UNIVERSIDAD NACIONAL DE CÓRDOBA
Facultad de Ciencias Exactas, Físicas y Naturales

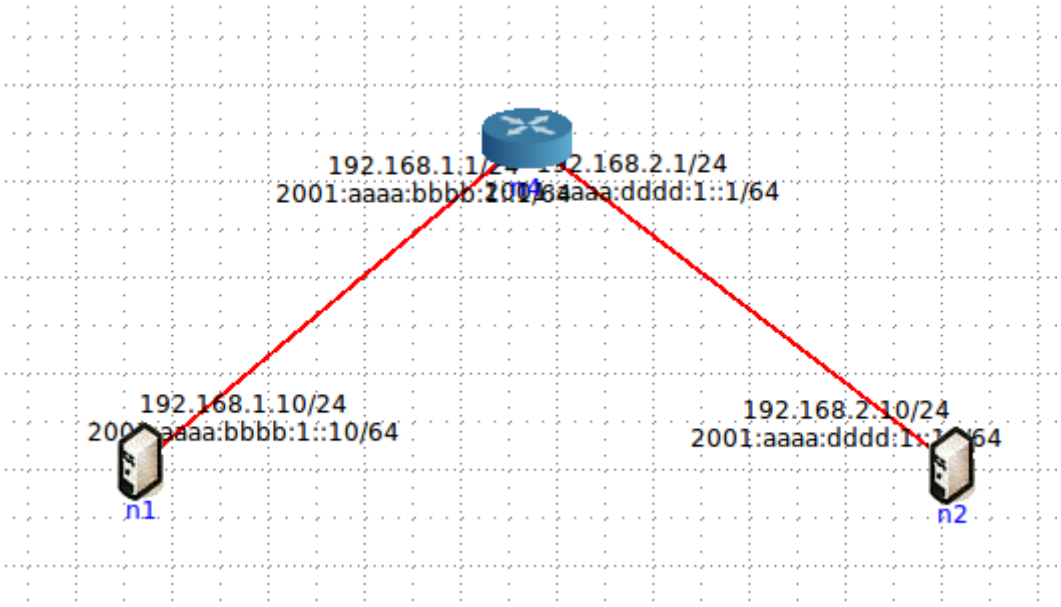
Redes de Computadoras

Práctico 1: Tráfico en capa de enlace relacionado con IPv4 e IPv6.

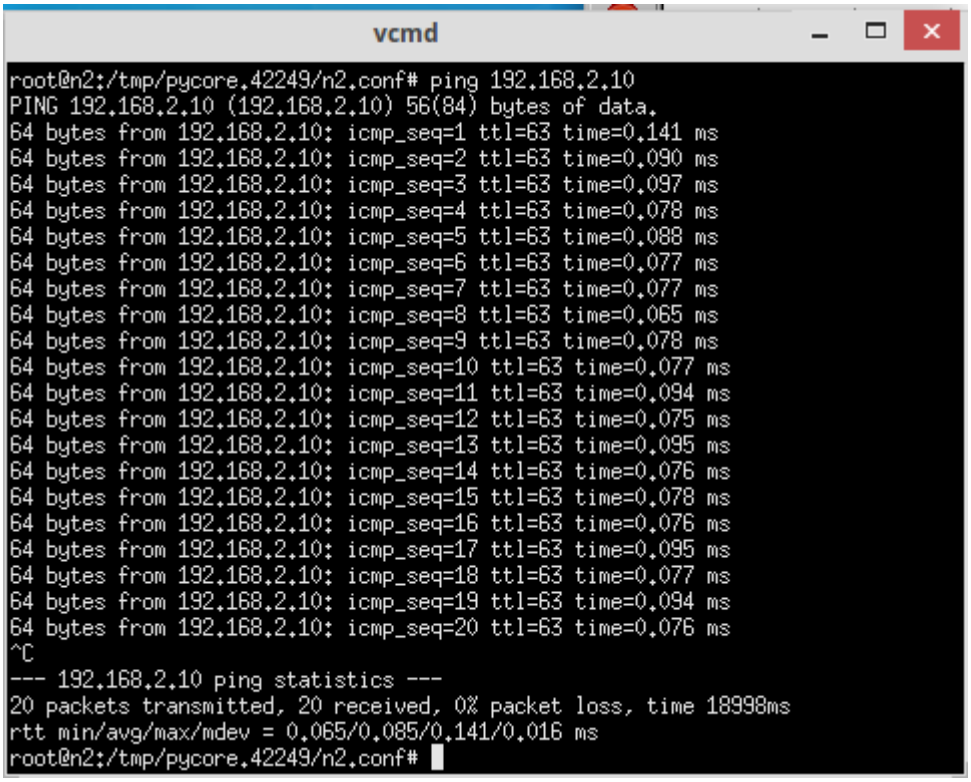
Contrera, Ivan
Malano, Leandro

PARTE 1:

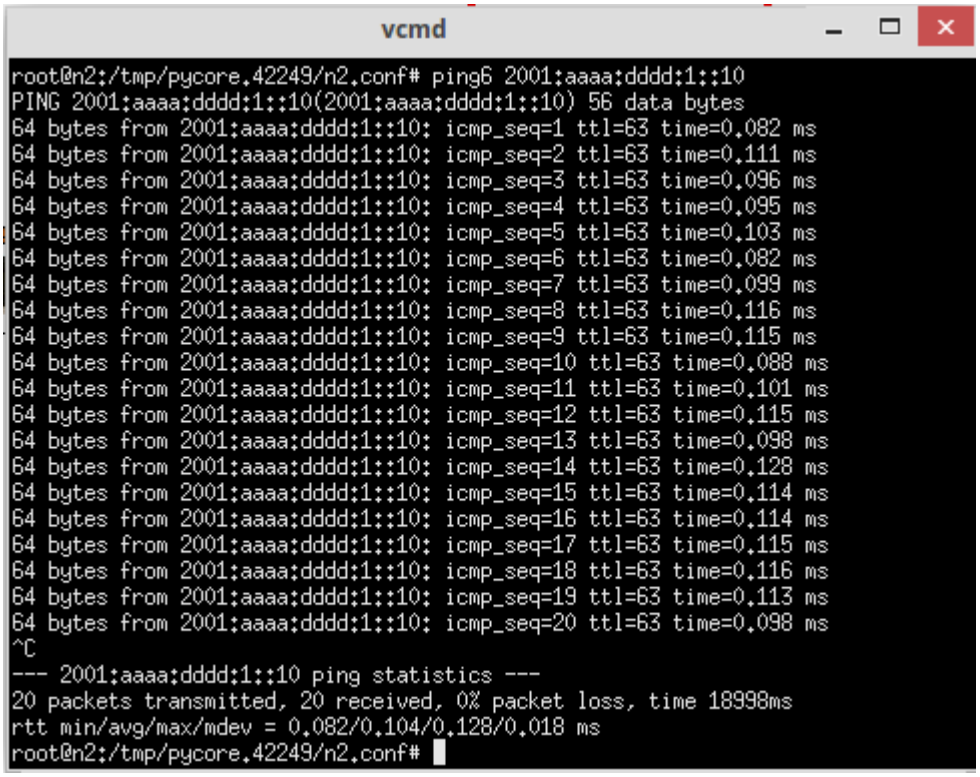
1) En primera instancia, se arma el esquema pedido en el software core, con los dos hosts conectados a un router estándar:



2) Una vez armado y configurado el esquema en Core, se probó la conectividad entre los dos hosts, obteniendo un resultado positivo:

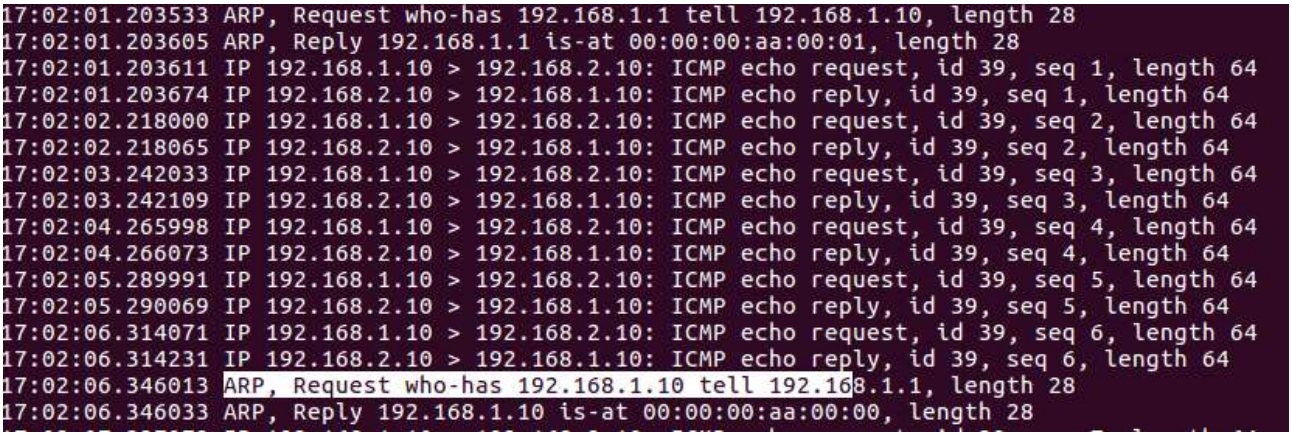


3) Lo mismo se hizo para el caso de IPv6:

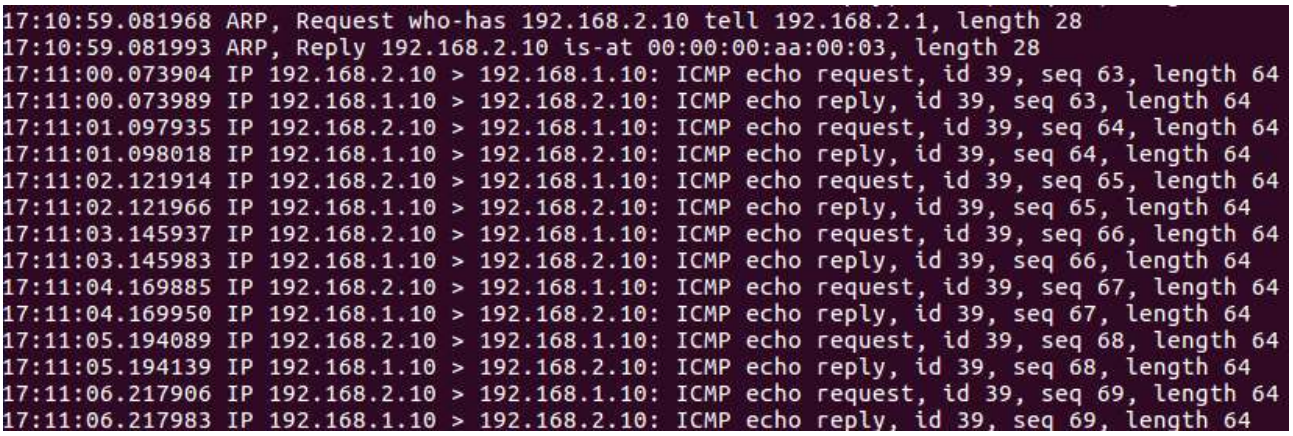


4) Utilizando el software tcpdump, a continuación se muestran capturas de los paquetes enviados y recibidos desde cada pc. Al principio de cada una, se observan las solicitudes y respuestas ARP, ya que en principio cada host no conocía la MAC de la interfaz del router conectada directamente:

desde PC1:



desde PC2:



4,1)

Suceden comunicaciones entre el host1 y la interfaz 1 del router, donde en un mensaje de solicitud, el host pregunta mediante un mensaje de broadcast cual es la MAC de la interfaz del router, y este responde devolviendo dicha dirección. También existe comunicación entre el host2 y la interfaz 2 del router, enviando y recibiendo mensajes similares. Todos son paquetes especiales ARP.

4,2) Se observan las direcciones ipv4 de los hosts cuyos valores son 192.168.1.10 y 192.168.2.10

4,3) Sabe cómo comunicar un host con otro host ya que estos están conectados directamente a dicho router y se agregan los valores de ambas redes a la tabla de enrutamiento automáticamente al hacer la configuración.

4,4) No es necesario contar con un switch para esta topología ya que existe nada más un host por cada red

4,5) la tabla ARP del host de origen (Host1) contiene la ip de la interfaz 1 del router con su respectiva dirección MAC, información de que la conexión es mediante ethernet, y la interfaz del router.

```
root@n1:/tmp/pycore.44729/n1.conf# arp
Dirección                TipoHW  DirecciónHW         Indic Máscara         Inter
faz
192.168.1.1              ether   00:00:00:aa:00:01   C                     eth0
```

4,6) la tabla ARP del host de destino contiene la ip de la interfaz 2 del router con su MAC, y el resto de la información es idéntica al caso anterior.

```
root@n2:/tmp/pycore.44729/n2.conf# arp
Dirección                TipoHW  DirecciónHW         Indic Máscara         Inter
faz
192.168.2.1              ether   00:00:00:aa:00:02   C                     eth0
```

4,7) la tabla ARP el router contiene las direcciones ip de los hosts con sus respectivas direcciones MAC, y como en los casos anteriores, indica la interfaz física donde está conectado cada host.

```
root@n4:/tmp/pycore.44729/n4.conf# arp
Dirección                TipoHW  DirecciónHW         Indic Máscara         Inter
faz
192.168.2.10             ether   00:00:00:aa:00:03   C                     eth1
192.168.1.10             ether   00:00:00:aa:00:00   C                     eth0
```

4,8) las direcciones de broadcast en ipv4, son direcciones ip especiales utilizadas para la difusión de datos dentro de una red, donde un solo paquete enviado por el transmisor, llegará a todos los hosts de la misma. Estas direcciones tienen siempre el valor más alto del rango de la red. Por ejemplo, para la red 20.100.25.0/24, la ip de broadcast sería la 20.100.25.255. Se suelen utilizar en los algoritmos de enrutamiento para descubrir vecinos y verificar si siguen conectados o no.

4,9) las direcciones multicast en ipv4, son direcciones donde se pueden enviar datos a múltiples hosts de múltiples redes. Se utiliza únicamente como experimento, a diferencia de ipv6, donde es obligatorio contar con una dirección multicast, siendo un concepto algo distinto. Se podría llegar a usar en videoconferencias y en transmisiones de radio y televisión por internet.

5,1) suceden 4 comunicaciones: neighbor solicitation, la cual es una solicitud para descubrir direcciones mac de vecinos o para verificar que sigue conectado. Neighbor advertisement, como respuesta a esa solicitud. Router solicitation, para descubrir routers de la red, y Advertisement como respuesta a dicha solicitud.

5.2) NDP NO reemplaza a ARP

5.3) Existen muchas diferencias: NDP detecta la inaccesibilidad de vecinos, utiliza ICMPv6 para detectar vecinos en lugar de definir otro tipo de paquete como el ARP, detectar direcciones duplicadas (ya que como se configuran automáticamente, existe el riesgo de error en caso de tener equipos con identificadores repetidos), permite que los dispositivos conozcan los distintos parámetros del enlace físico (como el número máximo de saltos permitidos). ARP no tiene ninguna de esas características. Otra diferencia importante es que las solicitudes en NDP se hacen en multicast, donde solo las recibirán los terminales que estén unidos a dicha difusión, a diferencia de ARP que hace solicitudes a través de un broadcast, trayendo el problema de que si la red es muy grande, podría saturarse la misma y ralentizar el tráfico de datos de usuarios.

5.4) algunas de las características más importantes de NDP son las siguientes:

- Se utiliza para descubrir enrutadores en forma automática en una red IPv6 usando los mensajes router solicitation y router advertisement.
- Se pueden descubrir prefijos de red dinámicamente utilizando los mensajes router solicitation y router advertisement.
- Detección de direcciones duplicadas. Dado que las direcciones se configuran de forma automática, existen riesgos de error en caso de tener equipos con identificadores repetidos. Este protocolo verifica que ningún otro equipo en el enlace tiene la misma dirección IPv6.
- Resuelve dinámicamente direcciones MAC, utilizando los mensajes neighbor solicitation y neighbor advertisement, la cual es una función muy parecida a la de ARP.
- Auto configura las direcciones IPv6 de los dispositivos una vez que aprende los prefijos de red, ya que genera automáticamente la parte del host utilizando el estándar IEEE EUI-64.

5.5) IPv6 no tiene direcciones broadcast, sino que usa comúnmente direcciones de multidifusión. La diferencia está en que con multicast solo determinados hosts (los necesarios) se unirán a ese grupo, y solo ellos recibirán los mensajes que tengan como destino esa dirección. Con broadcast no hay suscripciones, todos los hosts de la red van a recibir los mensajes, a veces innecesarios, causando congestión en la red interna.

5.6) Las direcciones link-local se utilizan en un enlace sencillo y no debe nunca ser enrutada. Se usa para mecanismos de autoconfiguración, descubrimiento de vecinos y en redes sin ruteadores. Es útil para crear redes temporales. Puede ser utilizada sin un prefijo global. Las direcciones site-local contienen información de subred dentro de la dirección. Son enrutadas dentro de un sitio, pero los ruteadores no deben enviarlas fuera de éste. Además es utilizada sin un prefijo global. Y las direcciones globales son utilizadas para el tráfico de IPv6 genéricos en el Internet de IPv6 y son similares a las direcciones unicast usadas para comunicarse a través del Internet de IPv4.

EJERCICIO 2:

1) Se comienza configurando el router (ubuntu server) con las ip estáticas del archivo “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

# The primary network interface

auto enp0s3
        iface enp0s3 inet static
                address 192.168.1.1
                netmask 255.255.255.0
                #gateway 192.168.1.1
        iface enp0s3 inet6 static
                address 2001:aaaa:bbbb:1::1
                netmask 64


auto enp0s8
        iface enp0s8 inet static
                address 192.168.2.1
                netmask 255.255.255.0
                #gateway 192.168.2.1
        iface enp0s8 inet6 static
                address 2001:aaaa:dddd:1::1
                netmask 64
```

2) Luego se configura el router para que realice ip-forwarding (o reenvío de paquetes ip) de manera permanente.

3)Luego se configura cada host con las ip solicitadas

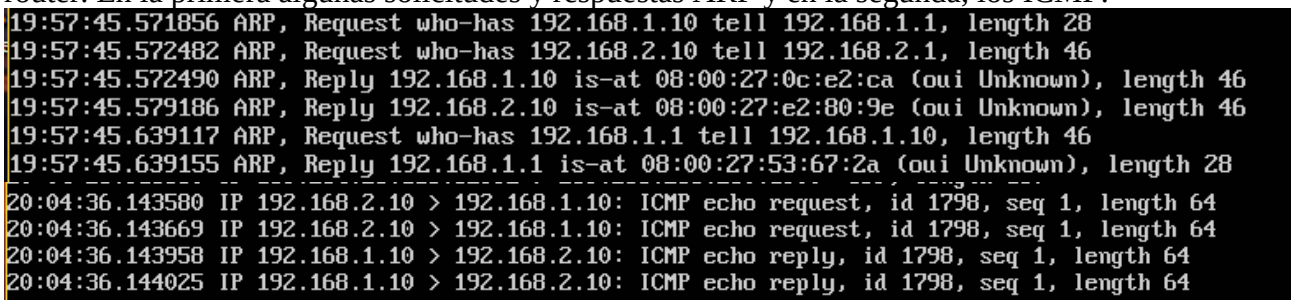
4.1) Para IPv4, apenas un host se conecta a la red, su tabla ARP (y la del router) no tienen información mutua de sus MAC. Cuando se intenta hacer un ping a la puerta de enlace predeterminada (default gateway), primero genera una solicitud ARP para la ip de dicho destino. Cuando la interfaz del router la recibe, responde con un mensaje de respuesta ARP indicando su MAC. Luego el host recibe la información necesaria y la agrega a la tabla ARP. Ahora éste ya sabe a que MAC enviar el paquete ICMP, y lo envía. En este trabajo, la interfaz del ubuntu server que corresponde a la red 192.168.2.0 en la maquina virtual, recibe este paquete, y en respuesta genera otro ICMP de respuesta y lo envía al host 192.168.2.10, que pertenece a la máquina virtual instalada en la otra PC fisica. La siguiente captura muestra lo explicado anteriormente utilizando tcpdump desde el ubuntu server:

```
15:34:55.233890 ARP, Request who-has 192.168.2.1 tell 192.168.2.10, length 46
15:34:55.233927 ARP, Reply 192.168.2.1 is-at 08:00:27:fb:39:90 (oui Unknown), length 46
15:34:55.234471 IP 192.168.2.10 > 192.168.2.1: ICMP echo request, id 1749, seq 1, length 64
15:34:55.234848 IP 192.168.2.1 > 192.168.2.10: ICMP echo reply, id 1749, seq 1, length 64
```

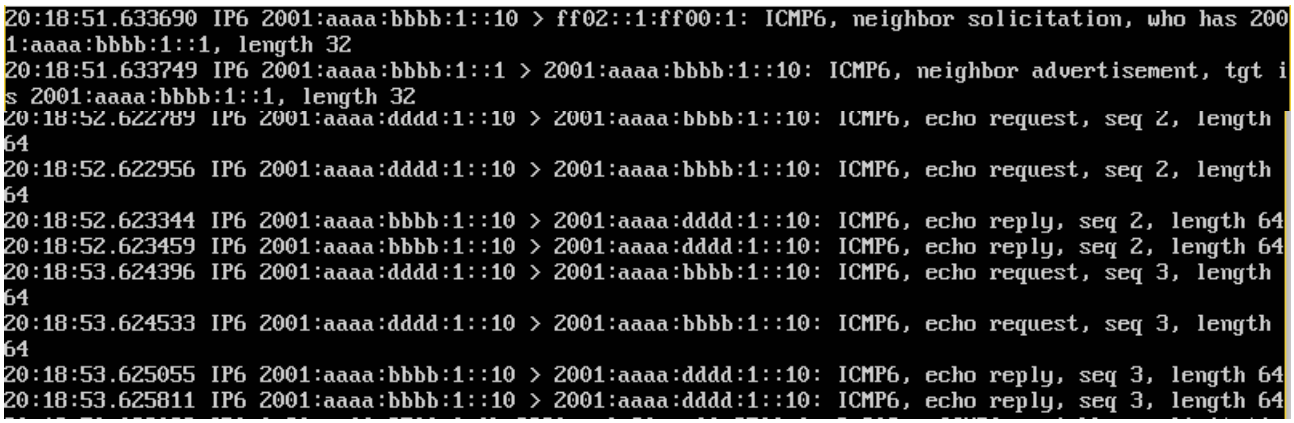
Para IPv6, como el host 192.168.2.10 no conoce la MAC de la interfaz del ubuntu server con la que se conecta directamente, aplica el protocolo NDP, enviando un datagrama ICMPv6 del tipo neighbor solicitation, teniendo la función similar a ARP. Ubuntu server (router) recibe el datagrama, interpreta el mensaje y responde al host con otro ICMPv6 del tipo neighbor advertisement indicando la MAC solicitada y dicho receptor la agrega a una tabla de correspondencia. Luego está en condiciones de enviar otro ICMPv6 del tipo echo request (ping),el cual es recibido por el router, lo procesa y responde con otro ICMPv6 del tipo echo reply. Lo explicado se muestra en la siguiente captura:

```
16:09:41.130032 IP6 2001:aaaa:dddd:1::10 > ff02::1:ff00:1: ICMP6, neighbor solicitation, who has 200
1:aaaa:dddd:1::1, length 32
16:09:41.130178 IP6 2001:aaaa:dddd:1::1 > 2001:aaaa:dddd:1::10: ICMP6, neighbor advertisement, tgt i
s 2001:aaaa:dddd:1::1, length 32
16:09:41.131483 IP6 2001:aaaa:dddd:1::10 > 2001:aaaa:dddd:1::1: ICMP6, echo request, seq 1, length 6
4
16:09:41.131866 IP6 2001:aaaa:dddd:1::1 > 2001:aaaa:dddd:1::10: ICMP6, echo reply, seq 1, length 64
16:09:42.131823 IP6 2001:aaaa:dddd:1::10 > 2001:aaaa:dddd:1::1: ICMP6, echo request, seq 2, length 6
4
16:09:42.131838 IP6 2001:aaaa:dddd:1::1 > 2001:aaaa:dddd:1::10: ICMP6, echo reply, seq 2, length 64
```

4.2) Para IPv4, sucede una situación similar a la anterior, solo que el datagrama ICMP llega hasta el host 192.168.1.10. Si el emisor no tiene en su tabla de correspondencias la MAC del router, aplica ARP para obtenerla. Luego envía el ICMP, el router lo recibe, abre el paquete e interpreta que tiene que enviarlo a la red 192.168.1.0, y de acuerdo a la tabla de enrutamiento, lo reenvía por el puerto indicado y así llega al receptor, el cual lo desempaqueta, analiza el tipo de dato que llegó y en respuesta envió otro ICMP. en las siguientes capturas se muestra el tráfico de datos visto por el router. En la primera algunas solicitudes y respuestas ARP y en la segunda, los ICMP:



Para IPv6, se produce lo mismo que en el caso del punto 4.1, en un principio el host emisor 2001:aaaa:dddd:1::10 utiliza el protocolo NDP enviando un paquete ICMPv6 para obtener la MAC del puerto conectado directamente con el router, y luego esperando su respuesta. Después le envía el ICMPv6 de echo request, se procesa y se reenvía a la red 2001:aaaa:bbbb:1:: y llega al host receptor. Si no llegara a existir la MAC de este último en el router, se aplica nuevamente NDP antes de reenviar el ICMPv6. Acto seguido se envía una respuesta, y no se necesitará aplicar nuevamente NDP. Las siguientes capturas fueron hechas desde el router:



parte 3

1. linux namespace es una característica de los sistemas operativos linux (del kernel) el cual divide los recursos del kernel, de modo que un proceso tenga aislado los recursos del hardware. De esta forma un recurso puede ser tomado por un " espacio de nombre".

   Cuando se necesita aislar un recurso hardware a un grupo de proceso (contenedor), este dependerá del tipo de namespace. Todos los procesos son asociados con un namespace y solo podrán utilizar los recursos únicamente asociados a ese namespace.

   Existen 6 namespaces:

   mnt: controla el aislamiento de los distintos puntos de montaje.
   pid: encargado de asignar un nuevo PID a cada proceso.
   net: proporciona el aislamiento de los recursos asociados con el networking (devices, IPv4, IPv6, etc…)
   igipc: identificadores IPC de SysV

uts: nombres de hosts y dominios
user: identificadores de usuarios y grupos.

Podemos ver estos grupos bajo el directorio /proc/[PID]/ns/

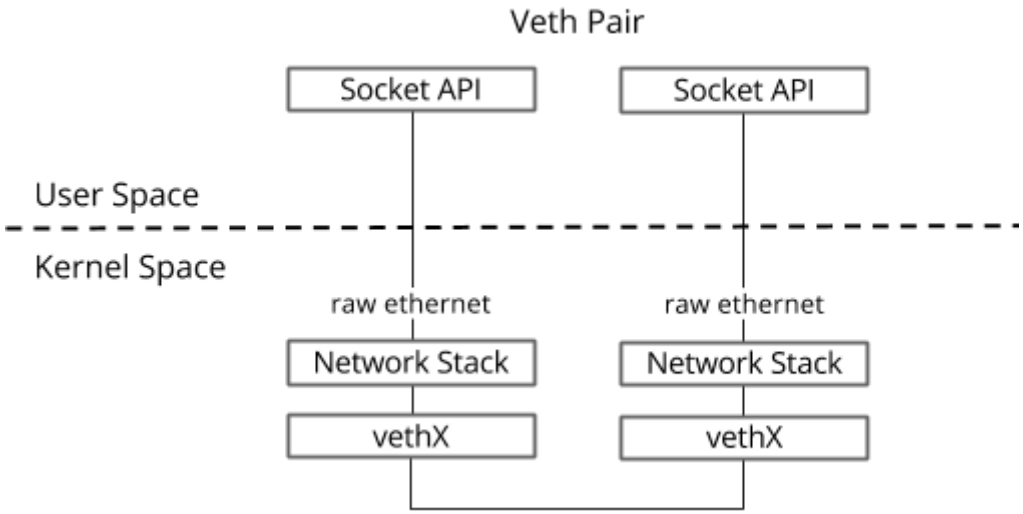
En definitiva los namespace se utilizan para agrupar procesos incluyendo árboles de procesos, red, ID de usuario y sistemas de archivos montados, para posteriormente aislarlos a nivel de recursos del resto de procesos pertenecientes a otros namespaces. (poner referencia a link https://nebul4ck.wordpress.com/2016/08/12/namespace-linux/)

2. el término brigde hace referencia a un puente que une dos redes diferentes por medio del sistema operativo linux.Trabaja en capa 2 (capa de enlace), ya que los paquetes se envían de acuerdo a la dirección ethernet y no por la dirección ip de destino. emula a un dispositivo físico bridge, el cual nomas ve la dirección física de destino del paquete a enviar.

(ref https://outlyer.net/howtos-linux/bridge-monopuerto/#pre || https://wiki.linuxfoundation.org/networking/bridge || https://claudiooq2.wordpress.com/switch-hub-router-bridge/)

3. A la hora de crear una red virtual, para tunelizar y reenviar paquetes dentro un sistema, utilizamos veth pair.

Los dispositivos veth funcionan como pares de interfaces de eth virtual conectadas extremo a extremo (considerado como un cable de conexión virtual)



FUENTES:

http://livre.g6.asso.fr/index.php/Protocolo_de_Descubrimiento_de_vecinos

http://www.ipv6.mx/index.php/informacion/fundamentos/ipv6

http://www.yourownlinux.com/2013/07/how-to-configure-ubuntu-as-router.html

http://www.omnisecu.com/tcpip/ipv6/ndp-neighbour-discovery-protocol-functions-of-ndp.php