11.1 Solución

Básicamente una VPN es un canal de comunicación (una simple conexión TCP es suficiente) sobre la que se despliega una pila entera de protocolos. Los dispositivos TUN son adecuados para esto y la herramienta SOCAT va a permitir el establecimiento de la conexión TCP y su uso para comunicar un dispositivo TUN en cada extremo. A partir del establecimiento de este túnel para comunicar datos, lo único necesario es modificar las rutas para que las comunicaciones se realicen a través del túnel.

Vamos a poner la primera de las interfaces (que corresponde con enp0s3 en la máquina virtual) en modo bridge para que todo funcione a la perfección. El modo bridge siempre todas las máquinas con el mismo dispositivo de salida.

Apartado 2

Aquí la idea es montar dos redes que se ven con sus rutas. Para ello habrá que indicar en los PC que para ir a la red del otro PC debe hacerlo por el router más cercano.

Los routers, tal como indica en el esquema (Figura) tienen que participar en las dos redes para poder intercambiar paquetes entre ellas.

Los cambios a realizar en las máquinas son los siguientes:

Máquina virtual 1:

```
# ip link set enp0s3 up
# ip address add enp0s3 192.168.201.3/24
# ip route add 192.168.202.0/24 via 192.168.201.1
```

Máquina virtual 2 (R1):

```
# ip link set enp0s3 up
# ip address add enp0s3 192.168.201.1/24
# ip address add enp0s3 192.168.202.2/24
```

Máquina virtual 3:

```
# ip link set enp0s3 up
# ip address add enp0s3 192.168.202.3/24
# ip route add 192.168.201.0/24 via 192.168.202.1
```

Máquina virtual 3 (R2):

```
# ip link set enp0s3 up
# ip address add enp0s3 192.168.202.1/24
# ip address add enp0s3 192.168.201.2/24
```

Todos estos pasos se van a completar con lo que se va a realizar en el apartado 3 pues mientras no se active el reenvío IP en los routers R1 y R2 no será posible que funcionen como router y por tanto no se producirá el enrutamiento.

Apartado 3

En los routers activaremos la función propia de router (el reenvío IP). Por tanto:

Máquina virtual 2 (R1):

```
# sysctl net.ipv4.ip_forward=1
```

Máquina virtual 4 (R2):

```
# sysctl net.ipv4.ip_forward=1
```

Apartado 4

En este apartado el objetivo es establecer el canal de comunicación. Para esto se usará socat para establecer una conexión TCP y en ambos extremos se colocarán dispositivos TUN.

Máquina virtual 2 (R1):

```
# socat TCP-LISTEN:8000 TUN:192.168.203.1/30,up
```

Máquina virtual 4 (R2):

```
# socat TCP-CONNECT:192.168.202.2:8000 TUN:192.168.203.2/30,up
```

Nótese que la conexión se establece entre las IPs (servidor, cliente): (192.168.202.2/R1 y 192.168.202.1/R2). Esto es necesario tenerlo claro para poder elaborar correctamente las rutas en el siguiente apartado.

Apartado 5

En este apartado el objetivo es jugar con las rutas para que la comunicación entre ambas redes se produzca usando el túnel socat establecido en el apartado anterior. Sin embargo, previamente, es necesario hacer rutas específicas para que conexión TCP establecida entre las IPs 192.168.202.1 y 192.168.202.2 siga yendo por el mismo canal.

Máquina virtual 2 (R1):

```
# ip route add 192.168.202.1/24 dev enp0s3
# ip route del 192.168.202.0/24 dev enp0s3
```

```
# ip route add default via 192.168.203.2
```

La primera de las instrucciones garantiza que para ir a la dirección 192.168.202.1 se seguirá usando directamente el dispositivo enp0s3. La segunda es para desactivar el acceso a la red 192.168.202.0 (resto de las IPs excepto 192.168.202.1) a través de la conexión directa. La tercera lo que hace es que la conexión a cualquier equipo de cualquier otra red (incluyendo 192.168.202.0) se haga a través del router 192.168.203.2.

Máquina virtual 4 (R2):

```
# ip route add 192.168.202.2/24 dev enp0s3
# ip route del 192.168.201.0/24 dev enp0s3
# ip route add default via 192.168.203.1
```

Probar todo

El mecanismo más eficaz para probar todo es usar la herramienta traceroute que permitirá, por medio del protocolo ICMP, determinar los saltos que van ejecutando los paquetes. Habrá que ver que los paquetes viajan por el túnel creado viendo si pasan por la red 192.168.203.0/30.

11.2. Poniendo seguridad

Para aportar seguridad sólo hace falta cambiar el SOCAT para que se ejecute con SSL ó mediante redirección de puertos SSH.

En el caso de SSL

Crear los certificados tal como se ha especificado en el guión de la práctica.

Máquina virtual 2 (R1):

```
# socat OPENSSL-LISTEN:8000,cert=server.pem,cafile=myCA.crt,verify=1 TUN:192.168.203.1/30,up
```

Máquina virtual 4 (R2):

```
# socat OPENSSL:192.168.202.2:8000, cert=client.pem,cafile=myCA.crt,verify=1 TUN:192.168.203.2/30,up
```

En el caso de SSH

Máquina virtual 2 (R1):

```
# socat TCP4-LISTEN:8000 TUN:192.168.203.1/30,up
```

Máquina virtual 4 (R2):

```
# ssh -n -F -L 8000:127.0.0.1:8000  user@192.168.202.2
# socat TCP4:127.0.0.1:8000 TUN:192.168.203.2/30,up
```

-n -F son opciones para no ejecutar ningún comando remoto y quedarse en segundo plano mientras se permite la ejecución de comandos adicionales en local.