

Routers Mikrotik hexS, RouterOS y protocolo RIPv1

Práctica 6
Gonzalo Tudela Chavero

ÍNDICE DE CONTENIDOS

EJERCICIO 1.....	1
Instalacion de routerOS.	1
Conexión a RouterOS.....	2
Winbox.	2
Password.....	3
Comprobación password.....	4
EJERCICIO 2	5
Esquema inicial:	6
Configuraciones previas y Powershell.....	6
¿Solución?	8
Enrutamiento manual en el host:	8
Pruebas de funcionamiento:.....	9
Router hexS: 192.168.2.1	10
VM con RouterOS: [192.168.2.3] - [192.168.4.1] - [192.168.3.1]	10
VM con Kali Linux: [192.168.3.2]	12
Esquema final:	13
EJERCICIO 3	13
Configuración RIPv1.	14
Fotografía del laboratorio de pruebas.	15
Esquema de la Red.....	15
Pruebas de funcionamiento.....	15
El ping extendido:	21

ÍNDICE DE FIGURAS

Figura 1. Página de mi cuenta Mikrotik donde se puede solicitar una licencia Demo que no expira.	1
Figura 2. Ventana de conexión de Winbox.	2
Figura 3. Ventana de gestión de licencias, Winbox.	2
Figura 4. Wireshark, captura comunicaciones Winbox.	3
Figura 5. Modificación de la contraseña por defecto del router.	3
Figura 6. Comprobación del cambio de contraseña.	4
Figura 7. Elementos utilizados para el ejercicio 2.	5
Figura 8. Esquema de red inicial.	6
Figura 9. Configuración y renombrado de los adaptadores vía Powershell.	7
Figura 10. Los adaptadores en modo Bridge no aparecen en la lista de interfaces.	7
Figura 11. Resultado del enrutamiento manual.	8
Figura 12. Comprobaciones Host.	9
Figura 13. Pings de router real al resto de redes.	10
Figura 14. Traceroute de RouterOS a Router real.	10
Figura 15. Traceroute de RouterOS a Ethernet Host.	11
Figura 16. Traceroute de RouterOS a VM Kali Linux	11
Figura 17. Ping de RouterOS a Microsoft Loopback	12
Figura 18. Pings de la VM Kali Linux al resto de redes.	12
Figura 19. Planteamiento final del ejercicio.	13
Figura 20. Configuración de RIPv1 mediante Winbox.	14
Figura 21. Configuración laboratorio Ejercicio 3.	15
Figura 22. Esquema de la red.	15
Figura 23. El router real Mikrotik hexS alcanza mediante ping al resto de elementos de la red, RIPv1.	16
Figura 24. Host alcanza todos los destinos RIPv1.	17
Figura 25. Mikrotik hexS alcanza el resto de las redes, RIPv1.	18
Figura 26. VM Kali Linux ping exitoso, RIPv1.	18
Figura 27. Wireshark escuchando la conversación RIPv1 entre los routers.	19
Figura 28. Listas de rutas del router virtual, RIPv1.	20
Figura 29. Listas de rutas del router real, RIPv1.	21
Figura 30. Ping extendido desde Mikrotik hexS a RouterOS.	22
Figura 31. Ping extendido desde RouterOS a Mikrotik hexS.	23

EJERCICIO 1

Con la herramienta “winbox” descargable desde <https://mikrotik.com/download>, conéctate con tu router Mikrotik. Tanto la máquina donde se ejecuta “winbox” como alguna de las interfaces del router tienen que estar en el mismo segmento de red, ya que la comunicación será a nivel 2/enlace (dirección MAC).

Contesta y documenta las siguientes preguntas:

- Utilizando el analizador de tráfico de red, indica qué mensajes se envían desde la máquina donde se ejecuta “winbox” y qué mensajes de respuesta envía el router. Indica qué protocolo utilizan relacionado con el descubrimiento de vecinos Mikrotik.
- Indica cuáles son los datos de acceso por defecto al router Mikrotik para su gestión y modifícalos por otros más seguros. Realiza una nueva conexión para ver si la modificación de los datos de acceso se ha tenido en cuenta.

Instalacion de routerOS.

Tras la instalación de la máquina virtual con RouterOS, al finalizar el primer arranque, el router nos informa de que tiene una licencia que expirará en 24h y deshabilitará ciertos servicios, por lo que procedemos a registrar una cuenta en la web de Mikrotik y solicitamos una licencia Demo.

Para ello se ha de suministrar el número de identificación de software (Software ID) que nos aporta el router al arrancar.

The screenshot shows the Mikrotik 'My account' page. The main heading is 'Make a demo key'. A green success message states: 'A new demo key was successfully generated!'. Below this, the 'Software ID' is 8458-8H99 and the 'Licence key' is a long alphanumeric string. The page also lists the features of the 'Free Demo (Trial) License Key for RouterOS 2.9 and up', including: demo (trial) license key is level 1 key, has limits of maximum connections each for PPTP, PPPoE, Queues, NAT, EoIP, and DHCP, does not have wireless interface support, does not include version upgrades, does not expire (no time limit), does not include support, and not for resale. At the bottom, there is a form to place the key in a folder (set to 'Demo keys') and a checkbox to send the key to the user's email.

Figura 1. Página de mi cuenta Microtik donde se puede solicitar una licencia Demo que no expira.

Conexión a RouterOS.

Procedemos a conectarnos al router, hemos de fijarnos en que las 2 interfaces de red que solicita el ejercicio aparecen como posibles direcciones (MAC) a través de las cuales podemos conectar y administrar el router.

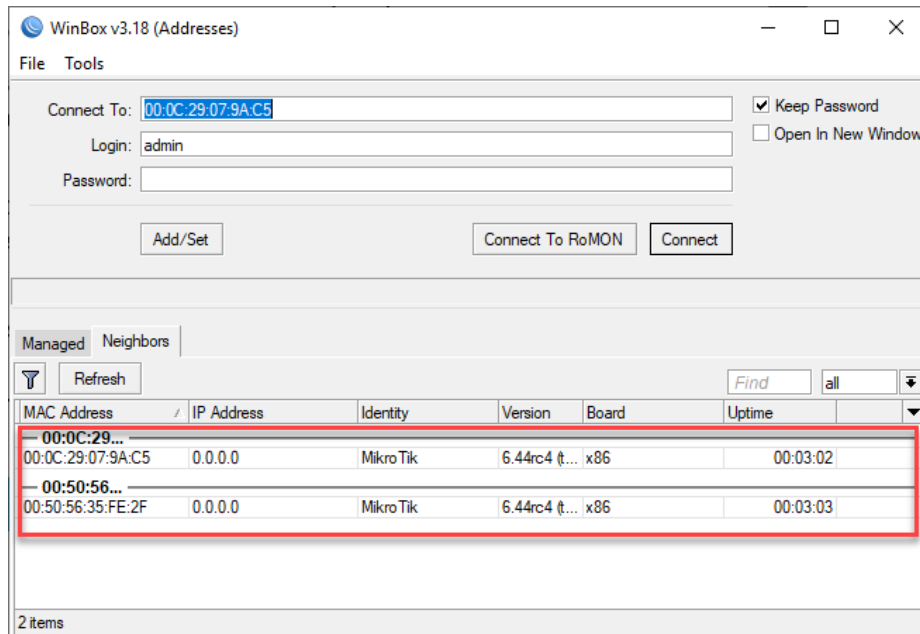


Figura 2. Ventana de conexión de Winbox.

Tras conectar, mediante la herramienta *license* del apartado *system* aplicamos la licencia haciendo *click* en el botón *import key*, necesitaremos que previamente hayamos copiado la licencia en el portapapeles.

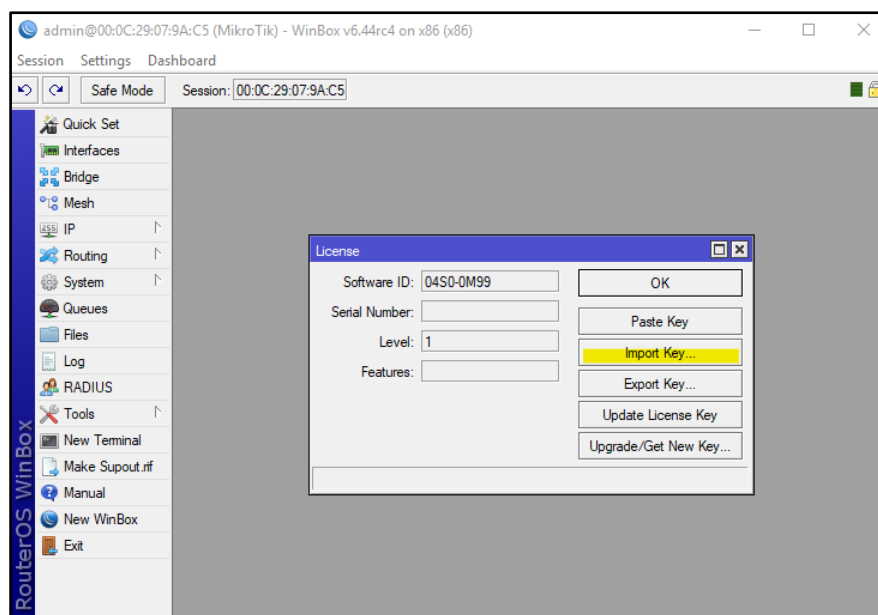


Figura 3. Ventana de gestión de licencias, Winbox.

a. Respuesta:

Winbox.

Cuando Winbox se conecta con el router se capturan paquetes de descubrimiento de vecinos por parte del router y también se captura la comunicación vía MAC-Telnet de Winbox con el router, ya que al conectar desde Winbox

hemos seleccionado la MAC como dirección (Figura 2) en vez de la IP, utilizar esta última nos hubiera llevado a capturar paquetes TCP para la comunicación con el router.

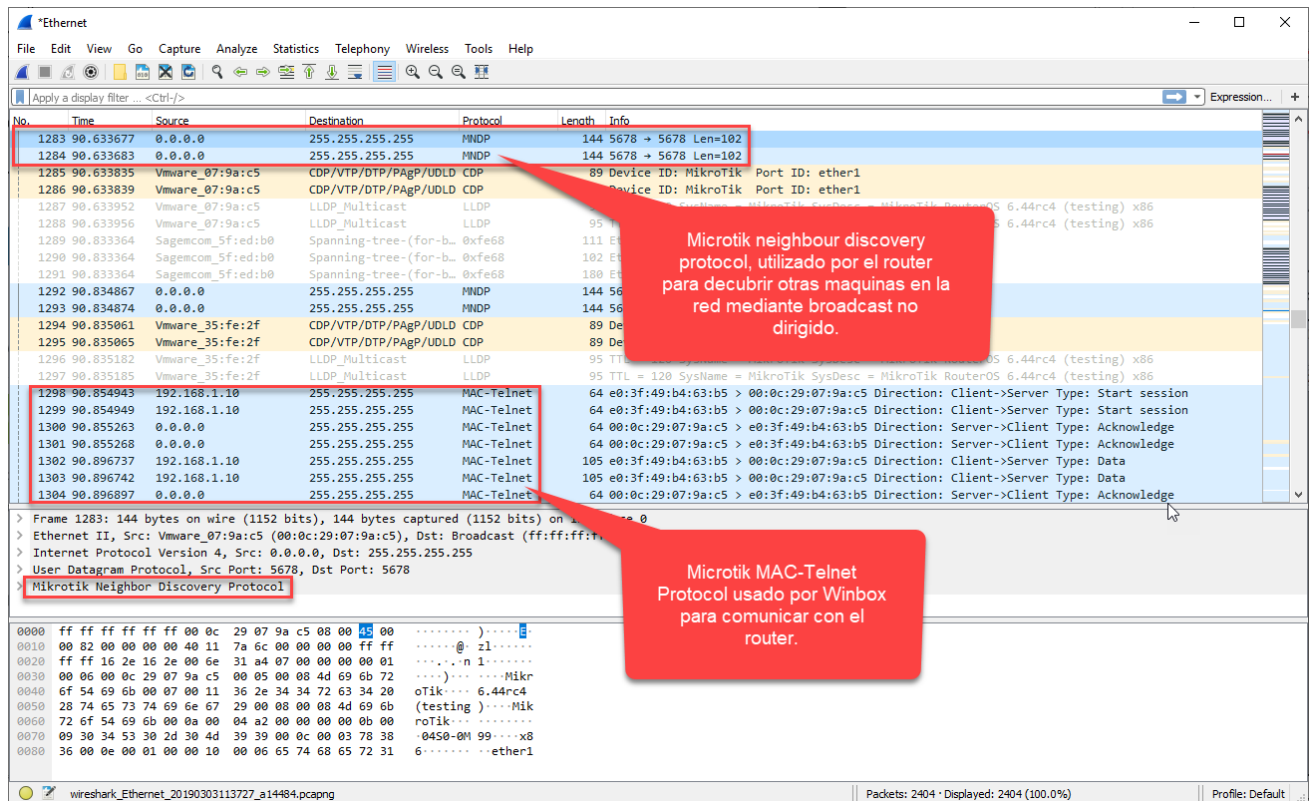


Figura 4. Wireshark, captura comunicaciones Winbox.

b. Respuesta:

Password.

Para acceder al router deberemos utilizar el usuario admin, para la contraseña simplemente no introduciremos ninguna.

Modificamos la contraseña del usuario actual (admin) utilizando el comando password.

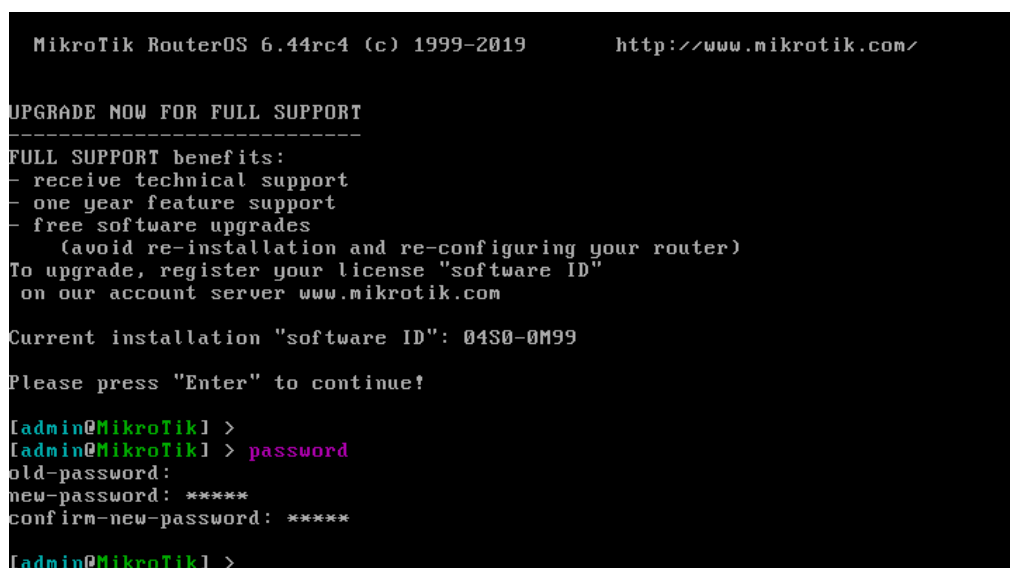


Figura 5. Modificación de la contraseña por defecto del router.

Ahora comprobamos que la contraseña aplicada funciona, reiniciaremos el router con el comando:

```
system reboot
```

Comprobación password.

Tras el reinicio comprobamos que el usuario *admin* ya no accede sin contraseña y utiliza la que hemos suministrado, todos los intentos fallidos serán informados una vez realicemos un *login* correcto como se puede apreciar en la siguiente imagen mi intento de acceder sin contraseña fue registrado y se muestra tras acceder con la contraseña correcta.

```

MM MM MMM III KKKKK RRR RRR 000 000 TTT III KKKKK
MMM MMM III KKK KKK RRRRRR 000 000 TTT III KKK KKK

UPGRADE NOW FOR FULL SUPPORT
-----
FULL SUPPORT benefits:
- receive technical support
- one year feature support
- free software upgrades
  (avoid re-installation and re-configuring your router)
To upgrade, register your license "software ID"
on our account server www.mikrotik.com

Current installation "software ID": 04S0-0M99

Please press "Enter" to continue!
Mar/03/2019 00:00:04 system,error,critical login failure for user admin via local
[admin@MikroTik] > _

```

Figura 6. Comprobación del cambio de contraseña.

EJERCICIO 2

Configura las dos interfaces de red con una configuración lógica (a nivel de red). Una de ellas se comunicará con el router Asus. Para comprobar la conectividad entre el host con el programa "winbox" y con el router Asus, se realizará un "ping extendido". Justifica y documenta los pasos y resultados obtenidos.

Para la realización de este ejercicio se han utilizado los componentes que aparecen en la siguiente imagen, router Mikrotik hexS (https://mikrotik.com/product/hex_s) y un portátil Lenovo T480.

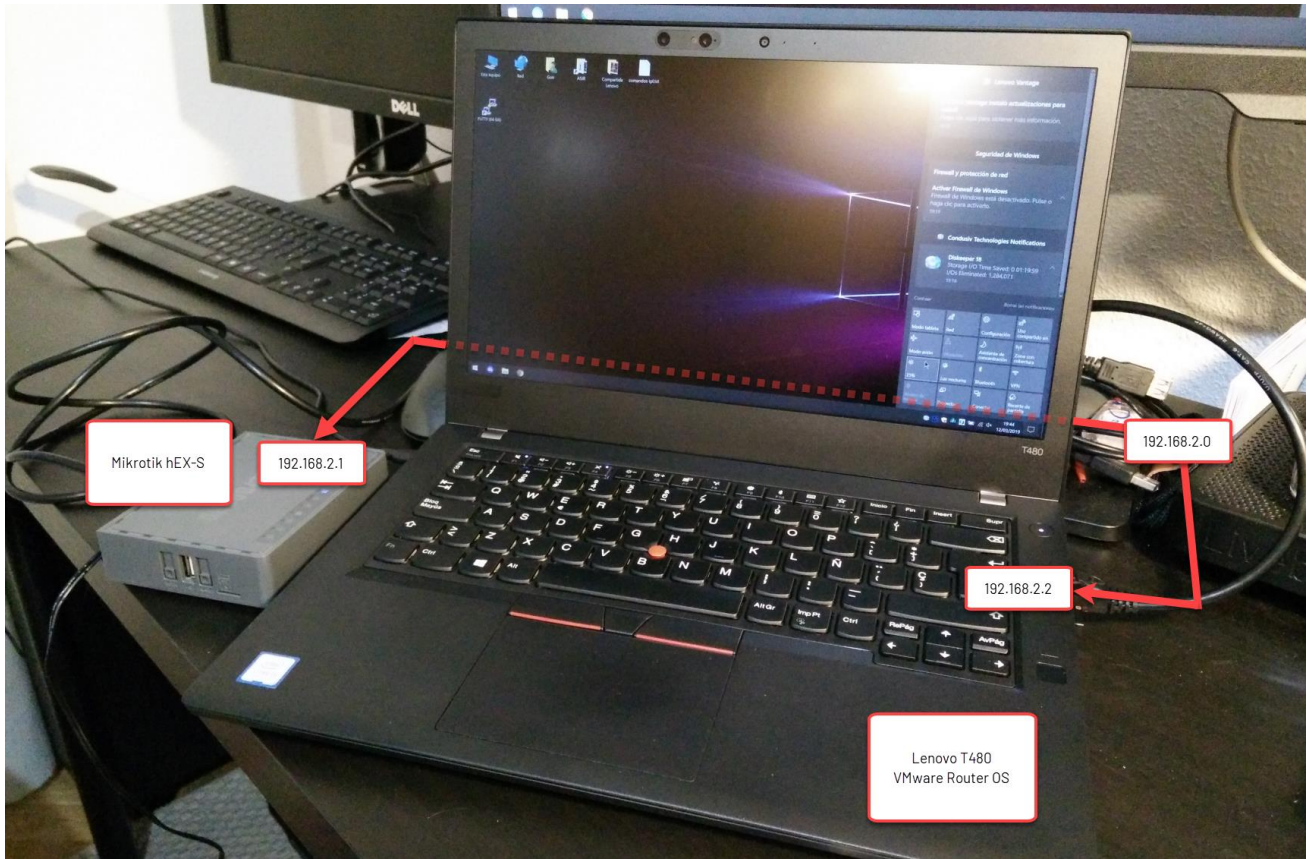


Figura 7. Elementos utilizados para el ejercicio 2.

En la Figura 8 se puede ver el esquema de red realizado para este ejercicio, la excesiva complejidad viene dada por el proceso de prueba, error y aprendizaje, así como la interpretación personal que he hecho de lo que solicitaba el ejercicio, ya que otros compañeros de curso han realizado otros esquemas mas coherentes comparados con este.

Para entender el porque de este esquema hemos de pensar en que se ha tomado como punto de partida el ejercicio anterior en el que se solicitaba la creación de una maquina virtual con Mikrotik RouterOS configurada con 2 adaptadores de red virtuales y conectarnos a esta mediante Winbox, en este punto la diferencia con otros compañeros ha sido que esta conexión mediante Winbox no se ha llevado a cabo desde una máquina virtual (a la que no se hace mención en el enunciado), sino desde el host real, lo que ha llevado a un planteamiento mucho más complejo y problemático, ya que el puenteo de Ether 1 y Ether 2 de VMware automático no funcionaba como pensaba.

Esto nos lleva al siguiente problema; el adaptador de red físico del portátil ha de estar en 2 redes, o, mejor dicho, conectado a 2 redes (la solución), la 192.168.2.0 y la 192.168.4.0 ya que los Ether 1 y 2 son los que inicialmente solicitaba el ejercicio.

Dicho de otra forma, Ether 1 y Kali estaban en una red, así como Ether 2 y el host en otra red.

¿Cómo con 1 solo adaptador de red físico por el que viajan los datos entre el router físico y el host hacemos que se dirija el tráfico a la red virtual de VMware?

¿Por qué con el modo bridge de VMware no he sido capaz de unir (BRIDGE) Ether 2 con el adaptador físico del host?

Todas estas preguntas van quedando aclaradas a medida que se realiza el montaje y configuración que se observa en la figura 7 así como un planteamiento más sencillo.

Esquema inicial:

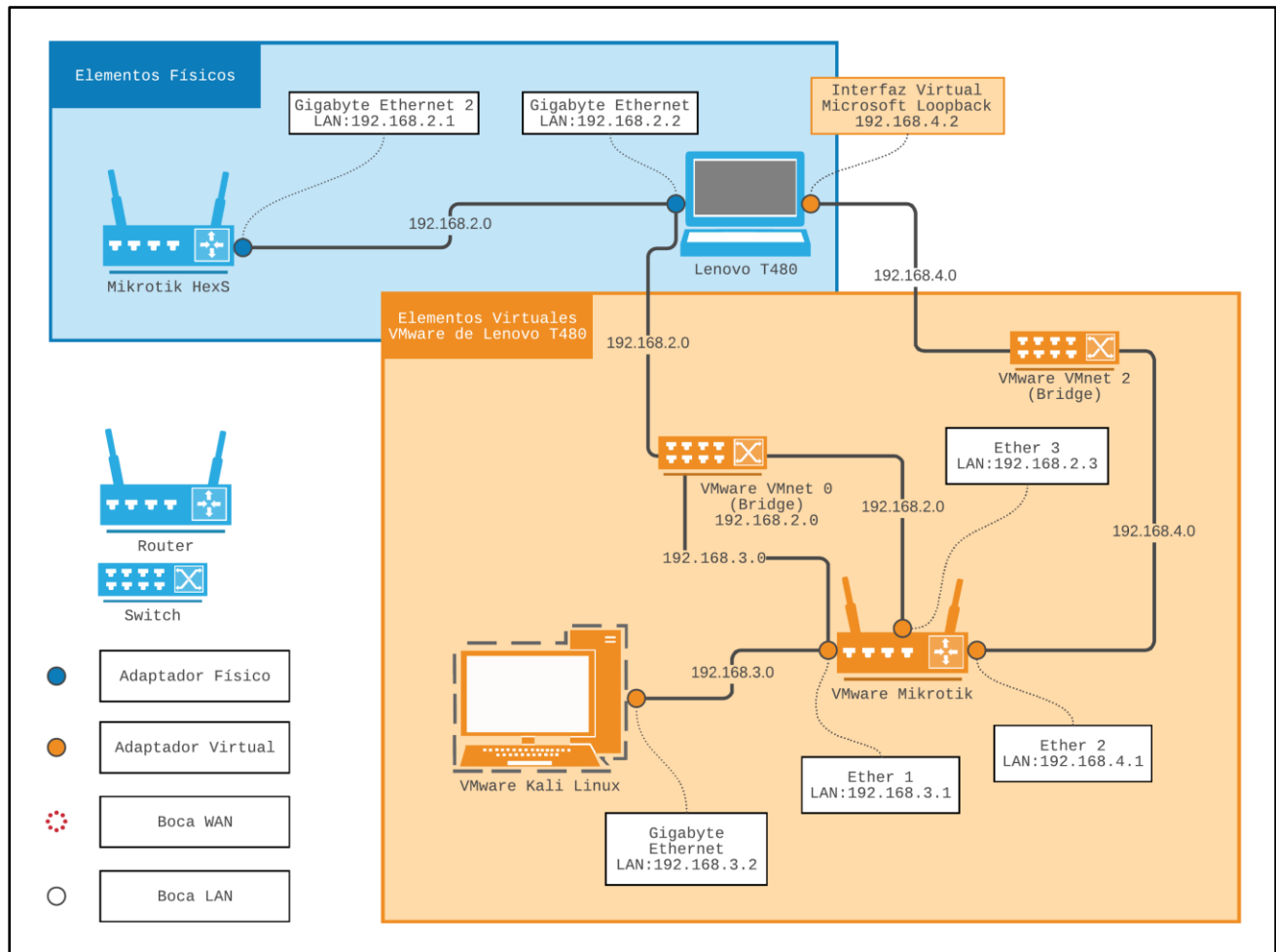


Figura 8. Esquema de red inicial.

Configuraciones previas y Powershell.

Antes de empezar para poder tener más claro que adaptador es quien en la ventana de configuración de Windows he utilizado los siguientes comandos de Powershell para renombrar los adaptadores de red, también se utilizó *powershell* para eliminar configuraciones problemáticas de puertas de enlace en otras redes que ocasionan la aparición de 2 puertas de enlace en la presentación que hace *ipconfig*.

```
Get-netadapter
```

```
Rename-netadapter ethernet eth0
```

```
Remove-netipaddress -interfacealias eth0
```

```
New-netipaddress -intefacealias eth0 -addressfamily ipv4 -ipaddress 192.68.2.2 -  
prefixlenght 24 -defaultgateway 192.168.2.1
```

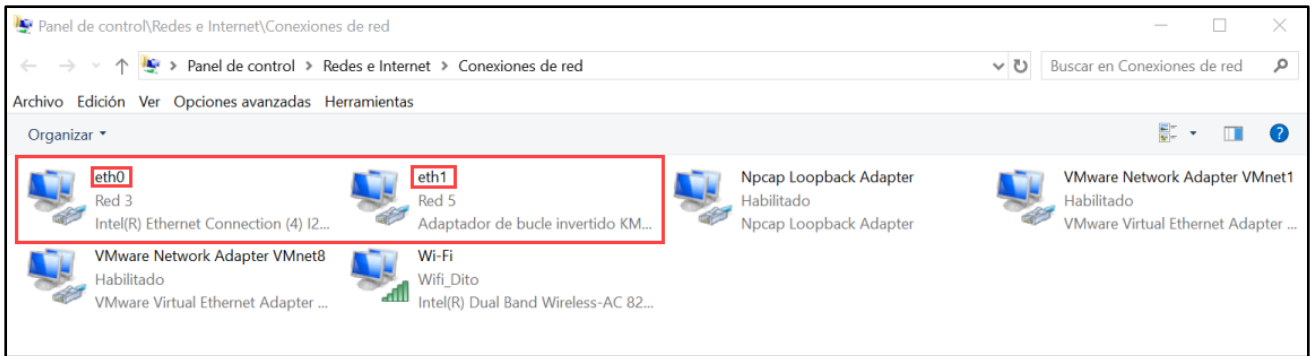


Figura 9. Configuración y renombrado de los adaptadores vía Powershell.

Problemas:

- a. ¿Por qué con el modo bridge de VMware no he sido capaz de unir (BRIDGE) Ether 2 con el adaptador físico del host?

Puentear los 2 Ether de RouterOS al adaptador físico del host fue en vano, ya que no se pueden añadir rutas a adaptadores de red que no aparecen en la lista de interfaces (los adaptadores en modo BRIDGE).

¡Sin embargo VMnet1 (host only) y VMnet8 (NAT) sí aparecen!

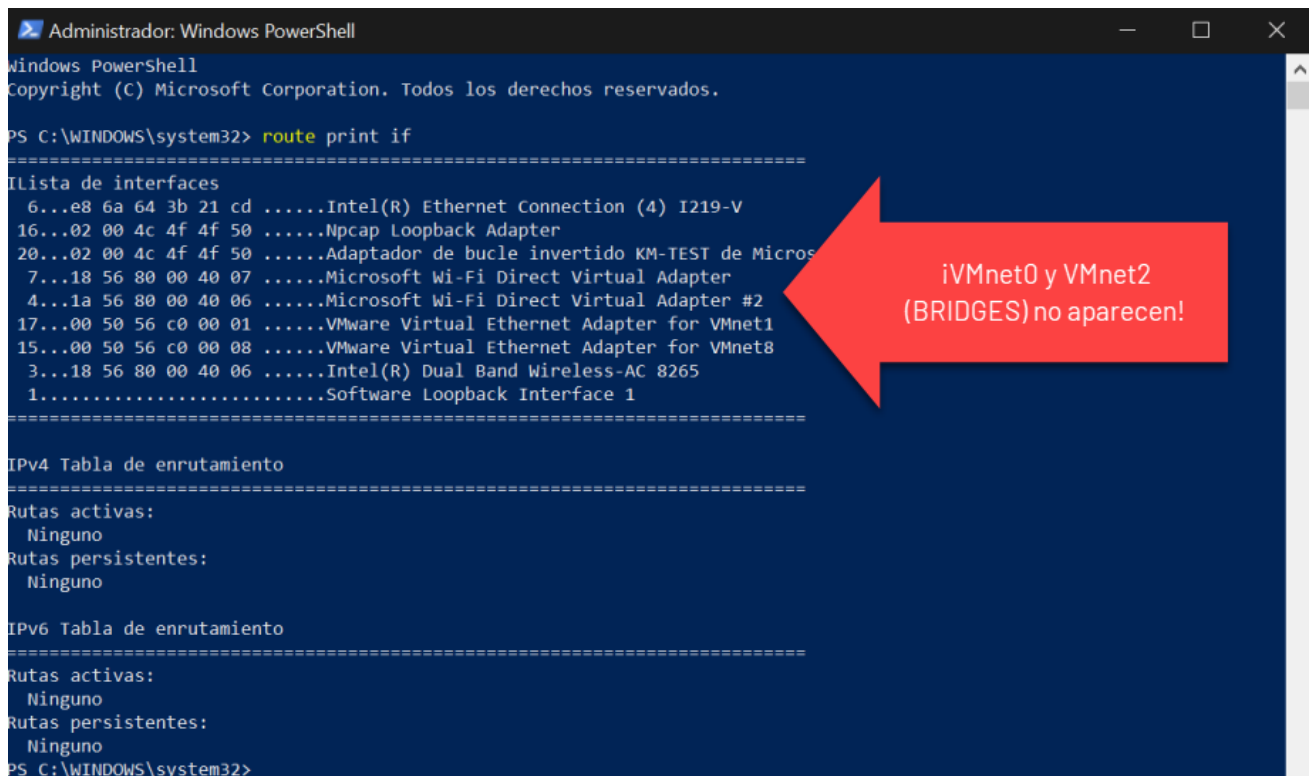


Figura 10. Los adaptadores en modo Bridge no aparecen en la lista de interfaces.

¿Solución?

Crear el adaptador de bucle invertido de Microsoft para que este sí aparezca y realizar un nuevo bridge a este, lo que implica crear un Ether 3 en RouterOS.

- b. ¿Cómo con 1 solo adaptador de red físico por el que viajan los datos entre el router físico y el host hacemos que se dirija el tráfico a la red virtual de VMware?

El problema parte del planteamiento erróneo o malentendido por mi parte, otras soluciones vistas en clase al compartir impresiones con mi compañero David Lagheza han entendido mejor aportando una solución más coherente.

¿Solución?

La solución a mi planteamiento pasa por introducir un router más en el esquema que como no puede ser de otra forma unirá las 2 redes (192.168.2.0) y (192.168.4.0) que responden a la red física del router físico y el host, y por otro lado el tramo de red que conecta este nuevo router (virtual) con el ya existente router virtual RouterOS.

La solución realizada en clase fue introducir un nuevo adaptador Ether 3 en RouterOS y asignarle un nuevo puente VMnet2, esta vez enrutado en el host con el adaptador que, si aparece en la lista de route, Microsoft Loopback (al que se le asigno la ip 192.168.4.2).

Enrutamiento manual en el host:

```
route add 192.168.2.0 mask 255.255.255.0 192.168.2.1 if 6
```

```
route add 192.168.2.0 mask 255.255.255.0 192.168.4.1 if 20
```



Figura 11. Resultado del enrutamiento manual.

Pruebas de funcionamiento:

Las pruebas constan de pruebas de comunicación mediante comandos ping y traceroute.

Host: 192.168.2.2 y 192.168.4.2

```

Administrador: Windows PowerShell

PS C:\WINDOWS\system32> ping 192.168.2.1

Haciendo ping a 192.168.2.1 con 32 bytes de datos:
Respuesta desde 192.168.2.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.1: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.2.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping 192.168.2.3

Haciendo ping a 192.168.2.3 con 32 bytes de datos:
Respuesta desde 192.168.2.3: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.3: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.3: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.2.3: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.2.3:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping 192.168.4.2

Haciendo ping a 192.168.4.2 con 32 bytes de datos:
Respuesta desde 192.168.4.2: bytes=32 tiempo<1m TTL=128
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping 192.168.4.1

Haciendo ping a 192.168.4.1 con 32 bytes de datos:
Respuesta desde 192.168.4.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.4.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.4.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.4.1: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.4.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping 192.168.3.1

Haciendo ping a 192.168.3.1 con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 192.168.3.1:
    Paquetes: enviados = 1, recibidos = 0, perdidos = 1
            (100% perdidos),
Control-c
PS C:\WINDOWS\system32> ping 192.168.3.2

Haciendo ping a 192.168.3.2 con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 192.168.3.2:
    Paquetes: enviados = 1, recibidos = 0, perdidos = 1
            (100% perdidos),
Control-c
PS C:\WINDOWS\system32> route add 192.168.3.0 mask 255.255.255.0 192.168.2.3 if 6
Correcto
PS C:\WINDOWS\system32> ping 192.168.3.2

Haciendo ping a 192.168.3.2 con 32 bytes de datos:
Respuesta desde 192.168.3.2: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.3.2: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.3.2: bytes=32 tiempo<1m TTL=63
Respuesta desde 192.168.3.2: bytes=32 tiempo<1m TTL=63

Estadísticas de ping para 192.168.3.2:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping 192.168.3.1

Haciendo ping a 192.168.3.1 con 32 bytes de datos:
Respuesta desde 192.168.3.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.3.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.3.1: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.3.1: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.3.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32>
  
```

Habia problemas con este ping, no había ruta para esa red!

```

Administrador: Windows PowerShell

Windows PowerShell
Copyright (c) Microsoft Corporation. Todos los derechos reservados.

PS C:\WINDOWS\system32> route print if

=====
Lista de interfaces
=====
6...e8 6a 64 3b 21 cd .....Intel(R) Ethernet Connec
16...02 00 4c 4f 4f 50 .....Npcap Loopback Adapter
20...02 00 4c 4f 4f 50 .....Adaptador de bucle inve
7...18 56 80 00 40 07 .....Microsoft Wi-Fi Direct
4...1a 56 80 00 40 06 .....Microsoft Wi-Fi Direct
17...00 50 56 c0 00 01 .....VMware Virtual Ethernet
15...00 50 56 c0 00 08 .....VMware Virtual Ethernet
3...18 56 80 00 40 06 .....Intel(R) Dual Band Wire
1.....Software Loopback Inter
=====
  
```

Figura 12. Comprobaciones Host.

Router hexS: 192.168.2.1

Serie de pings realizados, no se ha conseguido alcanzar la red 192.168.4.0 a pesar de que el host y la VM tienen rutas estáticas para esa red.

```

[admin@MikroTik_hEXs] > ping 192.168.2.2
  SEQ HOST                       SIZE TTL TIME  STATUS
  0 192.168.2.2                   56 128 0ms
  1 192.168.2.2                   56 128 1ms
  sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=1ms

[admin@MikroTik_hEXs] > ping 192.168.3.1
  SEQ HOST                       SIZE TTL TIME  STATUS
  0 192.168.3.1                   56 64 0ms
  1 192.168.3.1                   56 64 0ms
  sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTik_hEXs] > ping 192.168.3.2
  SEQ HOST                       SIZE TTL TIME  STATUS
  0 192.168.3.2                   56 63 1ms
  1 192.168.3.2                   56 63 1ms
  sent=2 received=2 packet-loss=0% min-rtt=1ms avg-rtt=1ms max-rtt=1ms

[admin@MikroTik_hEXs] > ping 192.168.4.2
  SEQ HOST                       SIZE TTL TIME  STATUS
  0 192.168.4.2                   56 63 1ms
  sent=1 received=0 packet-loss=100%

[admin@MikroTik_hEXs] > ping 192.168.4.2
  SEQ HOST                       SIZE TTL TIME  STATUS
  0 192.168.4.2                   56 63 1ms
  sent=1 received=0 packet-loss=100%

[admin@MikroTik_hEXs] >
  
```

Figura 13. Pings de router real al resto de redes.

VM con RouterOS: [192.168.2.3] - [192.168.4.1] - [192.168.3.1]

RouterOS considera que siempre hay 1 salto porque se cuenta a si mismo.

Traceroute To: 192.168.2.1

Packet Size: 56

Timeout: 1000 ms

Protocol: icmp

Port: 33434

☐ Use DNS

Count: [dropdown]

Max Hops: [dropdown]

Src. Address: [dropdown]

Interface: [dropdown]

DSCP: [dropdown]

Routing Table: [dropdown]

Hop	Host	Loss	Sent	Last	Avg.	Best	Worst	Std. Dev.	History	Status
1	192.168.2.1	0.0%	16	0.5ms	0.5	0.4	0.6	0.1	[bar chart]	[status icon]

Figura 14. Traceroute de RouterOS a Router real.

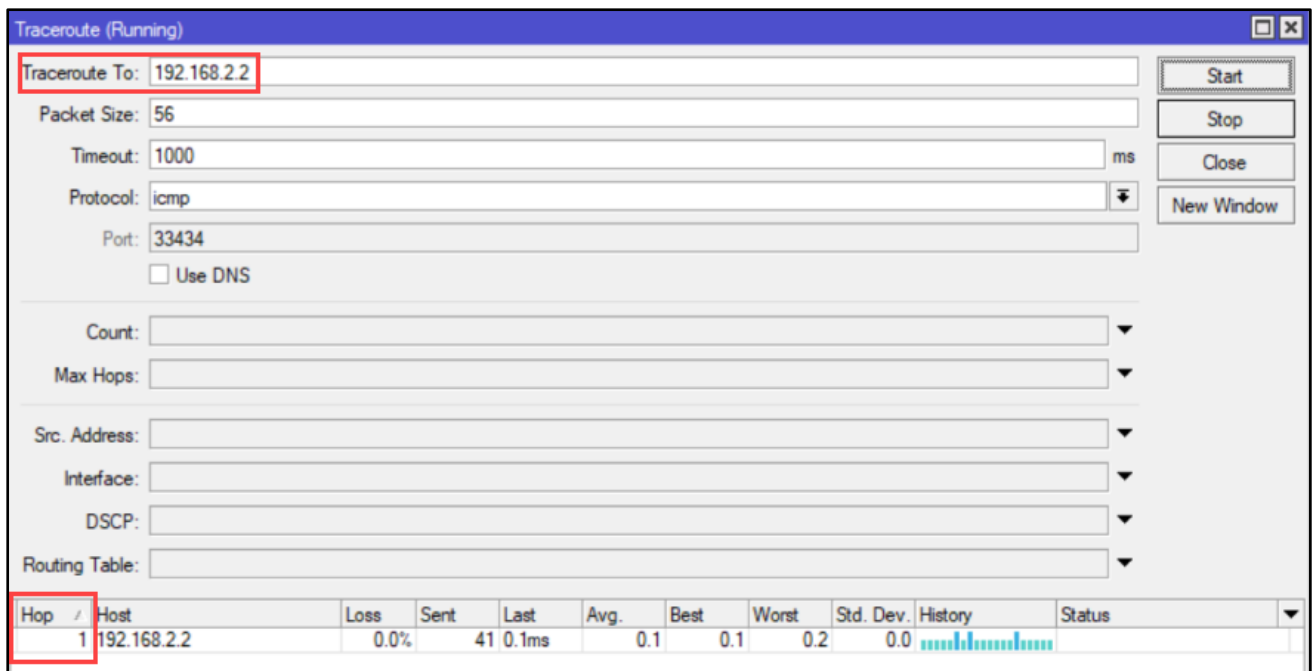


Figura 15. Traceroute de RouterOS a Ethernet Host.

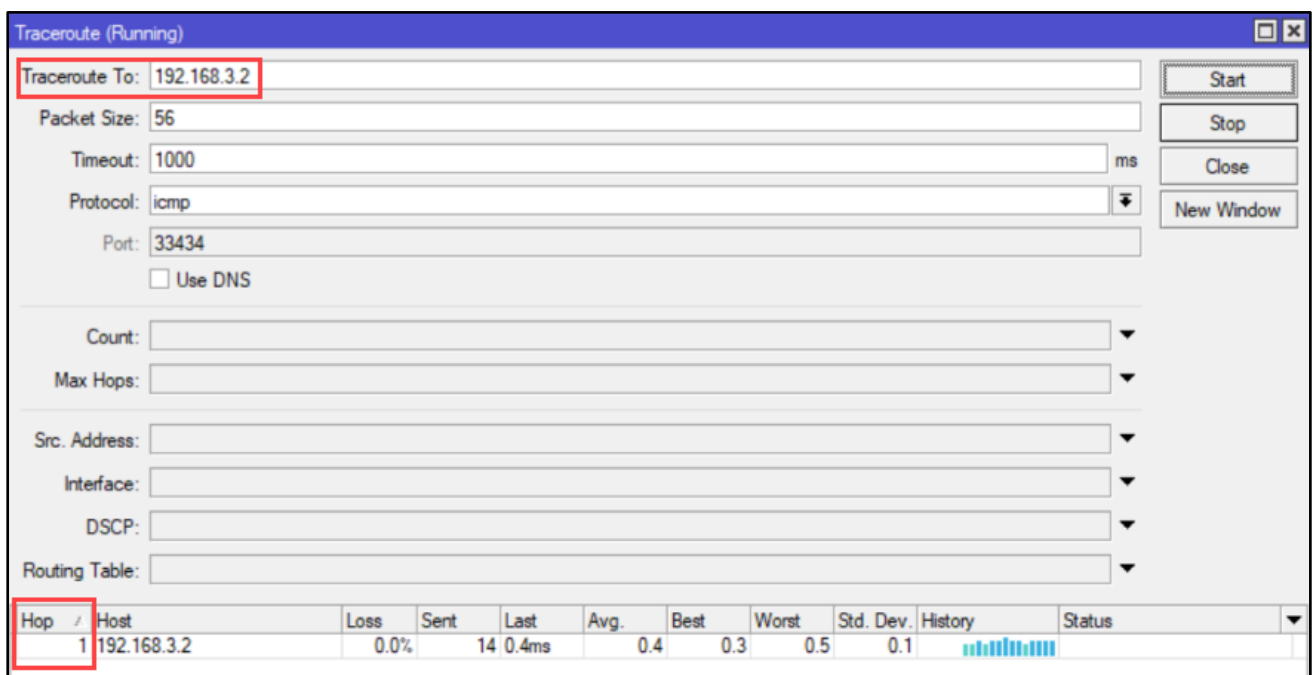


Figura 16. Traceroute de RouterOS a VM Kali Linux

El resto menos interesante he realizado ping.

```

Terminal
MikroTik RouterOS 6.44rc4 (c) 1999-2019      http://www.mikrotik.com/

UPGRADE NOW FOR FULL SUPPORT
-----
FULL SUPPORT benefits:
- receive technical support
- one year feature support
- free software upgrades
  (avoid re-installation and re-configuring your router)
To upgrade, register your license "software ID"
on our account server www.mikrotik.com

Current installation "software ID": 04S0-0M99

Please press "Enter" to continue!

[admin@MikroTik] > ping 192.168.4.2
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.4.2                          56 128 0ms
    1 192.168.4.2                          56 128 0ms
  sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms
[admin@MikroTik] >

```

Figura 17. Ping de RouterOS a Microsoft Loopback

El resto de los pings funcionan correctamente puesto son bocas conectadas directamente al router.

VM con Kali Linux: [192.168.3.2]

```

1/1 + [ ] [ ] Tilix: Default
root@kali:~# ip route
default via 192.168.3.1 dev eth0 onlink
192.168.3.0/24 dev eth0 proto kernel scope link src 192.168.3.2
192.168.4.0/24 via 192.168.3.1 dev eth0
root@kali:~# ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data:
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=1.02 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=1.57 ms
^C
--- 192.168.2.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt_min/avg/max/mdev = 1.018/1.296/1.574/0.278 ms
root@kali:~# ping 192.168.2.2
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data:
64 bytes from 192.168.2.2: icmp_seq=1 ttl=127 time=0.662 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=127 time=0.653 ms
^C
--- 192.168.2.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 5ms
rtt_min/avg/max/mdev = 0.653/0.657/0.662/0.026 ms
root@kali:~# ping 192.168.3.1
PING 192.168.3.1 (192.168.3.1) 56(84) bytes of data:
64 bytes from 192.168.3.1: icmp_seq=1 ttl=64 time=0.377 ms
64 bytes from 192.168.3.1: icmp_seq=2 ttl=64 time=0.532 ms
^C
--- 192.168.3.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 28ms
rtt_min/avg/max/mdev = 0.377/0.454/0.532/0.080 ms
root@kali:~# ping 192.168.4.2
PING 192.168.4.2 (192.168.4.2) 56(84) bytes of data:
64 bytes from 192.168.4.2: icmp_seq=1 ttl=127 time=0.679 ms
64 bytes from 192.168.4.2: icmp_seq=2 ttl=127 time=0.677 ms
^C
--- 192.168.4.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 21ms
rtt_min/avg/max/mdev = 0.677/0.678/0.679/0.001 ms
root@kali:~# ping 192.168.4.1
PING 192.168.4.1 (192.168.4.1) 56(84) bytes of data:
64 bytes from 192.168.4.1: icmp_seq=1 ttl=64 time=0.453 ms
64 bytes from 192.168.4.1: icmp_seq=2 ttl=64 time=0.595 ms
^C
--- 192.168.4.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 20ms
rtt_min/avg/max/mdev = 0.453/0.524/0.595/0.071 ms
root@kali:~#

```

Figura 18. Pings de la VM Kali Linux al resto de redes.

Esquema final:

Como se ha dicho mas arriba este hubiera sido el planteamiento a seguir con menos problemas, aunque el planteamiento anterior ha sido enriquecedor desde el punto de vista didáctico, posiblemente más que este último, aunque hay que decir que en el inicial no era posible hacer pruebas con protocolos de enrutamiento dinámicos.

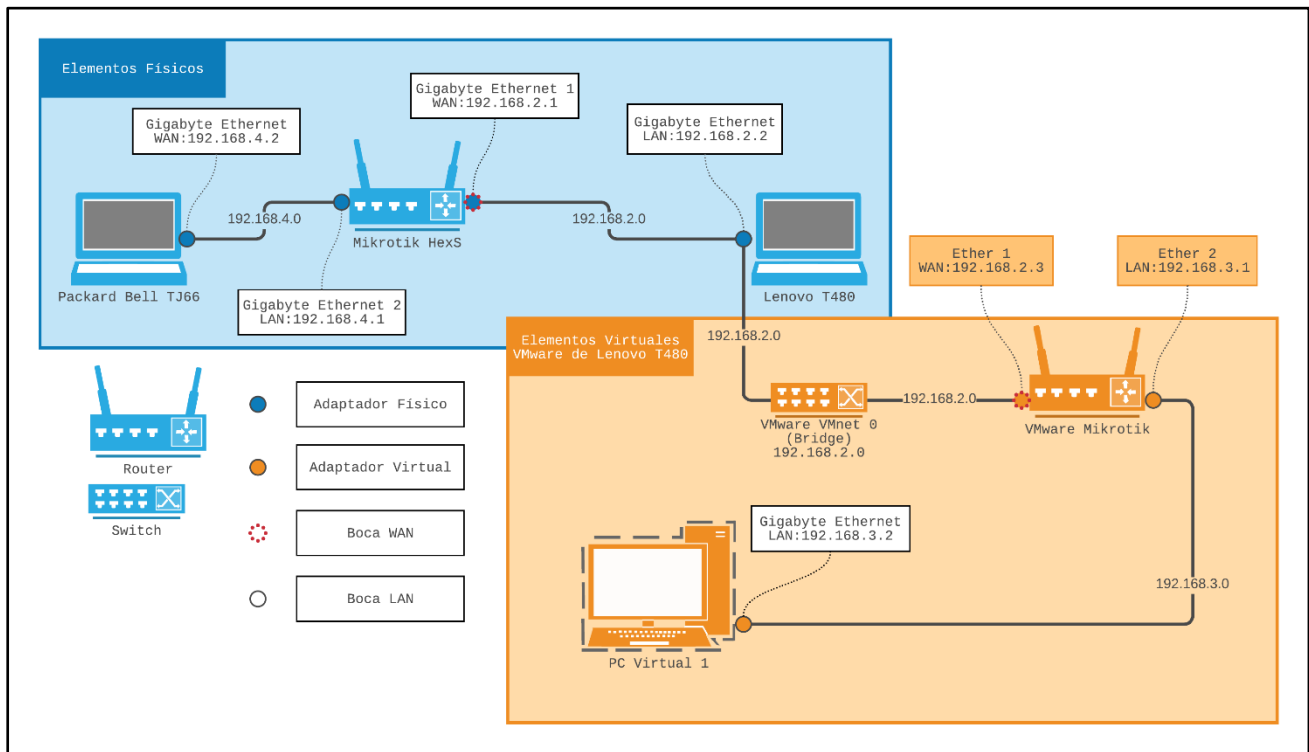


Figura 19. Planteamiento final del ejercicio.

EJERCICIO 3

Realiza el punto anterior pero esta vez utilizando RIPv1. ¿Puedes obtener un mensaje de RIPv1 cuando los routers se intercambian sus tablas de rutas? Justifica la respuesta.

Para la realización de este ejercicio hemos replanteado todo el esquema de la red ya que en el anterior al estar conectado mediante bocas pertenecientes a la parte LAN de los dispositivos no deberían recibir paquetes RIPv1. Aunque como hemos visto en anteriores ejercicios y practicas con Packet Tracer esto se ha de hacer en cisco mediante un comando para evitar la inundación de las partes LAN con paquetes RIP.

Configuración RIPv1.

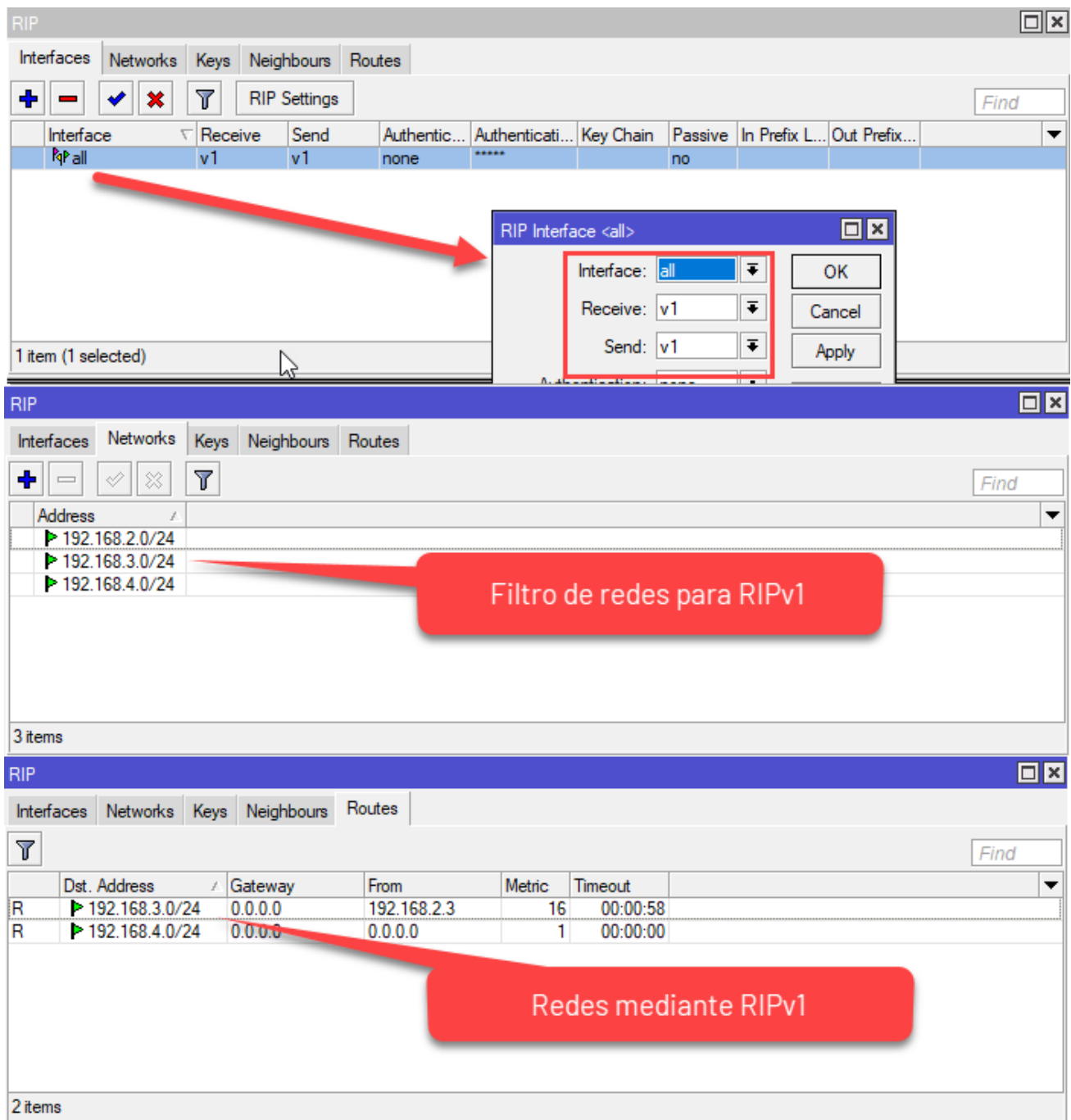


Figura 20. Configuración de RIPv1 mediante Winbox.

Fotografía del laboratorio de pruebas.

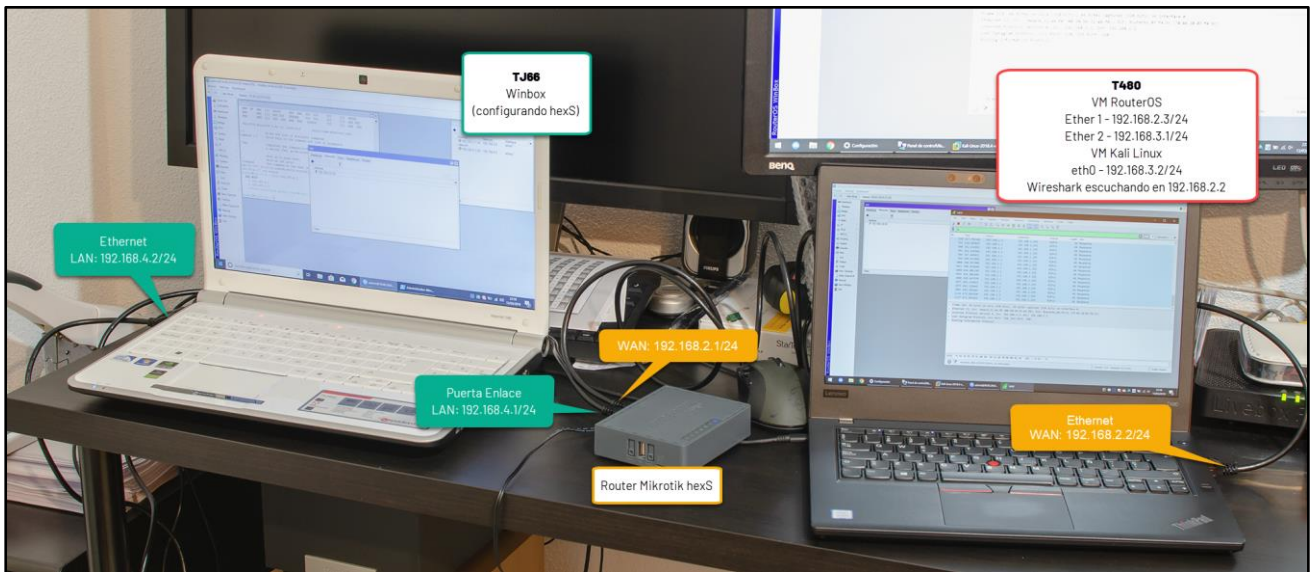


Figura 21. Configuración laboratorio Ejercicio 3.

Esquema de la Red.

Hemos de fijarnos en que para que la prueba sea correcta el adaptador de red de la maquina virtual Kali esta configurado como host only para que no haya dudas acerca de que el tráfico pueda estar puentado con la red 192.168.2.0/24

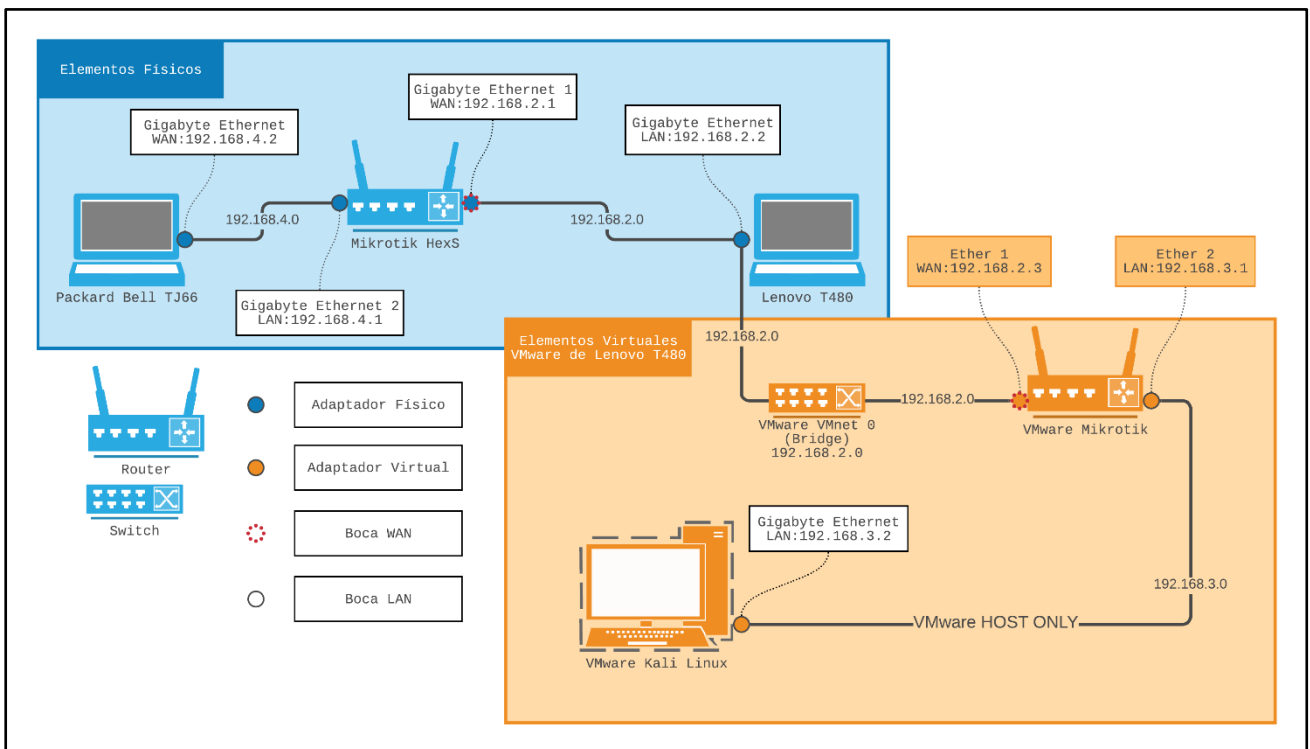
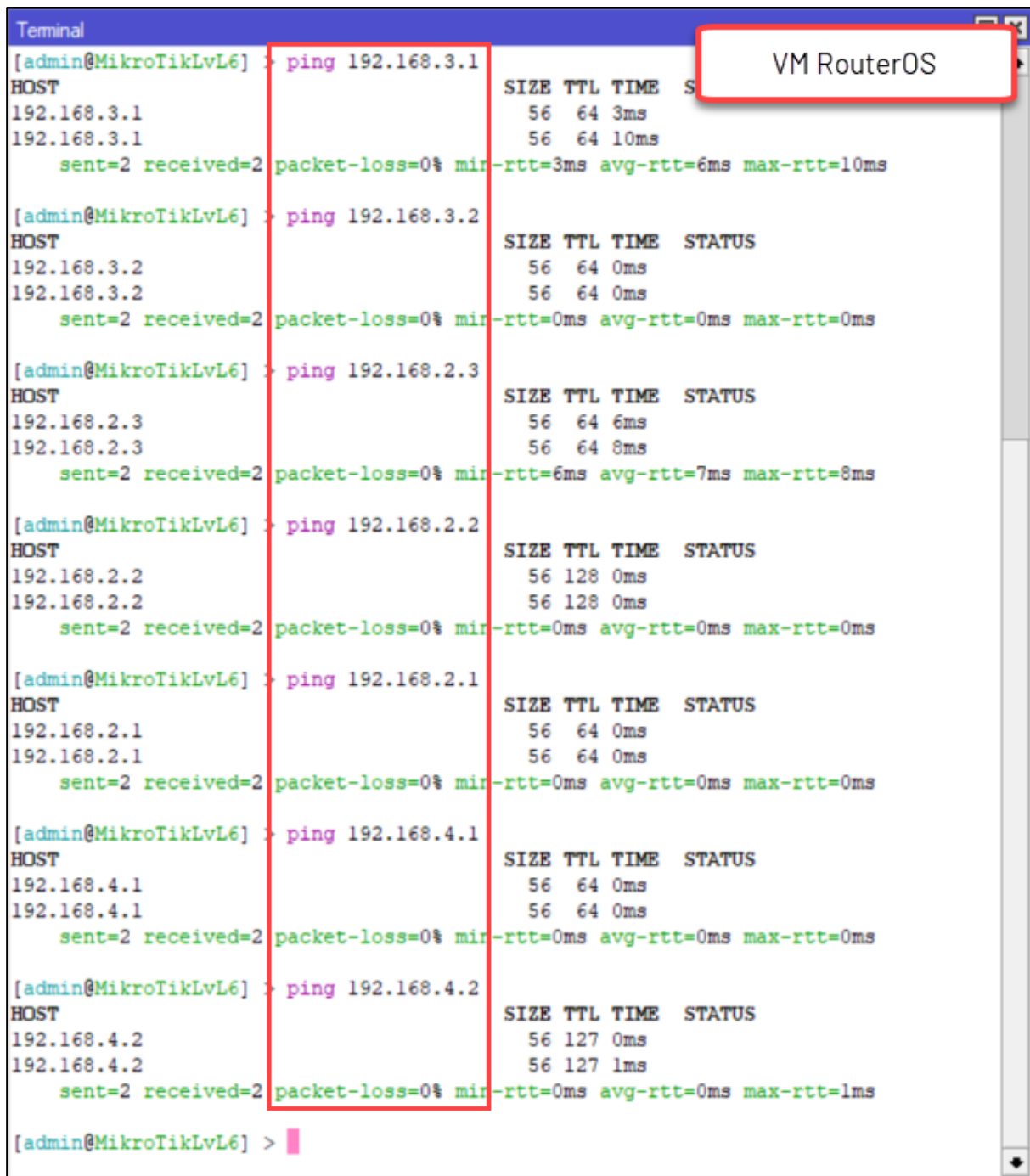


Figura 22. Esquema de la red.

Pruebas de funcionamiento.

La comprobación de este ejercicio primero realizaremos una revisión de que las comunicaciones están funcionando entre cada uno de los adaptadores de las redes, así como la captura de paquetes RIPv1 en el Wireshark que se ejecuta en el host y que está capturando todo lo que pasa por el puerto ethernet.

El router virtual Mikrotik alcanza mediante ping al resto de elementos de la red.



```

Terminal
[admin@MikroTikLvL6] > ping 192.168.3.1
HOST
192.168.3.1      SIZE TTL TIME  S
192.168.3.1      56  64 3ms
192.168.3.1      56  64 10ms
    sent=2 received=2 packet-loss=0% min-rtt=3ms avg-rtt=6ms max-rtt=10ms

[admin@MikroTikLvL6] > ping 192.168.3.2
HOST
192.168.3.2      SIZE TTL TIME  STATUS
192.168.3.2      56  64 0ms
192.168.3.2      56  64 0ms
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTikLvL6] > ping 192.168.2.3
HOST
192.168.2.3      SIZE TTL TIME  STATUS
192.168.2.3      56  64 6ms
192.168.2.3      56  64 8ms
    sent=2 received=2 packet-loss=0% min-rtt=6ms avg-rtt=7ms max-rtt=8ms

[admin@MikroTikLvL6] > ping 192.168.2.2
HOST
192.168.2.2      SIZE TTL TIME  STATUS
192.168.2.2      56 128 0ms
192.168.2.2      56 128 0ms
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTikLvL6] > ping 192.168.2.1
HOST
192.168.2.1      SIZE TTL TIME  STATUS
192.168.2.1      56  64 0ms
192.168.2.1      56  64 0ms
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

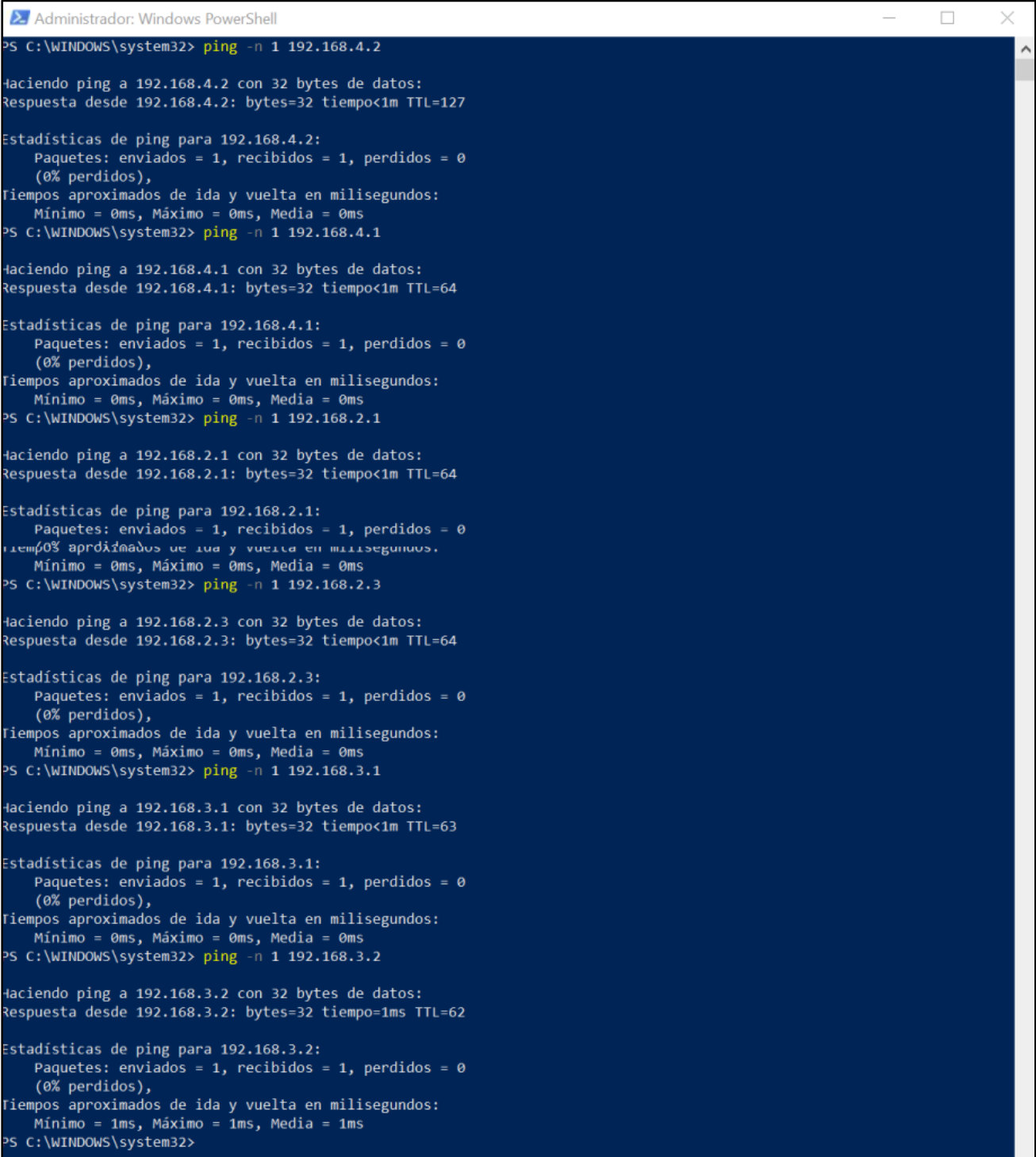
[admin@MikroTikLvL6] > ping 192.168.4.1
HOST
192.168.4.1      SIZE TTL TIME  STATUS
192.168.4.1      56  64 0ms
192.168.4.1      56  64 0ms
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTikLvL6] > ping 192.168.4.2
HOST
192.168.4.2      SIZE TTL TIME  STATUS
192.168.4.2      56 127 0ms
192.168.4.2      56 127 1ms
    sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=1ms

[admin@MikroTikLvL6] >
  
```

Figura 23. El router real Mikrotik hexS alcanza mediante ping al resto de elementos de la red, RIPv1.

El host realizará ping a los routers que tiene conectados excepto a su propia interfaz de red, todas los destinos fueron alcanzados mediante ping.



```

Administrador: Windows PowerShell

PS C:\WINDOWS\system32> ping -n 1 192.168.4.2

Haciendo ping a 192.168.4.2 con 32 bytes de datos:
Respuesta desde 192.168.4.2: bytes=32 tiempo<1m TTL=127

Estadísticas de ping para 192.168.4.2:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping -n 1 192.168.4.1

Haciendo ping a 192.168.4.1 con 32 bytes de datos:
Respuesta desde 192.168.4.1: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.4.1:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping -n 1 192.168.2.1

Haciendo ping a 192.168.2.1 con 32 bytes de datos:
Respuesta desde 192.168.2.1: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.2.1:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping -n 1 192.168.2.3

Haciendo ping a 192.168.2.3 con 32 bytes de datos:
Respuesta desde 192.168.2.3: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 192.168.2.3:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping -n 1 192.168.3.1

Haciendo ping a 192.168.3.1 con 32 bytes de datos:
Respuesta desde 192.168.3.1: bytes=32 tiempo<1m TTL=63

Estadísticas de ping para 192.168.3.1:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
PS C:\WINDOWS\system32> ping -n 1 192.168.3.2

Haciendo ping a 192.168.3.2 con 32 bytes de datos:
Respuesta desde 192.168.3.2: bytes=32 tiempo=1ms TTL=62

Estadísticas de ping para 192.168.3.2:
    Paquetes: enviados = 1, recibidos = 1, perdidos = 0
        (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 1ms, Máximo = 1ms, Media = 1ms
PS C:\WINDOWS\system32>
  
```

Figura 24. Host alcanza todos los destinos RIPv1.

El router real Mikrotik hexS alcanza sin problemas todos las redes mediante ping.

```

[admin@MikroTik] > ping 192.168.4.2
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.4.2                          56 128 0ms
  sent=1 received=1 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTik] > ping 192.168.2.2
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.2.2                          56 128 0ms
  sent=1 received=1 packet-loss=0% min-rtt=0ms avg-rtt=0ms max-rtt=0ms

[admin@MikroTik] > ping 192.168.2.3
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.2.3                          56  64 1ms
  sent=1 received=1 packet-loss=0% min-rtt=1ms avg-rtt=1ms max-rtt=1ms

[admin@MikroTik] > ping 192.168.3.1
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.3.1                          56  64 1ms
  sent=1 received=1 packet-loss=0% min-rtt=1ms avg-rtt=1ms max-rtt=1ms

[admin@MikroTik] > ping 192.168.3.2
  SEQ HOST                                SIZE TTL TIME  STATUS
    0 192.168.3.2                          56  63 1ms
  sent=1 received=1 packet-loss=0% min-rtt=1ms avg-rtt=1ms max-rtt=1ms

[admin@MikroTik] >
  
```

Figura 25. Mikrotik hexS alcanza el resto de las redes, RIPv1.

La VM Kali Linux alcanza sin problemas el resto de los componentes del esquema.

```

root@kali:~# ping -c 1 192.168.3.1
PING 192.168.3.1 (192.168.3.1) 56(84) bytes of data:
64 bytes from 192.168.3.1: icmp_seq=1 ttl=64 time=0.954 ms

--- 192.168.3.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.954/0.954/0.954/0.000 ms
root@kali:~# ping -c 1 192.168.2.3
PING 192.168.2.3 (192.168.2.3) 56(84) bytes of data:
64 bytes from 192.168.2.3: icmp_seq=1 ttl=64 time=0.541 ms

--- 192.168.2.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.541/0.541/0.541/0.000 ms
root@kali:~# ping -c 1 192.168.2.2
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data:
64 bytes from 192.168.2.2: icmp_seq=1 ttl=126 time=1.57 ms

--- 192.168.2.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.569/1.569/1.569/0.000 ms
root@kali:~# ping -c 1 192.168.2.1
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data:
64 bytes from 192.168.2.1: icmp_seq=1 ttl=63 time=1.17 ms

--- 192.168.2.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.172/1.172/1.172/0.000 ms
root@kali:~# ping -c 1 192.168.4.1
PING 192.168.4.1 (192.168.4.1) 56(84) bytes of data:
64 bytes from 192.168.4.1: icmp_seq=1 ttl=63 time=1.33 ms

--- 192.168.4.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.328/1.328/1.328/0.000 ms
root@kali:~# ping -c 1 192.168.4.2
PING 192.168.4.2 (192.168.4.2) 56(84) bytes of data:
64 bytes from 192.168.4.2: icmp_seq=1 ttl=126 time=1.50 ms

--- 192.168.4.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.503/1.503/1.503/0.000 ms
root@kali:~#
  
```

```

root@kali:~# nano /etc/network/interfaces
root@kali:~# cat /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

# Configuración de eth0
auto eth0
iface eth0 inet static
    address 192.168.3.2
    netmask 255.255.255.0
    gateway 192.168.3.1

#auto eth1
#iface eth1 inet static
#    address 192.168.2.4
#    netmask 255.255.255.0
  
```

Virtual Machine Settings

Device	Summary	Device status
Memory	4 GB	<input checked="" type="checkbox"/> Connected
Processors	2	<input checked="" type="checkbox"/> Connect at power on
Hard Disk (SCSI)	80 GB	
CD/DVD (IDE)	Auto detect	
Network Adapter	Host-only	
Display	Auto detect	

Network connection

- ☐ Bridged: Connected directly to the physical network
- ☐ Replicate physical network connection state
- ☐ NAT: Used to share the host's IP address
- ☒ Host-only: A private network shared with the host

Figura 26. VM Kali Linux ping exitoso, RIPv1.

¡Wireshark muestra la intensa conversación RIPv1 entre los routers, mas de 237 paquetes intercambiados!

The screenshot shows a Wireshark capture on interface eth0. The packet list is filtered for 'rip'. The following table represents the data visible in the packet list:

No.	Time	Source	Destination	Protocol	Length	Info
140...	1785.856555	192.168.2.1	192.168.2.255	RIPv1	106	Response
141...	1811.926767	192.168.2.3	192.168.2.255	RIPv1	106	Response
141...	1811.926780	192.168.2.3	192.168.2.255	RIPv1	106	Response
142...	1817.229369	192.168.2.1	192.168.2.255	RIPv1	106	Response
143...	1843.190482	192.168.2.1	192.168.2.255	RIPv1	106	Response
143...	1845.565589	192.168.2.3	192.168.2.255	RIPv1	106	Response
143...	1845.565602	192.168.2.3	192.168.2.255	RIPv1	106	Response
145...	1871.764615	192.168.2.3	192.168.2.255	RIPv1	106	Response
145...	1871.764627	192.168.2.3	192.168.2.255	RIPv1	106	Response
145...	1872.156732	192.168.2.1	192.168.2.255	RIPv1	106	Response
147...	1897.576483	192.168.2.3	192.168.2.255	RIPv1	106	Response
147...	1897.576496	192.168.2.3	192.168.2.255	RIPv1	106	Response
147...	1898.818594	192.168.2.1	192.168.2.255	RIPv1	106	Response
148...	1922.687142	192.168.2.3	192.168.2.255	RIPv1	106	Response
148...	1922.687148	192.168.2.3	192.168.2.255	RIPv1	106	Response
149...	1932.063470	192.168.2.1	192.168.2.255	RIPv1	106	Response

The packet details pane for Frame 3 shows:

- Frame 3: 126 bytes on wire (1008 bits), 126 bytes captured (1008 bits) on interface 0
- Ethernet II, Src: Vmware_31:a4:f0 (00:50:56:31:a4:f0), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.2.255
- User Datagram Protocol, Src Port: 520, Dst Port: 520
- Routing Information Protocol

Annotations in the image:

- Router Virtual:** Points to the source IP 192.168.2.1 in several packets.
- Router Real:** Points to the source IP 192.168.2.3 in several packets.
- Broadcast dirigido!:** Points to the destination IP 192.168.2.255.

A summary box at the bottom right states: "Wireshark capturando el trafico RIPv1 entre los routers real y virtual."

The status bar at the bottom shows: "Packets: 15002 · Displayed: 203 (1.4%) Profile: Default"

Figura 27. Wireshark escuchando la conversación RIPv1 entre los routers.

Comprobación de las listas de rutas en el router virtual.

The image shows two windows from a network configuration tool. The top window, titled 'RIP', has tabs for 'Interfaces', 'Networks', 'Keys', 'Neighbours', and 'Routes'. The 'Routes' tab is selected, showing a table of three routes. A red callout points to this table with the text 'Rutas RIPv1'. The bottom window, titled 'Route List', has tabs for 'Routes', 'Nexthops', 'Rules', and 'VRF'. The 'Routes' tab is selected, showing a table of three routes. A red callout points to the third route with the text 'Ruta recibida en un paquete RIPv1 desde el router real.'.

RIP Routes Table:

	Dst. Address	Gateway	From	Metric	Timeout
R	192.168.2.0/24	0.0.0.0	0.0.0.0	1	00:00:00
R	192.168.3.0/24	0.0.0.0	0.0.0.0	1	00:00:00
R	192.168.4.0/24	0.0.0.0	192.168.2.1	2	00:02:41

3 items

Route List Table:

	Dst. Address	Gateway	Distance	Routing Mark	Pref. Source
DAC	192.168.2.0/24	ether1 reachable	0		192.168.2.3
DAC	192.168.3.0/24	ether2 reachable	0		192.168.3.1
DAr	192.168.4.0/24	192.168.2.1 reachable ether1	120		

3 items

Figura 28. Listas de rutas del router virtual, RIPv1.

Comprobación de las listas de rutas en el router virtual.

RIP

	Dst. Address	Gateway	From	Metric	Timeout
R	192.168.2.0/24	0.0.0.0	0.0.0.0	1	00:00:00
R	192.168.3.0/24	0.0.0.0	192.168.2.3	2	00:02:44
R	192.168.4.0/24	0.0.0.0	0.0.0.0	1	00:00:00

3 items

Route List

	Dst. Address	Gateway	Distance	Routing Mark	Pref. Source
AS	0.0.0.0/0	192.168.2.2 reachable ether1	1		
DAC	192.168.2.0/24	ether1 reachable	0		192.168.2.1
DAr	192.168.3.0/24	192.168.2.3 reachable ether1	120		
DAC	192.168.4.0/24	bridge reachable	0		192.168.4.1

4 items

Figura 29. Listas de rutas del router real, RIPv1.

El ping extendido:

El ping extendido es una versión especial del comando ping que poseen algunos routers, ya que dispone de entre otros parámetros uno que permite seleccionar la interfaz de origen desde la que se manda el ping para poder detectar problemas de enrutamiento en el router o de configuración de la interfaz, de otra manera el ping tomaría como origen la interfaz de salida que corresponda para el destino.

En este caso voy a mostrar capturas de pantalla correspondientes a los ping extendidos desde cada router con la salvedad de que solo realizaremos 2 pruebas:

Un ping que nace de la boca LAN del router y llega a la boca LAN del otro router, y viceversa.

Mikrotik hexS -> VM RouterOS

The screenshot displays the Mikrotik WinBox interface with the Ping tool open. The 'General' tab is selected, and the 'Ping To' field is set to 192.168.3.1. The 'Packet Count' is 1000 and the 'Timeout' is 1000 ms. The 'Start' button is highlighted. To the right, a red-bordered box contains the text 'Ping Extendido: hexS -> RouterOS'.

The 'Advanced' tab is also visible, showing the 'Src. Address' field set to 192.168.4.1. Other fields include 'Packet Size: 50', 'TTL: 64', 'DSCP: 0', and 'Routing Table: 0'. The 'Dont Fragment' checkbox is unchecked.

The results table shows 12 items, with 12 of 12 packets received, 0% packet loss, and a minimum time of 0 ms. The table data is as follows:

Seq #	Host	Time	Reply Size	TTL	Status
0	192.168.3.1	0ms	50	64	
1	192.168.3.1	1ms	50	64	
2	192.168.3.1	1ms	50	64	
3	192.168.3.1	1ms	50	64	
4	192.168.3.1	1ms	50	64	
5	192.168.3.1	1ms	50	64	
6	192.168.3.1	1ms	50	64	
7	192.168.3.1	0ms	50	64	
8	192.168.3.1	1ms	50	64	
9	192.168.3.1	1ms	50	64	
10	192.168.3.1	1ms	50	64	
11	192.168.3.1	0ms	50	64	

The summary bar at the bottom of the results table shows: 12 items, 12 of 12 packets received, 0% packet loss, Min: 0 ms, Avg: 0 ms, Max: 1 ms.

Figura 30. Ping extendido desde Mikrotik hexS a RouterOS.

VM RouterOS -> Mikrotik hexS.

Ping extendido
VM Router OS->Mikrotik hexS

Seq # /	Host	Time	Reply Size	TTL	Status
0	192.168.4.1	0ms	50	64	
1	192.168.4.1	0ms	50	64	
2	192.168.4.1	0ms	50	64	
3	192.168.4.1	0ms	50	64	
4	192.168.4.1	0ms	50	64	
5	192.168.4.1	0ms	50	64	
6	192.168.4.1	0ms	50	64	
7	192.168.4.1	0ms	50	64	
8	192.168.4.1	0ms	50	64	
9	192.168.4.1	0ms	50	64	
10	192.168.4.1	0ms	50	64	
11	192.168.4.1	0ms	50	64	

12 items | 12 of 12 packets re... | 0% packet loss | Min: 0 ms | Avg:

Seq # /	Host	Time	Reply Size	TTL	Status
0	192.168.4.1	0ms	50	64	
1	192.168.4.1	0ms	50	64	
2	192.168.4.1	0ms	50	64	
3	192.168.4.1	0ms	50	64	
4	192.168.4.1	0ms	50	64	
5	192.168.4.1	0ms	50	64	
6	192.168.4.1	0ms	50	64	
7	192.168.4.1	0ms	50	64	
8	192.168.4.1	0ms	50	64	
9	192.168.4.1	0ms	50	64	
10	192.168.4.1	0ms	50	64	
11	192.168.4.1	0ms	50	64	

12 items | 12 of 12 packets re... | 0% packet loss | Min: 0 ms | Avg: 0 ms | Max: 0 ms

Figura 31. Ping extendido desde RouterOS a Mikrotik hexS.