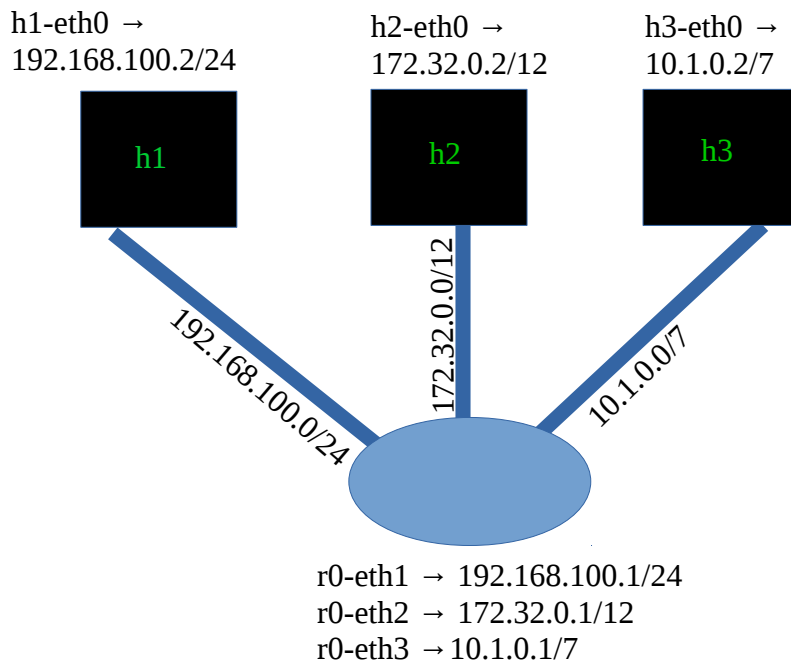


ESQUEMA ESCENARIO 2



COMANDOS DE CONFIGURACION

AÑADIR IP's

#ip a → Para comprobar las ip's de las interfaces de red.

```
"Node: h1" (on mininet-vm)
root@mininet-vm:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: h1-eth0@if12: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 3a:4c:81:c2:17:ef brd ff:ff:ff:ff:ff:ff
root@mininet-vm:~#
```

#ip a add → Para añadirles las ip's a las interfaces de red.

```
"Node: h1" (on mininet-vm)
root@mininet-vm:~# ip a add 192.168.0.2/24 dev h1-eth0
root@mininet-vm:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: h1-eth0@if12: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 3a:4c:81:c2:17:ef brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.2/24 scope global h1-eth0
        valid_lft forever preferred_lft forever
root@mininet-vm:~#
```

Una vez añadida la ip, el siguiente paso es añadirle una tabla de encaminamiento para que todos los host se interconecten.

AÑADIR RUTAS DE ENCAMINAMIENTO

#ip r add → Para añadirle ruta de encaminamiento a los hosts.

#ip r → Para comprobar la tabla de encaminamiento.

```
"Node: h1" (on mininet-virtual-machine)
root@mininet-virtual-machine:~# ip r add 10.1.0.2 via 192.168.100.1
root@mininet-virtual-machine:~# ip r add 172.32.0.2 via 192.168.100.1
root@mininet-virtual-machine:~# ip r
10.1.0.2 via 192.168.100.1 dev h1-eth0
172.32.0.2 via 192.168.100.1 dev h1-eth0
192.168.100.0/24 dev h1-eth0 proto kernel scope link src 192.168.100.2
root@mininet-virtual-machine:~#
```

```
"Node: h2" (on mininet-virtual-machine)
root@mininet-virtual-machine:~# ip r add 192.168.100.2 via 172.32.0.1
root@mininet-virtual-machine:~# ip r add 10.1.0.2 via 172.32.0.1
root@mininet-virtual-machine:~# ip r
10.1.0.2 via 172.32.0.1 dev h2-eth0
172.32.0.0/12 dev h2-eth0 proto kernel scope link src 172.32.0.2
192.168.100.2 via 172.32.0.1 dev h2-eth0
root@mininet-virtual-machine:~#
```

```
"Node: h3" (on mininet-virtual-machine)
root@mininet-virtual-machine:~# ip r add 172.32.0.2 via 10.1.0.1
root@mininet-virtual-machine:~# ip r add 192.168.100.2 via 10.1.0.1
root@mininet-virtual-machine:~# ip r
10.0.0.0/7 dev h3-eth0 proto kernel scope link src 10.1.0.2
172.32.0.2 via 10.1.0.1 dev h3-eth0
192.168.100.2 via 10.1.0.1 dev h3-eth0
root@mininet-virtual-machine:~#
```

```
"Node: r0" (on mininet-virtual-machine)
root@mininet-virtual-machine:~# ip r
10.0.0.0/12 dev r0-eth3 proto kernel scope link src 10.1.0.1
172.32.0.0/12 dev r0-eth2 proto kernel scope link src 172.32.0.1
192.168.100.0/24 dev r0-eth1 proto kernel scope link src 192.168.100.1
root@mininet-virtual-machine:~#
```

PROBAR CONECTIVIDAD ENTRE LOS HOSTS

#ping ip_host → Para hacer ping entre las maquinas.

CONECTIVIDAD ENTRE H1 Y H3

```
"Node: h1" (on mininet-virtual-machine)

root@mininet-virtual-machine:~# ping 10.1.0.2
PING 10.1.0.2 (10.1.0.2) 56(84) bytes of data.
 64 bytes from 10.1.0.2: icmp_seq=1 ttl=63 time=16.4 ms
 64 bytes from 10.1.0.2: icmp_seq=2 ttl=63 time=4.01 ms
 64 bytes from 10.1.0.2: icmp_seq=3 ttl=63 time=0.176 ms
 64 bytes from 10.1.0.2: icmp_seq=4 ttl=63 time=0.146 ms
 64 bytes from 10.1.0.2: icmp_seq=5 ttl=63 time=0.149 ms
 64 bytes from 10.1.0.2: icmp_seq=6 ttl=63 time=0.141 ms
 64 bytes from 10.1.0.2: icmp_seq=7 ttl=63 time=2.88 ms
 64 bytes from 10.1.0.2: icmp_seq=8 ttl=63 time=0.058 ms
 64 bytes from 10.1.0.2: icmp_seq=9 ttl=63 time=0.148 ms
 64 bytes from 10.1.0.2: icmp_seq=10 ttl=63 time=0.148 ms
 64 bytes from 10.1.0.2: icmp_seq=11 ttl=63 time=0.146 ms
 64 bytes from 10.1.0.2: icmp_seq=12 ttl=63 time=0.150 ms
 64 bytes from 10.1.0.2: icmp_seq=13 ttl=63 time=0.163 ms
 64 bytes from 10.1.0.2: icmp_seq=14 ttl=63 time=0.145 ms
^C
--- 10.1.0.2 ping statistics ---
14 packets transmitted, 14 received, 0% packet loss, time 13010ms
rtt min/avg/max/mdev = 0.058/1.777/16.422/4.226 ms
root@mininet-virtual-machine:~#
```

CONECTIVIDAD ENTRE H1 Y H2

```
"Node: h1" (on mininet-virtual-machine)

root@mininet-virtual-machine:~# ping 172.32.0.2 -c 5
PING 172.32.0.2 (172.32.0.2) 56(84) bytes of data.
 64 bytes from 172.32.0.2: icmp_seq=1 ttl=63 time=0.118 ms
 64 bytes from 172.32.0.2: icmp_seq=2 ttl=63 time=0.129 ms
 64 bytes from 172.32.0.2: icmp_seq=3 ttl=63 time=0.130 ms
 64 bytes from 172.32.0.2: icmp_seq=4 ttl=63 time=0.130 ms
 64 bytes from 172.32.0.2: icmp_seq=5 ttl=63 time=0.133 ms

--- 172.32.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 0.118/0.128/0.133/0.005 ms
root@mininet-virtual-machine:~#
```

CONECTIVIDAD ENTRE H2 Y H3

```
"Node: h2" (on mininet-virtual-machine)

root@mininet-virtual-machine:~# ping 10.1.0.2 -c 5
PING 10.1.0.2 (10.1.0.2) 56(84) bytes of data.
 64 bytes from 10.1.0.2: icmp_seq=1 ttl=63 time=0.897 ms
 64 bytes from 10.1.0.2: icmp_seq=2 ttl=63 time=0.140 ms
 64 bytes from 10.1.0.2: icmp_seq=3 ttl=63 time=0.135 ms
 64 bytes from 10.1.0.2: icmp_seq=4 ttl=63 time=0.120 ms
 64 bytes from 10.1.0.2: icmp_seq=5 ttl=63 time=0.035 ms

--- 10.1.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4001ms
rtt min/avg/max/mdev = 0.035/0.265/0.897/0.318 ms
root@mininet-virtual-machine:~#
```

La opción -c sirve para indicar cuantos paquetes de ping queremos capturar.

CAPTURA DE TRAFICO DE LA RED EN EL ROUTER

`#tcpdump -i nombre_interfaz` → Se hace en r0 para captura el tráfico.

TRAFICO ENTRE H1 Y H2

```
"Node: r0" (on mininet-vm)
6 packets captured
6 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~# tcpdump -i r0-eth1 -c 3
root@mininet-vm:~# tcpdump -i r0-eth1 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
09:53:43.583069 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 91, length 64
09:53:43.583144 IP 172.32.0.2 > 192.168.100.2: ICMP echo reply, id 2754, seq 91, length 64
09:53:44.582542 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 92, length 64
3 packets captured
4 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~# tcpdump -i r0-eth2 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth2, link-type EN10MB (Ethernet), capture size 262144 bytes
09:53:49.582500 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 97, length 64
09:53:49.582533 IP 172.32.0.2 > 192.168.100.2: ICMP echo reply, id 2754, seq 97, length 64
09:53:50.582601 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 9
```

TRAFICO ENTRE H1 Y H3

```
"Node: r0" (on mininet-vm)
6 packets captured
6 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~# tcpdump -i r0-eth1 -c 3
root@mininet-vm:~# tcpdump -i r0-eth1 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
09:53:43.583069 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 91, length 64
09:53:43.583144 IP 172.32.0.2 > 192.168.100.2: ICMP echo reply, id 2754, seq 91, length 64
09:53:44.582542 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 92, length 64
3 packets captured
4 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~# tcpdump -i r0-eth2 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth2, link-type EN10MB (Ethernet), capture size 262144 bytes
09:53:49.582500 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 97, length 64
09:53:49.582533 IP 172.32.0.2 > 192.168.100.2: ICMP echo reply, id 2754, seq 97, length 64
09:53:50.582601 IP 192.168.100.2 > 172.32.0.2: ICMP echo request, id 2754, seq 9
```

TRAFICO ENTRE H2 Y H3

```
"Node: r0" (on mininet-vm)
root@mininet-vm:~# tcpdump -i r0-eth2 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth2, link-type EN10MB (Ethernet), capture size 262144 bytes
09:58:30.302478 IP 172.32.0.2 > 10.1.0.2: ICMP echo request, id 2871, seq 36, length 64
09:58:30.302548 IP 10.1.0.2 > 172.32.0.2: ICMP echo reply, id 2871, seq 36, length 64
09:58:31.302629 IP 172.32.0.2 > 10.1.0.2: ICMP echo request, id 2871, seq 37, length 64
3 packets captured
4 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~# tcpdump -i r0-eth3 -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on r0-eth3, link-type EN10MB (Ethernet), capture size 262144 bytes
09:58:37.302350 IP 172.32.0.2 > 10.1.0.2: ICMP echo request, id 2871, seq 43, length 64
09:58:37.302377 IP 10.1.0.2 > 172.32.0.2: ICMP echo reply, id 2871, seq 43, length 64
09:58:37.306467 ARP, Request who-has 10.1.0.1 tell 10.1.0.2, length 28
3 packets captured
4 packets received by filter
0 packets dropped by kernel
root@mininet-vm:~#
```