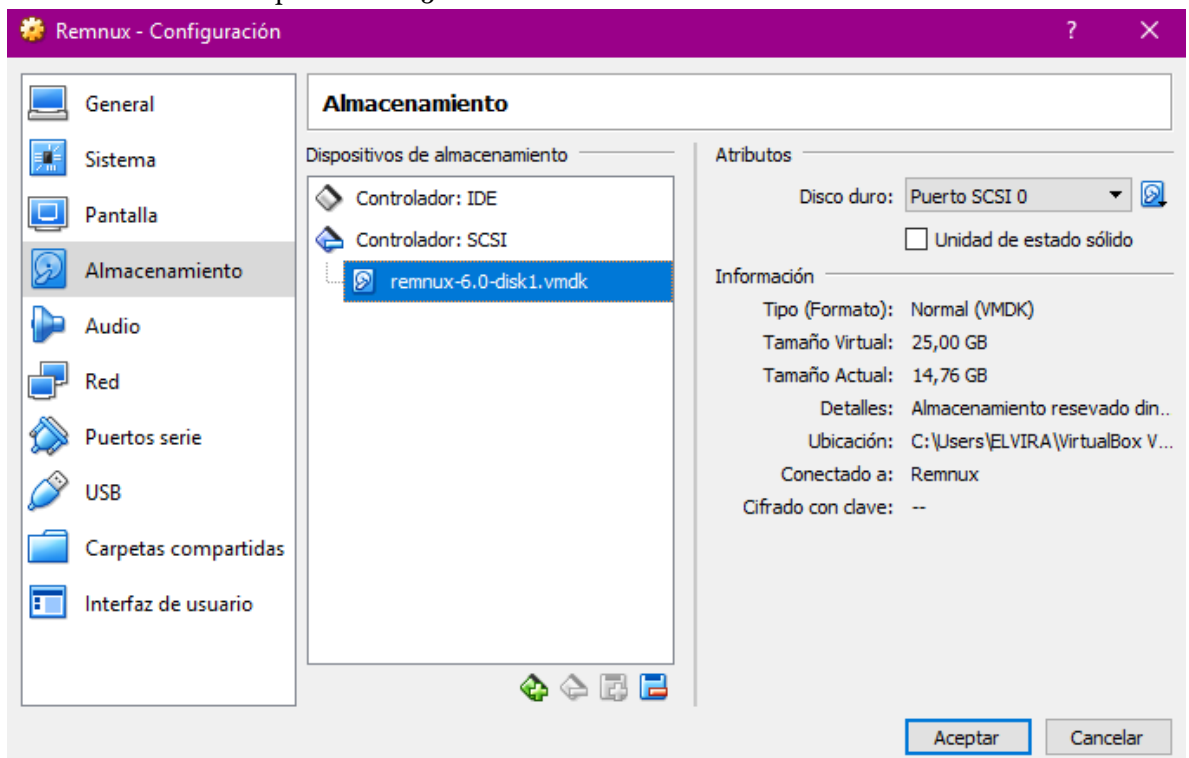
- Tiene 2GB de Ram



- Y 1 procesador con 1core

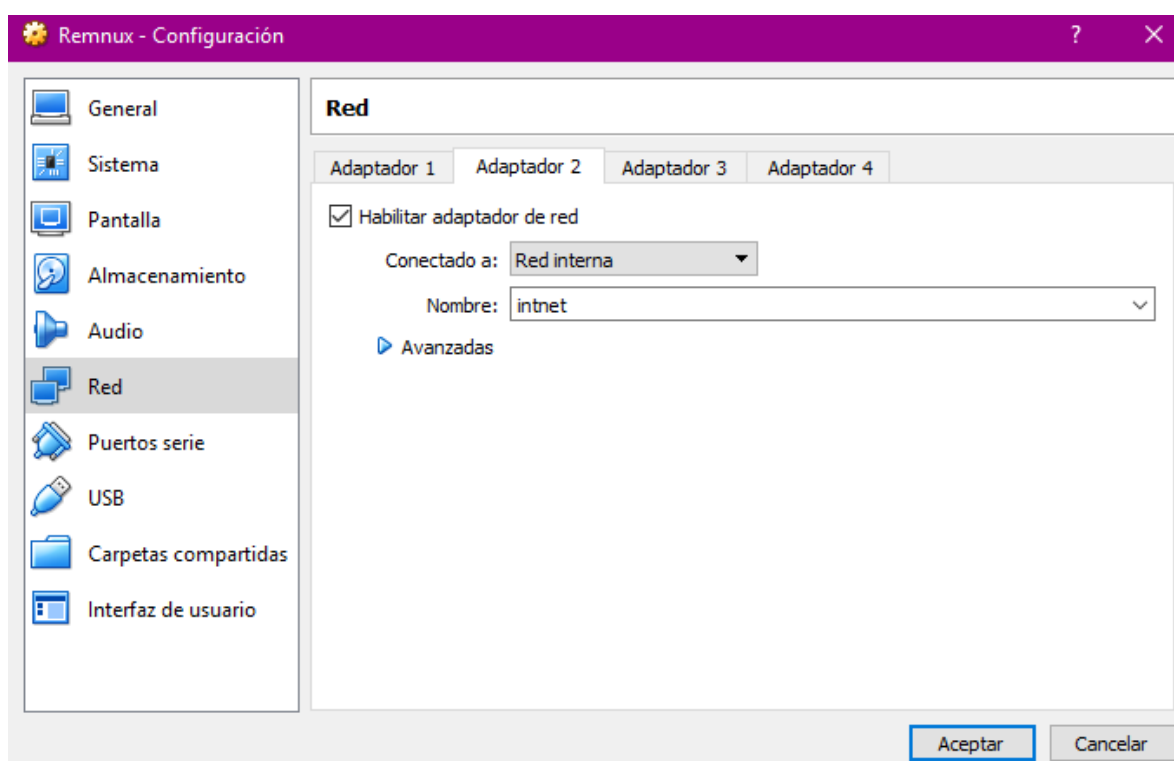
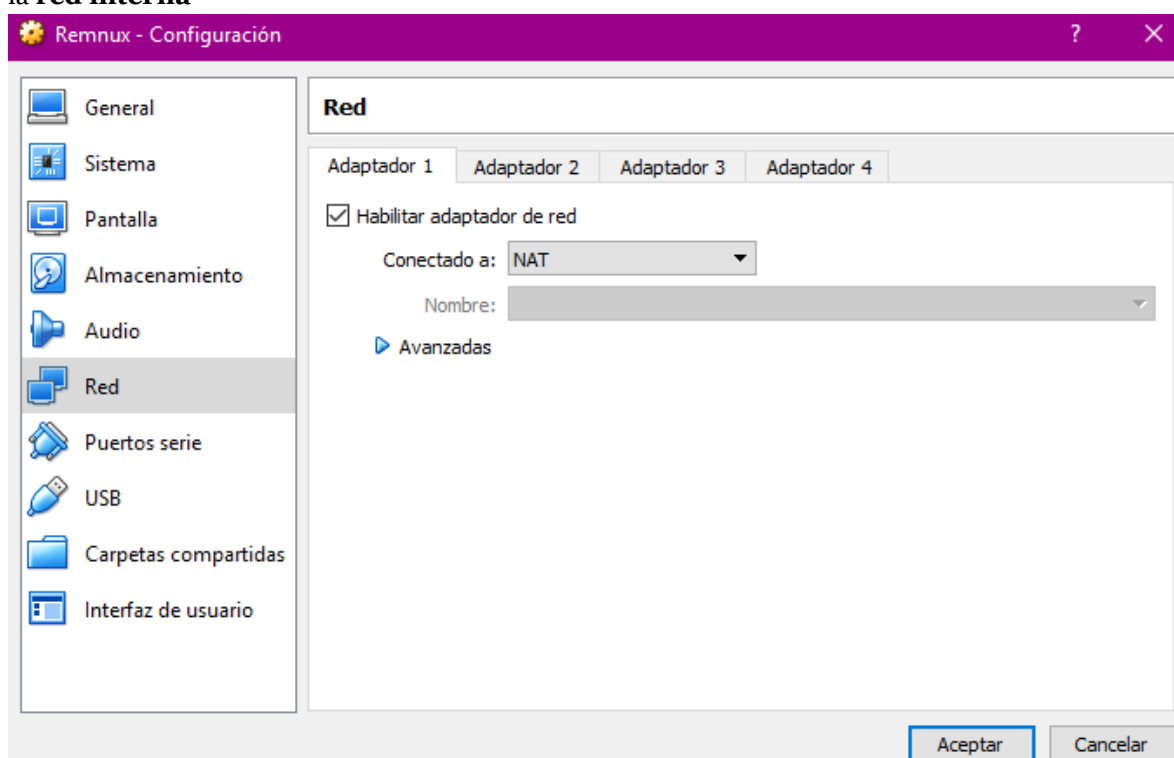


- El disco duro de la máquina es de 25GB

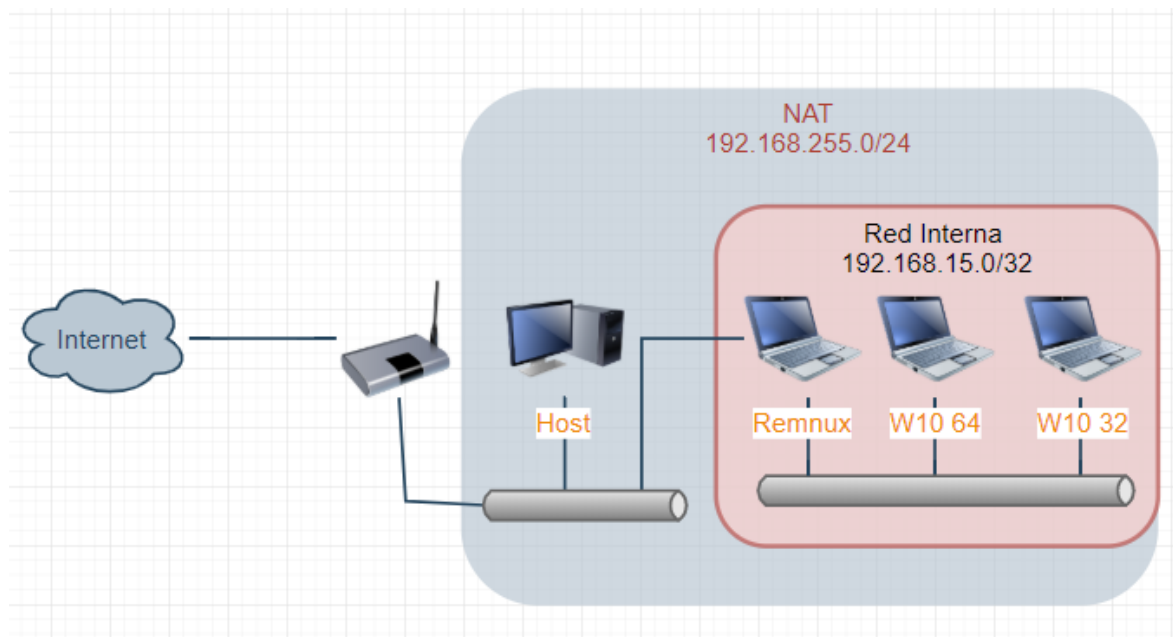


- En esta máquina podemos tener 2 adaptadores de red uno en nat y otro con la configuración de la red interna. Pero más tarde explicaremos la configuración del cortafuegos para no dejar que salga nada de la red interna. Por ahora vamos a dejar habilitado el **adaptador1** con un **NAT** (para poder actualizar aplicaciones) y el **adaptador 2** que será el que está conectado a

## la red interna



El esquema de red implementado es el siguiente:



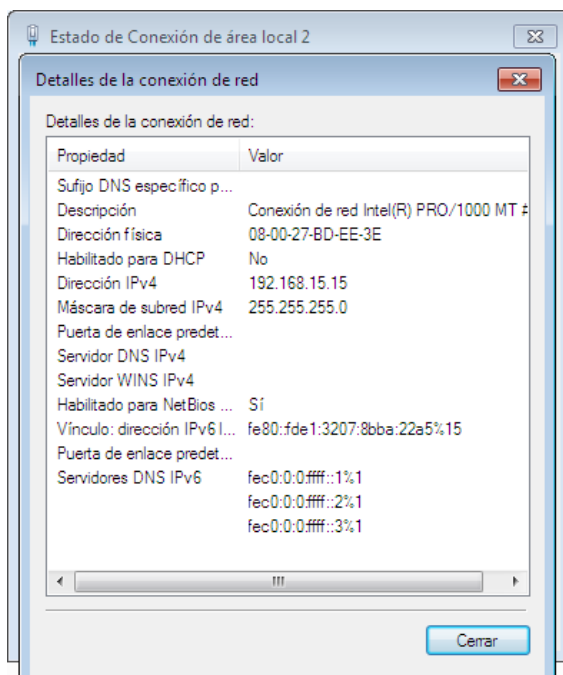
Como vemos hay dos redes bien diferenciadas una interna en la que están los equipos Windows con Remnux y no tienen salida a internet ni acceso a la red del host. La configuración de red en Remnux es la siguiente:

```
remnux@remnux: ~
File Edit Tabs Help
remnux@remnux:~$ sudo ifconfig eth0 192.168.15.16 netmask 255.255.255.0
remnux@remnux:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f7:c9:76
          inet addr:192.168.15.16  Bcast:192.168.15.255  Mask:
255.255.255.0
          inet6 addr: fe80::a00:27ff:fef7:c976/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:202 errors:0 dropped:0 overruns:0 frame:0
TX packets:98 errors:0 dropped:0 overruns:0 carrier:
0
          collisions:0 txqueuelen:1000
          RX bytes:30925 (30.9 KB)  TX bytes:31428 (31.4 KB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:5b:b1:01
          inet addr:192.168.225.1  Bcast:192.168.225.255  Mask
:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe5b:b101/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
```

Como vemos esta la tarjeta de red eth0 que es la que está conectada a la red interna y la tarjeta eth1 que es la que hace NAT con el host.





El equipo con Windows solo dispone de una tarjeta de red que es la que está conectada a la red interna. Como no va a tener ni a poder salir a internet, no le he puesto la puerta de enlace ni los dns. Para el análisis dinámico probablemente tendremos que simular algunos servicios como el dns, pero por ahora para la configuración inicial no es necesario.

Comprobamos que las máquinas están en red y sobre todo que la máquina con Windows que es la que contiene los malware y la que usaremos para ejecutar y analizar los archivos malicioso, no puede salir a ninguna red que no sea la interna.

```

C:\> Administrador: C:\Windows\system32\cmd.exe
^C
C:\Users\Practicas>ping 192.168.15.16

Haciendo ping a 192.168.15.16 con 32 bytes de datos:
Respuesta desde 192.168.15.16: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.15.16: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.15.16: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.15.16: bytes=32 tiempo<1m TTL=64

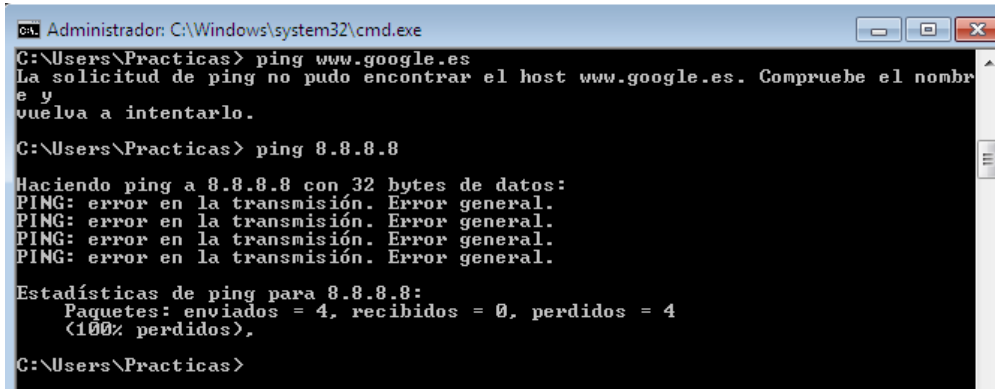
Estadísticas de ping para 192.168.15.16:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\Users\Practicas>ping 192.168.255.1

Haciendo ping a 192.168.255.1 con 32 bytes de datos:
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.

Estadísticas de ping para 192.168.255.1:
    Paquetes: enviados = 4, recibidos = 0, perdidos = 4
    (100% perdidos),

```



```

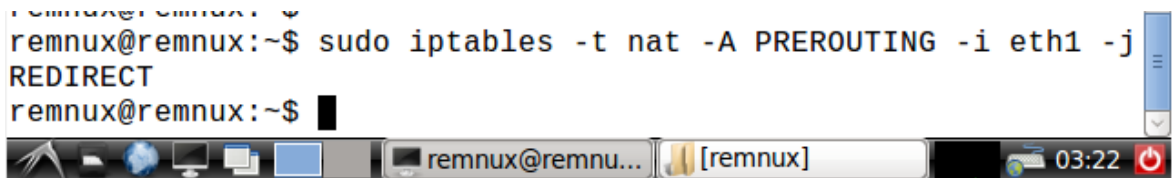
C:\Windows\system32\cmd.exe
C:\Users\Practicas> ping www.google.es
La solicitud de ping no pudo encontrar el host www.google.es. Compruebe el nombre y vuelva a intentarlo.

C:\Users\Practicas> ping 8.8.8.8
Haciendo ping a 8.8.8.8 con 32 bytes de datos:
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.
PING: error en la transmisión. Error general.

Estadísticas de ping para 8.8.8.8:
    Paquetes: enviados = 4, recibidos = 0, perdidos = 4
              (100% perdidos),
C:\Users\Practicas>

```

Para más seguridad en la máquina con Remnux, vamos a añadir una regla a iptables para que si en algún momento desde alguna máquina Windows quieren intentar salir através del interfaz de Remnux lo bloquee.



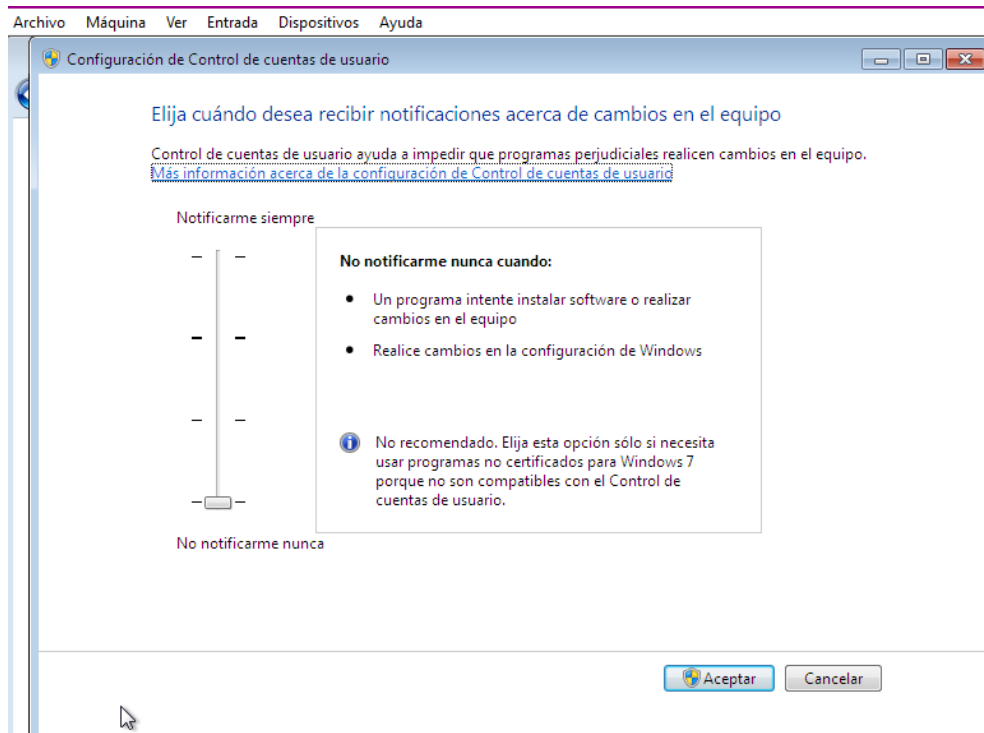
```

remnux@remnux:~$ sudo iptables -t nat -A PREROUTING -i eth1 -j REDIRECT
remnux@remnux:~$

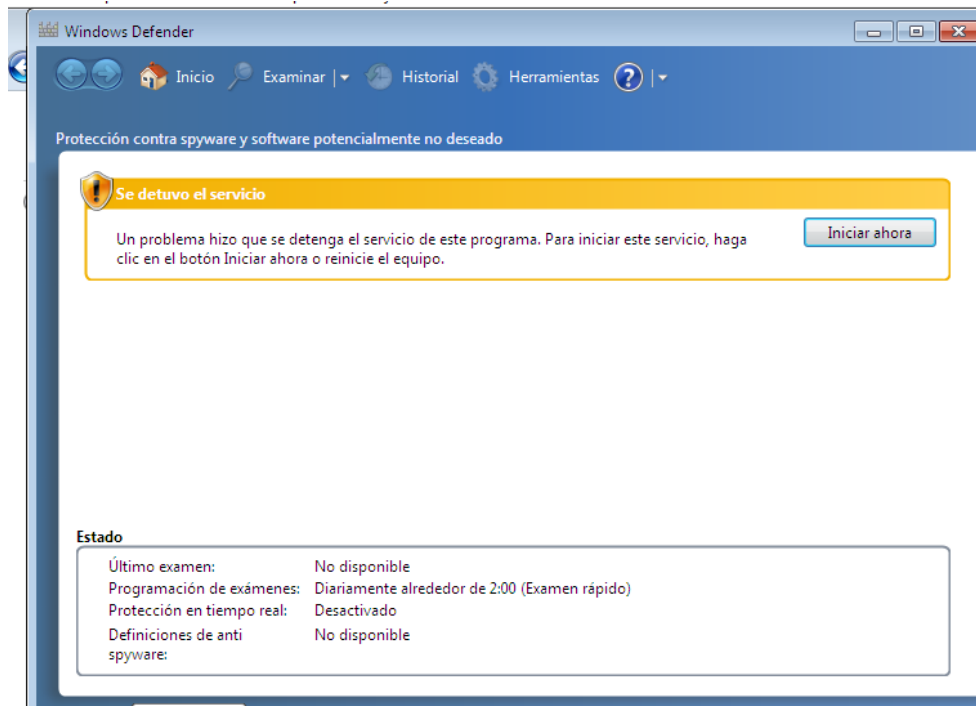
```

A continuación, vamos a silenciar la máquina con Windows

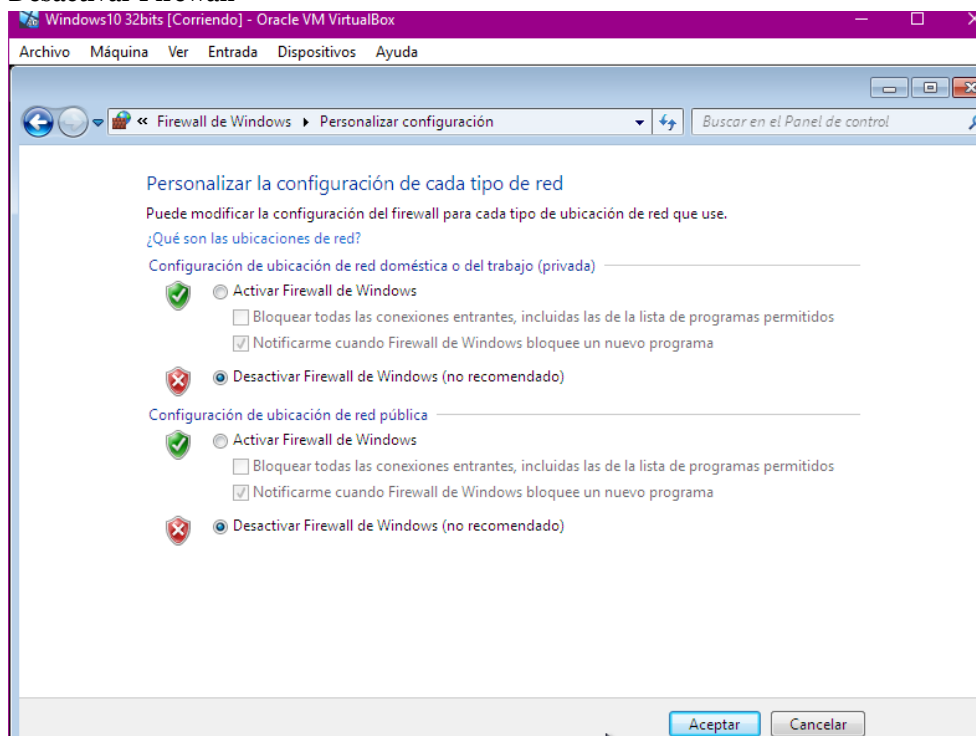
- Desactivamos el UAC.



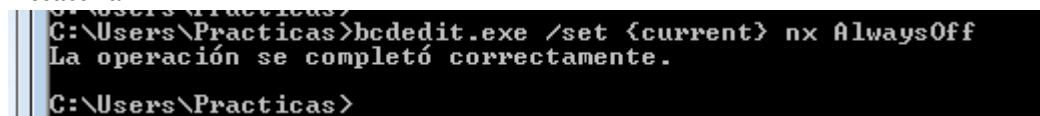
- Desactivar Windows Defender



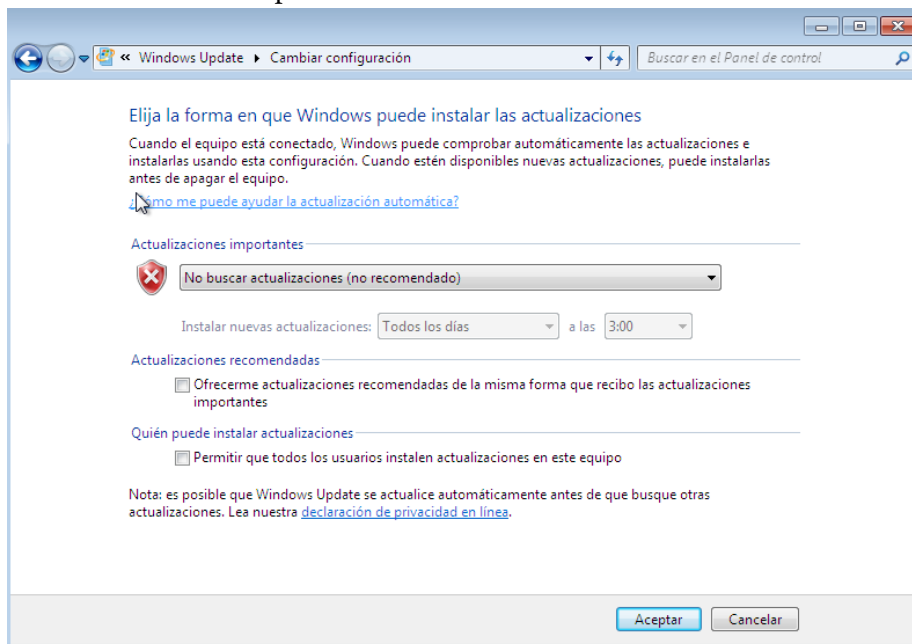
- Desactivar Firewall



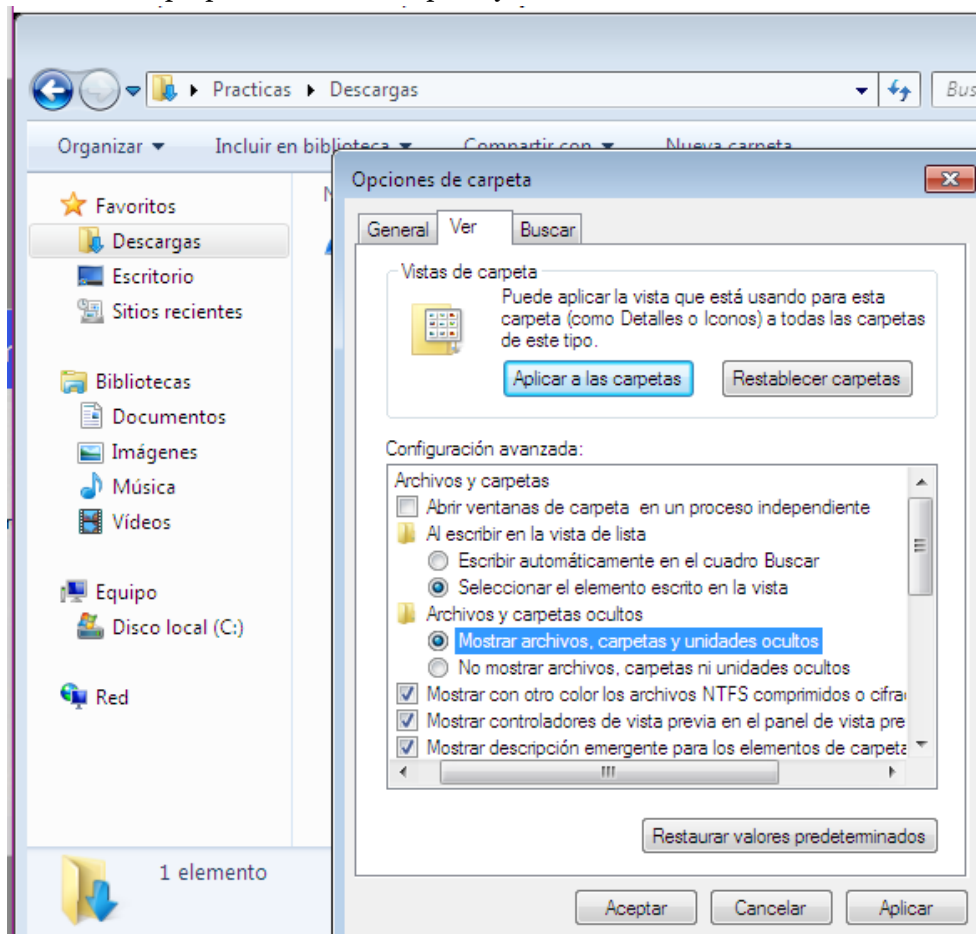
- Desactivar DEP



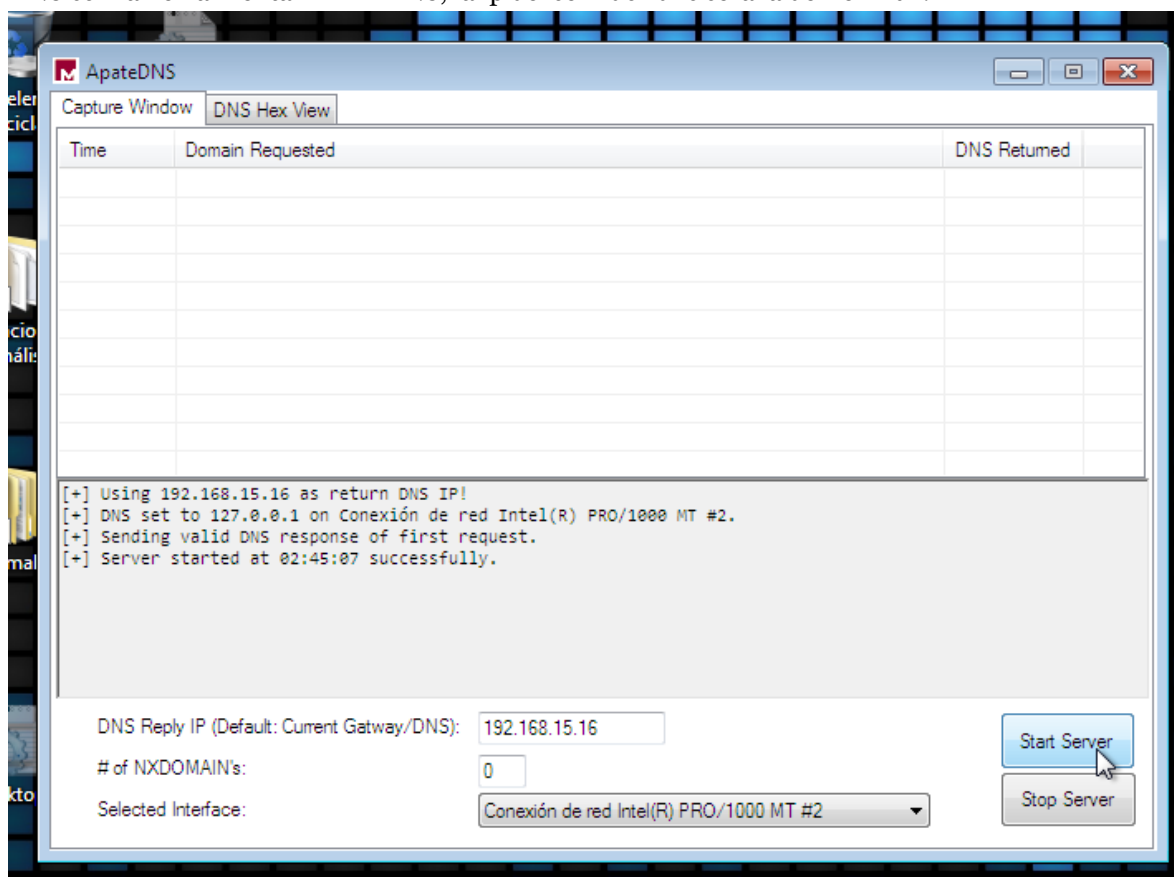
- Desactivar Windows Update



- Editamos las propiedades de las carpetas y marcamos mostrar los ficheros ocultos de sistema



- Simulación de servicios: Ahora mismo para la configuración inicial vamos a simular el servicio DNS con la herramienta APATEDNS, la ip del servidor dns será la de Remnux.



Inicialmente dejaremos en NAT las interfaces de red de ambos equipos para poder actualizar. Después de actualizar e instalar las herramientas necesarias para el análisis de malware cambiaremos los adaptadores de red para formar la red como se indicó en los pasos anteriores.

Nos vamos a basar en la plataforma Windows para analizar los posibles malware y los cambios que realizan en el equipo etc. Hemos elegido este sistema porque Windows es un sistema muy común en todo el mundo y es sobre el que más se ha trabajado para crear programas y archivos maliciosos.

Los archivos que vamos a analizar siguen el formato PE, por lo que conociendo la estructura de este formato cuando analicemos los ficheros podremos saber si es ejecutable o es una dll, las string que contiene, si está o no comprimido, podremos comparar con otros y saber si pertenecen a la misma familia...

El software que voy a necesitar será, instalar inetsim en la máquina Remnux, porque es la que nos proveerá los servicios necesarios para satisfacer las necesidades del malware en ejecución en la máquina Windows., además como permite registrar datos y capturar tráfico nos va ideal para poder realizar el análisis y ver paso a paso que está haciendo el malware. Necesitamos también herramientas como Process Explorer, Process Monitor, CFF Explorer, PE Studio, Vmmap. Estas herramientas nos ayudarán a realizar el análisis estático y dinámico del malware. En ellos podremos ver por ejemplo si un fichero es un ejecutable o una DLL, podremos ver los cambios realizados en memoria, podremos ver un registro de todos los eventos que se producen como creación de ficheros, cambios en el registro de Windows, creación de hebras... También podemos usar la herramienta strings en Remnux para ver

las cadenas que contiene el ejecutable que estamos analizando, o pscanner para ver la entropía y tratar de determinar si el archivo está comprimido o no.