

# ARQUITECTURAS DE REDES AVANZADAS

## PRÁCTICA 1

### ENERO 2015

José Luis Cánovas Sánchez  
jose Luis.canovas2@um.es  
48636907A

#### Resumen

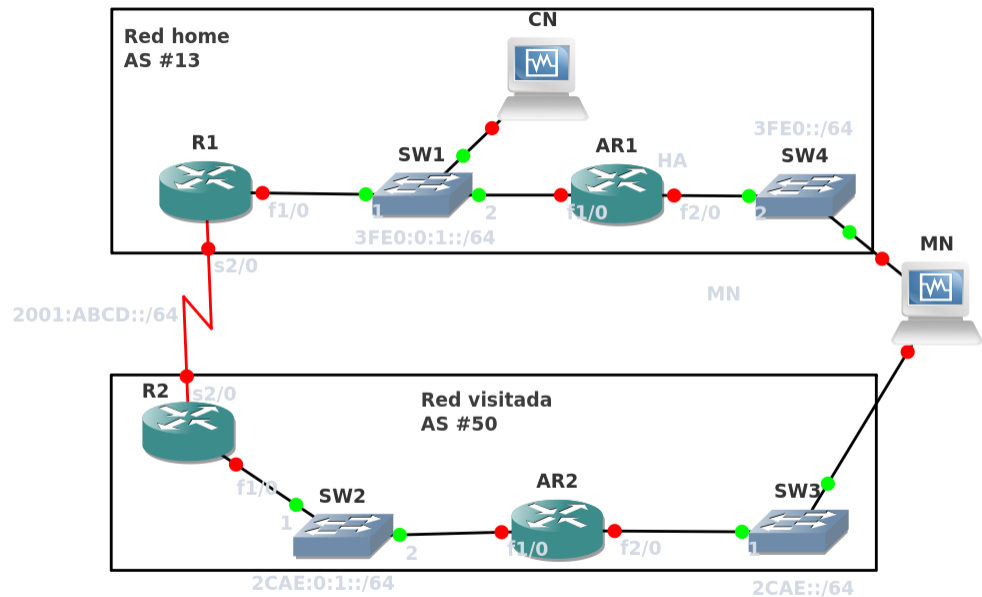
En este informe se redacta el desarrollo usando la herramienta GNS3 para el despliegue del escenario de red IPv6 con movilidad.

## Índice

1. Topología inicial	1
2. Configuración opcional BGP	3
3. Configuración Home Agent	5
4. Correspondent Node	6
5. Mobile Node: MIP6D	6
6. Ping6 CN→N	7
7. Traceroute6 CN→N	8
8. Análisis opcional Neighbour Discovery	8
A. Ping6 salida por pantalla	8

## 1. Topología inicial

Partimos en este proyecto creando una nueva topología en GNS3 equivalente a la del enunciado y que se puede ver en la imagen inferior. Para los routers se utiliza la ios del modelo C7200 de CISCO, con soporte para IPv6. Para los hosts, una imagen linux sin interfaz gráfica y con gestor de paquetes para poder instalar el soporte de mobile IPv6.



La configuración básica para que todos los dispositivos y redes tengan conectividad IPv6 entre ellas se puede realizar por medio del fichero de configuración del router directamente, y su equivalente en comandos de la terminal suelen ser las mismas líneas aplicadas en el nivel *configure terminal*. En cuanto a los hosts, como utilizan la autoconfiguración de IPv6 no hace falta configurar manualmente nada.

Veamos primero la configuración común de todos los routers:

Para activar IPv6 es necesario indicarlo antes de la configuración de las interfaces con *ipv6 unicast-routing*. En cada interfaz que activemos, debemos indicar *ipv6 enable* y la orden *ipv6 address diripv6*, donde *diripv6* es la dirección IPv6 asignada manualmente, o el prefijo de red seguido de *eui-64* para permitir que la interfaz se autoconfigure su dirección.

Con esto tendríamos activado IPv6 en cada router, pero al intentar hacer *ping* a una interfaz de otra red, vemos que no se recibe respuesta. Esto es porque no hay información en las tablas de rutas más que de las redes conectadas directamente a las interfaces de cada router.

Para cada Sistema Autónomo vamos a usar OSPF con soporte de IPv6. El diseño será de una única área 0 para todo el AS, lo que incluye las subredes en AS#13 3FE0:0:1::/64 y 3FE0::/64, y en AS#50 las subredes 2CAE:0:1::/64 y 2CAE::/64. Para ello, debemos activar OSPF en las dos interfaces de los routers AR1 y AR2, y en la interfaz FastEthernet1/0 de los routers R1 y R2 que se pueden ver en la figura de la topología más arriba.

Las órdenes para activar OSPF en cada router son:

Dentro de una interfaz que deba soportar OSPF (está dentro del área 0):

```
1 | ipv6 ospf 100 area 0
```

Tras la configuración de las interfaces:

```
1 | ipv6 router ospf 100
2 |   router-id a.b.c.d
```

Donde *a.b.c.d* es un valor identificador del router con 32 bits que se puede escribir como una dirección IPv4. Para cada router hemos elegido los identificadores *1.1.1.1*, *2.2.2.2*, *3.3.3.3* y *4.4.4.4* para R1, R2, AR1 y AR2 respectivamente. El valor 100 es indicador del proceso OSPF en el router y puede tomar cualquier otro valor, el tomar 100 es arbitrario.

Llegados a este punto, tenemos conectividad dentro de cada área, de modo que ahora un ping desde R1 a la interfaz de AR1 en la subred 3FE0::/64 funciona, pero de AR1 a R2 o AR2 no, porque no se conoce la ruta. Ahora mismo sólo R1 y R2 podrían hacer ping a las interfaces de la red 2001:ABCD::/64.

Para solucionarlo la forma más rápida es añadir una ruta estática en R1 y R2 a las redes del otro AS con la línea: *ipv6 route 2CAE::/42 Serial2/0* en R1, y la equivalente *ipv6 route 3FE0::/42 Serial2/0* en R2. Finalmente, en la configuración de OSPF, tras el *router-id* añadimos la línea *redistribute static*, sólo en R1 y R2, para que comuniquen con LSA5 (rutas fuera de dominio) que para alcanzar el otro AS a través de ellos hay una ruta.

Ahora sí se puede realizar un ping desde cualquier dispositivo a cualquier otro, una vez OSPF ha convergido y compartido todos los mensajes. Pero como la mejora para configurar BGP entre R1 y R2 sobrescribe la configuración estática, mostraré las pruebas en el apartado siguiente.

## 2. Configuración opcional BGP

En los routers R1 y R2 debemos comentar o borrar las líneas de las rutas estáticas y su distribución en OSPF. A continuación, tras la configuración de OSPF podemos escribir la de BGP:

Para R1:

```
1 | router bgp 13
2 |   bgp router-id 1.1.1.1
3 |   no bgp default ipv4-unicast
4 |   !--- Without configuring ""no bgp default ipv4-unicast"" only IPv4 will be !---
   |   advertised
5 |   bgp log-neighbor-changes
6 |   neighbor 2001:ABCD::2 remote-as 50
7 |   !
8 | address-family ipv6
9 |   neighbor 2001:ABCD::2 activate
10 |  network 3FE0:0:1::/64
11 |  network 3FE0::/64
```

Donde 13 indica el número de Sistema Autónomo y 1.1.1.1 es un identificador equivalente al de OSPF, en este caso elegimos el mismo. La línea *no bgp default ipv4-unicast* es necesaria para permitir las rutas con IPv6. La primera línea de *neighbor* indica que en la interfaz de dirección 2001:ABCD::2, que corresponde a la IPv6 fija de R2, se encuentra el AS 50. A continuación, en el bloque *address-family* activamos dicho vecino e indicamos que se deben anunciar las redes 3FE0:0:1::/64 y 3FE0::/64, correspondientes a las del AS 13 de la topología dada.

Para R2 la configuración sería equivalente cambiando los valores del 13 a 50 para el AS, el del *router-id*, la dirección de R1 y las subredes a compartir.

Con esto R1 y R2 conocerían las redes de cada AS, pero AR1 y AR2 todavía no, pues no hablan BGP. Sin embargo, sí hablan OSPF y añadiendo la línea en R1 y R2 *redistribute bgp 10* en la configuración de OSPF (donde antes *redistribute static*) se comunicarán las rutas externas con métrica 10 a los routers de cada AS. Podemos observarlo en la tabla de rutas de AR2, por ejemplo:

```

1 Router>show ipv6 route
2 IPv6 Routing Table - 8 entries
3 Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
4 U - Per-user Static route
5 I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
6 O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
7 ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
8 C    2CAE::/64 [0/0]
9   via ::, FastEthernet2/0
10 L    2CAE::C801:1AFF:FE5F:38/128 [0/0]
11   via ::, FastEthernet2/0
12 C    2CAE:0:1::/64 [0/0]
13   via ::, FastEthernet1/0
14 L    2CAE:0:1:0:C801:1AFF:FE5F:1C/128 [0/0]
15   via ::, FastEthernet1/0
16 OE2   3FE0::/64 [110/2]
17   via FE80::C803:1AFF:FE5F:1C, FastEthernet1/0
18 OE2   3FE0:0:1::/64 [110/1]
19   via FE80::C803:1AFF:FE5F:1C, FastEthernet1/0
20 L    FE80::/10 [0/0]
21   via ::, Null0
22 L    FF00::/8 [0/0]
23   via ::, Null0

```

En las líneas 16 a 19 se puede observar rutas a las redes del AS 13 conocidas por OSPF de rutas externas.

Ahora las pruebas con ping y traceroute:

Ping y traceroute de AR1 a AR2 en la interfaz de la subred 2CAE::/64:

```

1 Router>ping 2CAE::C801:1AFF:FE5F:38
2
3 Type escape sequence to abort.
4 Sending 5, 100-byte ICMP Echos to 2CAE::C801:1AFF:FE5F:38, timeout is 2 seconds:
5 !!!!!
6 Success rate is 100 percent (5/5), round-trip min/avg/max = 60/65/84 ms

```

```

1 Router>traceroute 2CAE::C801:1AFF:FE5F:38
2
3 Type escape sequence to abort.
4 Tracing the route to 2CAE::C801:1AFF:FE5F:38
5
6 1 3FE0:0:1:0:C802:1AFF:FE5F:1C 16 msec 20 msec 20 msec
7 2 2001:ABCD::2 40 msec 40 msec 40 msec
8 3 2CAE::C801:1AFF:FE5F:38 60 msec 60 msec 60 msec
9 Router>

```

Donde 3FE0:0:1:0:C802:1AFF:FE5F:1C es la IP autoconfigurada de R1 en la subred 3FE0:0:1::/64, 2001:ABCD::2 es la IP manual de R2 en la subred 2001:ABCD::/64 y finalmente 2CAE::C801:1AFF:FE5F:38 la IP de AR2.

Considerando que para llegar de AR1 a AR2 con ping y traceroute todos los routers deben conocer el camino en un sentido u otro, con esta única prueba podemos concluir que la configuración de la topología básica con OSPF y BGP es correcta.

### 3. Configuración Home Agent

Según mi topología, el router AR1 por la interfaz FastEthernet2/0 debería actuar como Home Agent para la subred 3FE0::/64. Para activar el proceso del router que actúe como Home Agent basta incluir la siguiente línea en la configuración de la interfaz: **ipv6 mobile home-agent**

Además, incluimos las siguientes configuraciones para el neighbor discovery del router por dicha interfaz, para mejorar la movilidad, al incrementar los anuncios en la subred del router, de modo que el Mobile Node detecta antes los cambios:

```

ipv6 nd prefix 3FE0::/64
ipv6 nd ra-interval 3

```

La primera línea es obvia, indica el prefijo a anunciar, el de la subred. La segunda línea indica el intervalo del Router Advertisement en segundos, en este caso con 3 segundos parece razonable. Para indicarlo en milisegundos habría que añadir **msec** en la línea.

Como el Mobile Node en la prueba pasará a la subred de AR2, incrementamos también el intervalo del Router Advertisement en su interfaz FastEthernet2/0 (la de la red visitada).

## 4. Correspondent Node

El CN no es más que una máquina cualquiera conectada en la red 3FE0:0:1::/64, entre R1 y AR1. La IP que en la ejecución de las pruebas se ha autoconfigurado (por EUI64) es 3fe0:0:1:0:a00:27ff:fe1:1d6d.

No necesita configuración especial para la red. Como máquina virtual de Virtual Box en GNS3, se ha configurado con la interfaz eth0 para el NAT con salida a internet, y eth1 conectado al switch de la topología. En `/etc/network/interfaces` se definen ambas interfaces, y a eth1 se le añade la línea `iface eth1 inet6 auto` de modo que se configura automáticamente una dirección IPv6.

## 5. Mobile Node: MIP6D

El MN es otra máquina virtual conectada a GNS3 con eth0 para el NAT y conectividad a internet, como el CN, y añadimos eth1 y eth2 con `inet6 auto` y conectadas respectivamente a las subredes 3FE0::/64 y 2CAE::/64. La configuración del demonio de movilidad se realiza con el fichero `/usr/local/etc/mip6d.conf` que se muestra a continuación:

```
1 NodeConfig MN;
2 DebugLevel 1;
3 OptimisticHandoff enabled;
4 DoRouteOptimizationMN disabled;
5 MnMaxHaBindingLife 60;
6 Interface "eth1" {
7     MnIfPreference 1;
8 }
9 Interface "eth2" {
10     MnIfPreference 2;
11 }
12 MnHomeLink "eth1" {
13     HomeAgentAddress 3FE0::1;
14     HomeAddress 3fe0::a00:27ff:feed:7e79/64;
15 }
16 UseMnHaIPsec disabled;
17 KeyMngMobCapability disabled;
```

La primera línea indica al demonio que debe actuar como Mobile Node, mostrando mensajes de debug (0 para no mostrar ninguno), sin optimización de ruta y un tiempo de vida de un binding con el Home Agent de 60 segundos.

A continuación se indican las interfaces que formarán parte de mobile ipv6, en este caso eth1 y eth2, con preferencias de a menor valor se elige antes, de modo que de estar disponibles ambas interfaces, se elige antes eth1 (la conectada directamente a la red del HA).

Indicamos también la dirección IPv6 del router AR1 por su interfaz FastEthernet2/0 (IP fija) y la dirección Home Address del MN.

Finalmente indicamos que no haya túnel IPsec entre el Mobile Node y Home Agent.

Para iniciar el demonio basta hacer `sudo mip6d -c /usr/local/etc/mip6d.conf`.

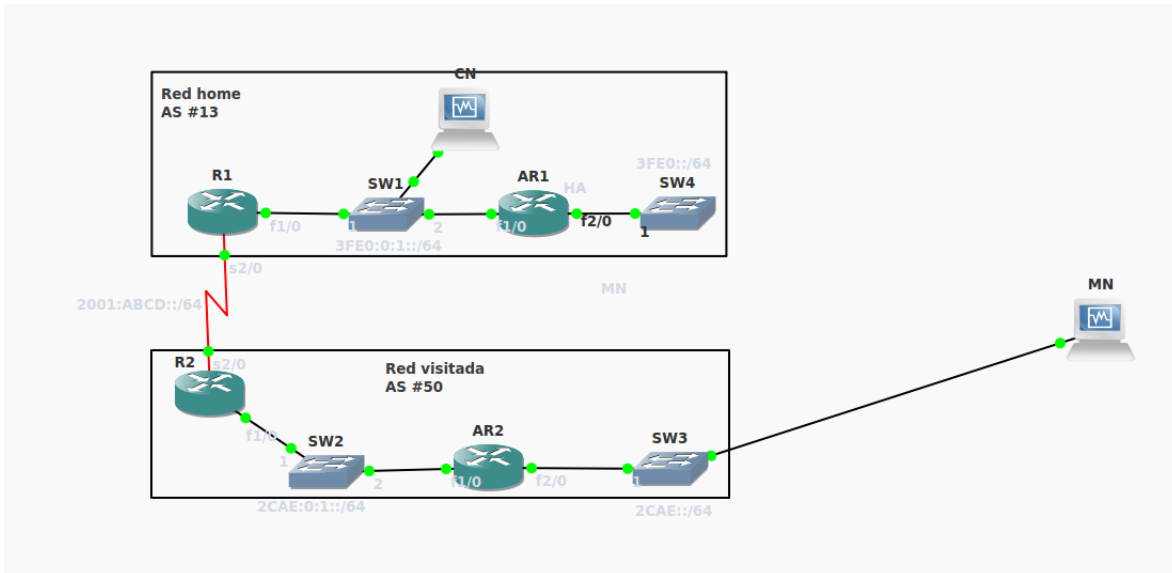
## 6. Ping6 CN→N

Ejecutamos la orden en la máquina del CN:

```
ping6 -I eth1 3fe0::a00:27ff:feed:7e79
```

En las trazas capturadas tenemos:

A continuación se corta el cable en la interfaz de GNS3 que une el MN con la subred de AR1:



Observamos en la salida de *ping6* (ver anexo) que en los primeros mensajes tenemos un tiempo cercano a los 20ms de respuesta, y en el mensaje *icmp\_seq=35* pasamos a tiempos de varios cientos, incluso mil milisegundos, y en el proceso hemos perdido el mensaje *icmp\_seq=34*

En el router AR1 podemos ver también el listado de bindings del Home Agent:

```
Router#show ipv6 mobile binding
Mobile IPv6 Binding Cache Entries:

3FE0::A00:27FF:FEED:7E79
  link local address FE80::A00:27FF:FEED:7E79
  via care-of address 2CAE::A00:27FF:FE7C:6171
  home-agent 3FE0::1
  state ACTIVE, sequence 43521, flags AHLk
  lifetime: remaining 30 (secs), granted 60 (secs), requested 60 (secs)
  interface FastEthernet2/0
  85 tunneled, 61 reversed tunneled
Selection matched 1 bindings
Router#
```

Observamos la dirección del Home Address del MN (la dirección a la que hacemos ping6); la Care-of Address, que pertenece a la subred de AR2 a la que está conectado el MN por la interfaz eth2; la dirección del Home-Agent, la de AR1 por la interfaz FE2/0 que fijamos antes. También notamos cómo tiene en lifetime un tiempo de 60 segundos de vida para cada binding, los configurados en *mip6d.conf*.

Ahora veamos las trazas capturadas:

Finalmente volvemos a unir el cable de eth1 con AR1. Se pierden varios mensajes, casi 50, y en el 154 nos contestan *destino inalcanzable*. Finalmente, en el mensaje 155 volvemos a recibir respuesta y además con un ping cercano a los 20ms iniciales.

## 7. Traceroute6 CN→N

En la prueba de antes hacíamos ping, cortábamos el cable, y volvíamos a unirlo, mientras que el ping6 seguía funcionando. Ahora iniciamos con ping6, cortamos el cable, comprobamos que la movilidad funciona, y ejecutamos sin volver a unir el cable:

```
traceroute6 3fe0::a00:27ff:feed:7e79
```

En la salida del comando observamos que

## 8. Análisis opcional Neighbour Discovery

### A. Ping6 salida por pantalla

```
PING 3fe0::a00:27ff:feed:7e79(3fe0::a00:27ff:feed:7e79)
from 3fe0:0:1:0:a00:27ff:fef1:1d6d eth1: 56 data bytes
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=1 ttl=63 time=44.3 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=2 ttl=63 time=17.1 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=3 ttl=63 time=20.8 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=4 ttl=63 time=14.8 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=5 ttl=63 time=18.6 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=6 ttl=63 time=22.0 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=7 ttl=63 time=15.4 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=8 ttl=63 time=19.4 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=9 ttl=63 time=22.8 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=10 ttl=63 time=16.5 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=14 ttl=63 time=17.3 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=15 ttl=63 time=20.5 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=16 ttl=63 time=23.6 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=17 ttl=63 time=15.3 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=18 ttl=63 time=18.0 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=19 ttl=63 time=20.9 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=20 ttl=63 time=22.1 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=21 ttl=63 time=14.0 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=22 ttl=63 time=16.9 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=23 ttl=63 time=18.9 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=24 ttl=63 time=21.5 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=25 ttl=63 time=23.6 ms
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp_seq=26 ttl=63 time=16.0 ms
```



64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=27 ttl=63 time=19.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=28 ttl=63 time=22.9 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=29 ttl=63 time=15.9 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=30 ttl=63 time=18.0 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=31 ttl=63 time=21.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=32 ttl=63 time=13.9 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=33 ttl=63 time=17.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=35 ttl=63 time=410 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=36 ttl=63 time=1067 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=37 ttl=63 time=198 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=38 ttl=63 time=202 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=39 ttl=63 time=247 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=40 ttl=63 time=583 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=41 ttl=63 time=1060 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=42 ttl=63 time=1066 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=43 ttl=63 time=1071 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=44 ttl=63 time=1064 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=45 ttl=63 time=807 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=46 ttl=63 time=1063 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=47 ttl=63 time=197 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=48 ttl=63 time=244 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=49 ttl=63 time=1062 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=50 ttl=63 time=845 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=51 ttl=63 time=86.1 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=52 ttl=63 time=1075 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=53 ttl=63 time=1073 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=54 ttl=63 time=1067 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=55 ttl=63 time=1062 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=56 ttl=63 time=172 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=57 ttl=63 time=177 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=58 ttl=63 time=222 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=59 ttl=63 time=1082 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=60 ttl=63 time=817 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=61 ttl=63 time=1074 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=62 ttl=63 time=1071 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=63 ttl=63 time=1066 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=64 ttl=63 time=1071 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=65 ttl=63 time=975 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=66 ttl=63 time=1072 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=67 ttl=63 time=155 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=68 ttl=63 time=200 ms

64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=69 ttl=63 time=1070 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=70 ttl=63 time=194 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=71 ttl=63 time=1074 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=72 ttl=63 time=790 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=73 ttl=63 time=191 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=74 ttl=63 time=1059 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=75 ttl=63 time=984 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=76 ttl=63 time=1080 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=77 ttl=63 time=121 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=78 ttl=63 time=175 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=79 ttl=63 time=1074 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=80 ttl=63 time=769 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=81 ttl=63 time=1075 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=82 ttl=63 time=1074 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=83 ttl=63 time=1070 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=84 ttl=63 time=1067 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=85 ttl=63 time=1072 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=86 ttl=63 time=311 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=87 ttl=63 time=94.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=88 ttl=63 time=137 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=89 ttl=63 time=111 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=90 ttl=63 time=97.1 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=91 ttl=63 time=92.0 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=92 ttl=63 time=187 ms  
From 3fe0:0:1:0:c800:44ff:fe8d:1c icmp\_seq=154 Destination unreachable: Address unreachable  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=155 ttl=63 time=21.6 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=156 ttl=63 time=14.4 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=157 ttl=63 time=18.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=158 ttl=63 time=21.4 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=159 ttl=63 time=14.5 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=160 ttl=63 time=17.5 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=161 ttl=63 time=21.1 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=162 ttl=63 time=14.9 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=163 ttl=63 time=17.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=164 ttl=63 time=20.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=165 ttl=63 time=13.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=166 ttl=63 time=16.4 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=167 ttl=63 time=19.9 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=168 ttl=63 time=22.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=169 ttl=63 time=15.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=170 ttl=63 time=18.1 ms

64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=171 ttl=63 time=11.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=172 ttl=63 time=14.8 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=173 ttl=63 time=16.6 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=174 ttl=63 time=18.0 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=175 ttl=63 time=21.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=176 ttl=63 time=14.8 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=177 ttl=63 time=17.2 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=178 ttl=63 time=20.1 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=179 ttl=63 time=13.3 ms  
64 bytes from 3fe0::a00:27ff:feed:7e79: icmp\_seq=180 ttl=63 time=15.8 ms

## Referencias

- [1] Apuntes