Práctica 4 – Networking entre hosts con contenedores

1. Introducción y objetivos

El objetivo de esta práctica es probar escenarios de comunicación entre contenedores que se encuentran en diferentes Kernels Linux (en diferentes hosts). Emplearemos como hosts para estos contenedores VMs de VirtualBox. Serán escenarios sin el empleo de overlays.

2. Escenario puenteado a través de LAN

En este escenario tendremos dos hosts (que en realidad son VMs), cada uno de ellos con contenedores configurados, tal que todos esos contenedores acaben estando en la misma LAN, que será la LAN de los hosts. Se ha intentado representar esto en la Figura 1.

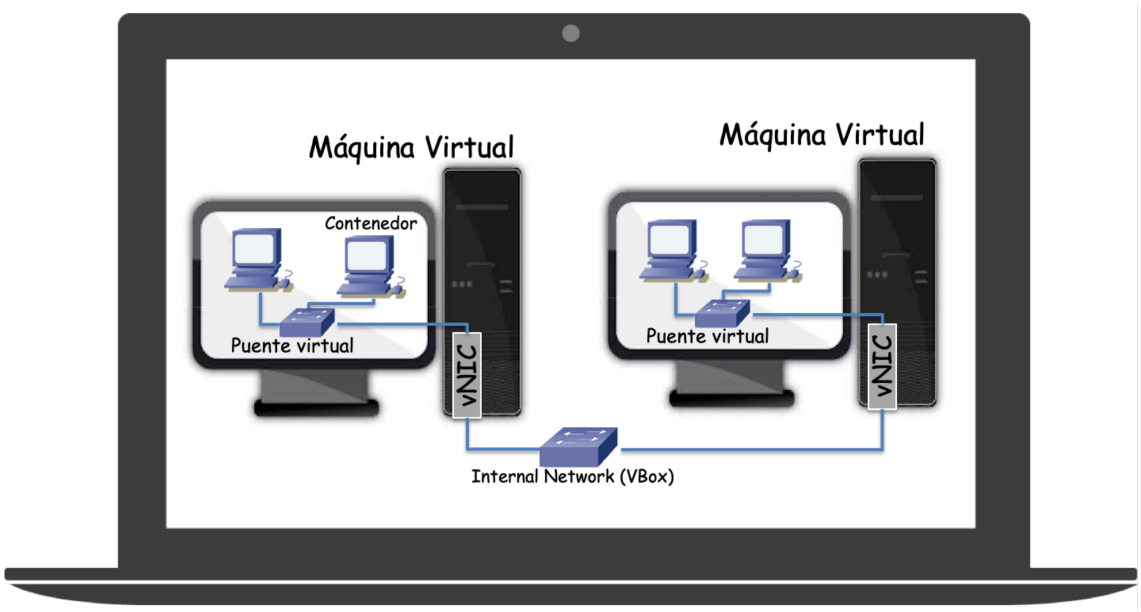

Figura 1 - Escenario puenteado a través de LAN

Prepare dos VMs con LXD como hizo en la práctica anterior (o prepare una y clónela), descargando previamente alguna imagen de contenedor en ellas. Ambas VMs tendrán un interfaz conectado a una *Internal Network*.

En uno de los hosts cree un bridge y añada al mismo el interfaz físico de la internal network.

A continuación, lance un contenedor en ese host y conecte ese contenedor mediante veths con el bridge.

Repita la configuración en el segundo host. Asigne dirección IP a los contenedores que se encuentren en los dos hosts, todas en la misma subred IP. Debería poder hacer ping de un contenedor a otro siguiendo el escenario puenteado tal y como se describe en la Figura 1. Puede ver el tráfico con *tcpdump,* por ejemplo, en uno de los hosts, y verá las tramas Ethernet con las direcciones MAC de los interfaces virtuales de los contenedores. Esto es tráfico que está circulando por la internal network creada en VirtualBox. En un escenario donde los hosts no son VMs sino el host nativo de un servidor este tráfico estaría circulando por uno de los conmutadores hardware de nuestra infraestructura de red (elimine de la Figura 1 el portátil que engloba a todo el escenario y se ha quedado con un switch hardware con dos PCs que corren los contenedores).

En este punto no hemos configurado dirección IP a los hosts (las máquinas virtuales en VirtualBox). Si quisiéramos darles dirección IP deberíamos asignarla al interfaz del puente,

dado que el interfaz físico está conectado al mismo. Esto es similar a lo que hacemos en un switch Cisco capa 2/3 donde no asignamos dirección IP al interfaz físico, sino que lo hacemos al interfaz lógico de la VLAN porque el interfaz físico está funcionando en modo conmutación en capa 2 (en muchos switches capa 2/3 se puede cambiar el modo de funcionamiento de ese interfaz para que pase a ser un interfaz capa 3, que sería como si sacáramos nuestro interfaz del host del puente).

Reflexione sobre las direcciones MAC que son vistas por el switch que interconecta los hosts.

Punto de control 1 (50%): Muestre al profesor el escenario en funcionamiento.

3. Escenario enrutado

Este escenario está representado en la Figura 2. El cambio en el dibujo respecto al escenario puenteado es sutil. Ahora cada host actúa como un router, de forma que sus contenedores están en una subred IP diferente de los contenedores del otro host. La interconexión entre los hosts la hemos mantenido con una internal network de VirtualBox (es decir, en capa 2) pero valdría cualquier conectividad en capa 3 (es decir, podría haber routers entre los hosts). A nivel lógico capa 3 las subredes existentes se ven en la Figura 3.

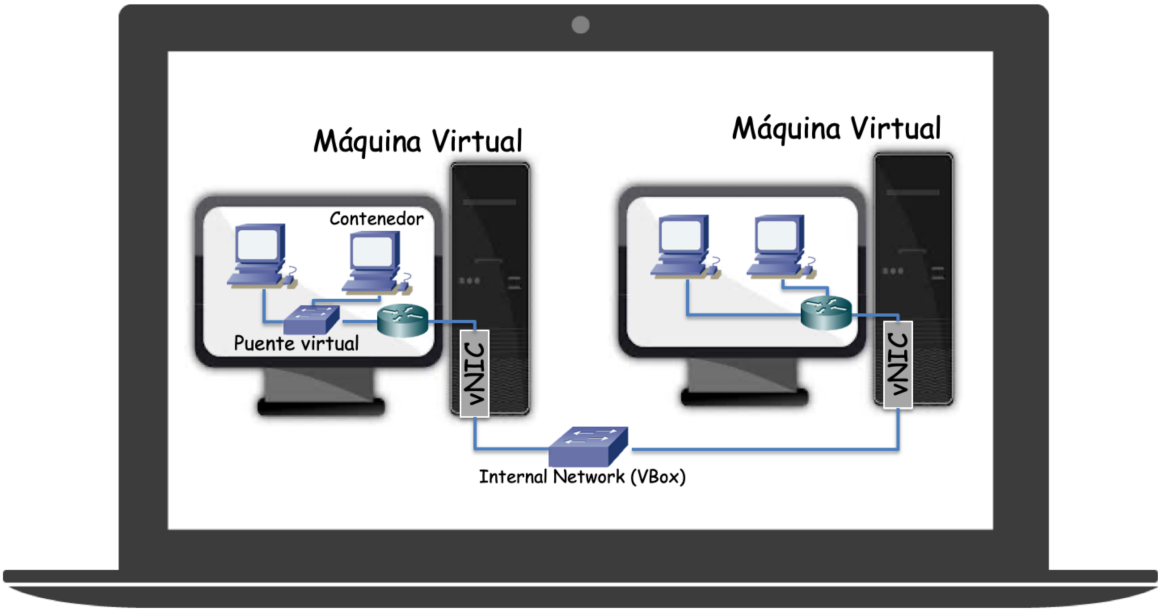

Figura 2 - Escenario enrutado

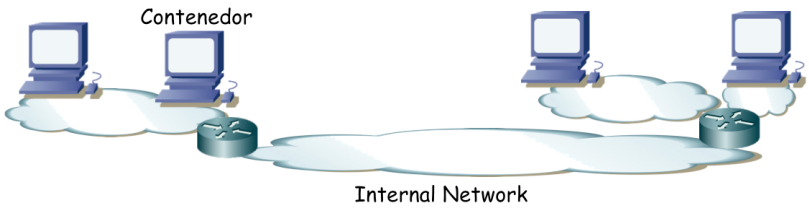

Figura 3 - Topología capa 3 del escenario enrutado

Partiremos de los hosts sin la configuración del escenario puenteado.

Asigne dirección IP a los interfaces físicos de los hosts en la internal network en una subred propia para ella. Deberían alcanzarse el uno al otro (es decir, que pueden hacerse ping). Los hosts van a enrutar así que hay que activar el reenvío de paquetes IP en ellos. Por simplicidad pondremos una ruta por defecto de cada uno apuntando al otro.

Cree un puente en uno de los hosts. Configure en el interfaz del puente la dirección IP que quiera que tenga el router en la subred de los contenedores. Lance un par de contenedores y enlácelos con el puente mediante veths. Configure en los veths internos de los contenedores dirección IP de esa misma subred y como siguiente salto en su ruta por defecto ponga la dirección IP que asignó al bridge.

En el segundo host podríamos hacer una configuración similar y tendríamos resuelto el escenario, pero vamos a hacerla ligeramente diferente para practicar con el entorno virtualizado. No vamos a utilizar el bridge. Cree un contenedor y un par de veths asignando uno al contenedor. El veth en el namespace del host es el interfaz del router virtual de un enlace punto a punto con el contenedor así que puede emplear una subred con prefijo de 30 bits. Ahora dentro del contenedor configure la dirección IP del otro extremo de ese veth con la otra dirección IP de la subred y la ruta por defecto apuntando a la dirección IP del router. En la Figura 2 se han representado dos contenedores con una configuración como ésta descrita.

Con eso ya debería funcionar el ping entre los contenedores de los dos hosts. Con *traceroute* o con *ping –R* podrá ver los saltos que dan los paquetes. También puede observarlos con *tcpdump* en los hosts y ver las direcciones MAC que aparecen en esas tramas.

Reflexione sobre las direcciones MAC que son vistas por el switch que interconecta los hosts.

Punto de control 2 (45%): Muestre al profesor el escenario en funcionamiento.

4. Control remoto

Con el comando lxc puede controlar demonios lxd que se encuentren en máquinas diferentes. Busque documentación[1] sobre cómo configurar en uno de los dos hosts que pueda controlar el demonio lxd del otro y pruebe con ello a lanzar y detener contenedores en él o incluso a ejecutar procesos en un contenedor que se encuentra en otro host a través del comando lxc.

Punto de control 3 (5%): Muestre al profesor control desde un host de contenedores de otro.

[1] https://documentation.ubuntu.com/lxd/en/latest/remotes/