Puente ethernet con OpenVPN

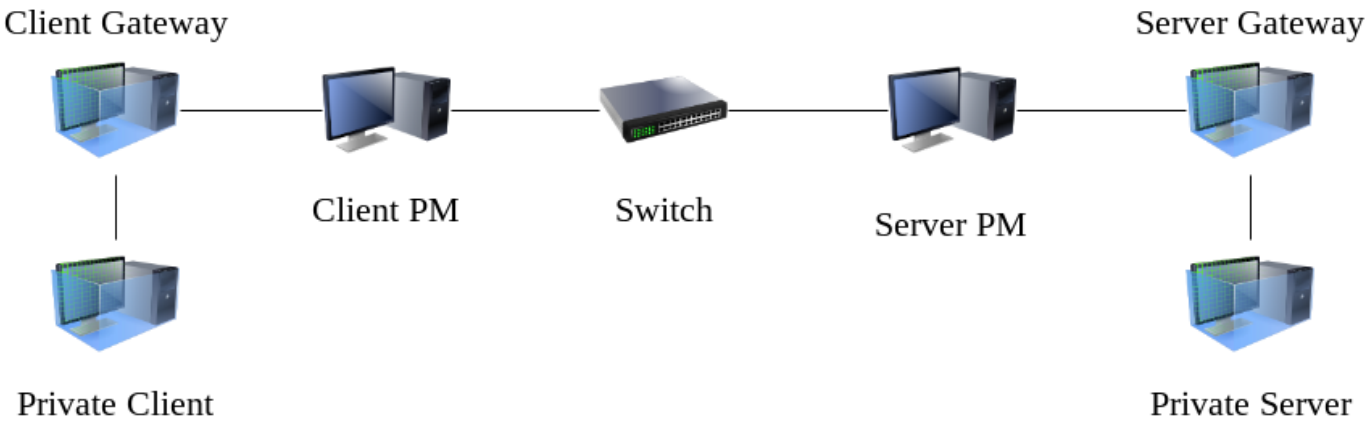
Otras guías detallan la configuración de OpenVPN para un puente 'tun', donde el tráfico es enrutado eficientemente entre un par de sitios alejados geográficamente, pero esta guía trata acerca de los puentes 'tap'. De hecho, describe cómo unirse a un par de sitios que reenvían todo el tráfico ethernet, independientemente del protocolo. Esto es útil para el desarrollo, probar redes y si necesitas reenviar protocolos que no sean IP. Esto es también más simple que los puentes tun porque no tiene que preocuparse por el enrutamiento, solo se necesita ejecutar un único servidor DHCP en uno u otro lado del puente.

Acerca de OpenVPN

OpenVPN es cubierto en este otro artículo, así que haré referencias a las secciones en lugar de copiar y pegar partes de esto aquí. Espero que esto cubra las brechas necesarias para obtener un puente ethernet que funcione.

Introducción

Solo para que quede claro de qué estamos hablando, aquí hay un diagrama:



Hay dos máquinas físicas (cliente PM y servidor PM) y cuatro máquinas virtuales. El objetivo es que el 'Cliente privado' se comunique directamente con el 'Servidor privado', como si estuvieran en el mismo segmento de Ethernet. El conmutador (switch) en el medio es solo mi red domestica pero usted puede considerar que se trata de internet si lo desea. Muchas configuraciones reales harán que 'el servidor de puerta de enlace (Server Gateway)' y el 'Servidor privado' sean lo mismo, no hay razón para que sean distintas, sin embargo, siento que es más fácil seguir lo que está sucediendo si son distintas.

VirtualBox

VirtualBox es mi solución preferida para virtualización por que se instala limpiamente en Slackware

14.2 y -current y es fácil de usar. ¡Si tiene cuatro máquinas físicas en las que puede instalar Slackware, por supuesto puede hacer todo esto sin virtualización! En mi caso yo tengo dos máquinas físicas que representan cada 'sitio' que quieres unirte. Todas las máquinas virtuales corren Slackware de 64 bit versión 14.2 sin parchar.

Servidor privado

Sin más preámbulos, sigamos configurando nuestro servidor privado. Le he dado a la máquina virtual los siguientes recursos:

- 1024 MB RAM,
- 8GB de disco duro.

En la configuración de red de Virtualbox, le he dicho a la máquina virtual que use 'Red interna' y escribí el nombre de la red 'slacknet' por falta de un nombre mejor. Si usted quiere podrías llamarlo 'privado'. Nada alcanzará esta red a menos que configuremos una máquina como enrutador.

Yo instalé Slackware con el conjunto de discos A, AP, N y L por que no necesito ninguna GUI pero es necesario las herramientas de red, pero esto realmente no importa, si querés podes hacer una instalación completa. Configure esta máquina con una dirección IP estática, en este caso 10.0.0.1. De otra forma, la instalación sigue las opciones por defecto.

Luego del arranque, solo hay una cosa que debemos hacer para que esta prueba sea realista. Es necesario habilitar Dnsmasq. Puede seguir ese enlace para obtener más detalles, pero solo necesitamos un pequeño archivo de configuración /etc/dnsmasq.conf que contiene solo:

```
interface=eth0
dhcp-range=10.0.0.50,10.0.0.70,12h
```

¡Eso es! Primero haga una copia de respaldo del archivo antiguo dnsmasq.conf en algún lugar si está preocupado por perder todos los comentarios, pero no importa realmente, ya que es una máquina de pruebas desechable. Lo anterior es todo lo que necesita para trabajar con el servidor dhcp. Eso es lo que hace tan grande a Dnsmasq.

Ahora solo necesitamos iniciar el servidor:

```
# chmod 755 /etc/rc.d/rc.dnsmasq
# /etc/rc.d/rc.dnsmasq start
```

¡Entonces ahora tenemos nuestro servidor privado corriendo sobre tu red privada! Por el momento, nada puede verlo, pero si instalara otra máquina virtual y la conectara a esa red privada (slacknet), obtendría una dirección IP. Hemos hecho esto simplemente para demostrar que el tráfico que no es IP (es decir, DHCP) puede atravesar nuestro puente y DHCP es solo una forma conveniente de hacerlo.

Server Gateway

Configuración de la máquina virtual en VirtualBox

El servidor de puerta de enlace es otra máquina Slackware cuya configuración de memoria es la misma, disco duro y conjuntos de discos como el 'Servidor privado' anterior. Se diferencia en que tiene un segundo NIC configurado en VirtualBox.

El primer NIC en VirtualBox también conocido como 'Adaptador 1' en la interfaz gráfica (GUI) de VirtualBox terminará siendo eth0 en la máquina virtual Slackware iniciada, solo tenga en cuenta esto. Yo configuraré esto como 'Adaptador puenteado' en 'Adjunto a:' en la GUI. Note que para todas las NICs yo estoy configurando 'modo promiscuo' a 'permitir todo', aunque no tengo idea si eso es necesario para todas las NICs en todas las máquinas de esta guía.

El 'Adaptador 2' se conectará a la red privada 'slacknet' que se describe en la configuración de 'Servidor privado' más arriba.

Configuración de red

Ahora todas las preguntas que Slackware le hace durante la configuración de la red se aplican a la interfaz pública que está conectada directamente (a través del puente de VirtualBox) al conmutador (switch) en el centro del diagrama anterior. Lo más probable es que sea el enrutador de banda ancha de su hogar el que proporcionará una dirección IP y tal vez (si tiene suerte) un servidor de nombres (DNS) apropiado. No importa, siempre que haya algo que pueda usar para identificar está interfaz pública en la red. De aquí en adelante, me referiré a la interfaz ethernet de está maquina como 'vpn', o podemos decir 'vpn.localnet', aunque, si solo tiene una dirección IP, solo use esa.

Las buenas noticias es que no importa si esta 'choca' con el rango de direcciones IP de las que hablamos en la configuración de dnsmasq de el 'servidor privado' por que... bueno... esto es realmente privado. ¿Limpio eh? Obviamente, esto sería un gran problema si queremos agregar más adelante el enrutamiento del 'Servidor privado' al mundo exterior, pero ese será otro ejemplo.

dispositivo tap0

Habiendo instalado una segunda máquina virtual Slackware con dos tarjetas ethernet, solo una de las cuales está activa, necesitamos hacer algunas cosas adicionales antes de hablar sobre OpenVPN.

El primer paso es instalar tunctl. Voy a hacer la suposición (posiblemente injusta) de que sabes cómo instalar paquetes de Slackware desde slackbuilds.

Una vez instalado tunctl, edita el archivo /etc/rc.d/rc.inet1, busca las siguientes líneas:

```
# Funciones para iniciar la red:
start() {
  lo_up
...
```

Haz que digan:

```
# Funciones para iniciar la red:
start() {
  lo_up
  /usr/sbin/tunctl -u nobody -t tap0
...
```

Esto creará (en el arranque) un dispositivo tap0 con los permisos correctos de modo que OpenVPN pueda acceder a él y, más concretamente, lo creará antes de que comience la configuración de puentes en los scripts de inicio de Slackware. También se podría crear en rc.local pero se iniciaría demasiado tarde.

configuración del puente

Ok, ahora estamos hablando acerca de kernel network bridging, no del puente ethernet que es el tema de este artículo.

Este es un hecho triste que a pesar de que tenemos una NIC asignada a propósito a la tarea de hablar con nuestra red privada, es decir, no se comparte con ningún otro tráfico, todavía necesitamos crear un puente del kernel para que OpenVPN pueda hablar correctamente con la NIC. En otras palabras, no podemos decirle a OpenVPN que simplemente 'use eth1' (o que sea tan simple). Esto requiere un dispositivo 'tap'. Y un tap puede solo hablar a un dispositivo NIC a través de un puente. Así que echemos un vistazo a la configuración de red en Slackware.

Para el dispositivo eth0 en /etc/rc.d/rc.inet1.conf ahora tendrá una configuración como esta, suministrado como cortesía por el script netconfig de Slackware:

```
IPADDR[0]=""
NETMASK[0]=""
USE_DHCP[0]="yes"
...
```

O puede tener una dirección IP estática. Debes tener *algo*. Ahora desplácese hacia abajo a la sección sobre puentes:

```
# Ejemplo de como configurar un puente:
# Note el agregado de la variable "BRNICS" la cual contiene un lista
separada por espacios
# de las interfaces de red físicas que tu quieres agregar a el puente.
...
```

Justo debajo se desea que aparezca lo siguiente:

```
IFNAME[1]="br0"
BRNICS[1]="eth1 tap0"
IPADDR[1]="0.0.0.0"
```

Se cuidadoso con los números. Los corchetes son los que le dicen a Slackware que este es el segundo dispositivo que configuramos como puente, y nosotros queremos agregar el dispositivo eth1 a el puente junto con el dispositivo tap0. Slackware no configura el dispositivo eth1, ni siquiera necesitamos mencionarlo o referirlo de otra manera en los archivos de configuración como esclavo del puente.

Algunas personas pueden configurar aquí br0 como br1, para indicar que esta conectado al segundo NIC y mantenerlos igual. Usted puede hacer esto. También puedes cambiar el nombre de tap0 si tu lo deseas, la experiencia me dice que es mejor usar siempre números bajos para todo independientemente de lo que hagan. Para mí, al menos, reduce la confusión.

No necesitas configurar la dirección IP a 0.0.0.0, pero esto es una forma práctica de hacer que el puente sea levantado sin darle una dirección ip, y no es un problema. Si usted no configura alguna dirección ip, usted puede hacer 'ifconfig br0 up' posteriormente.

Cuando se arranca con esta configuración, usted debé chequear que br0 esta arriba con:

```
# ifconfig br0
```

y usted puede también chequear el puente con 'brctl show br0' y se debería mostrar algo como esto:

```
# brctl show br0
bridge name bridge id        STP enabled    interfaces
br0     8000.485b39c042ee    no        eth0
                                                 tap0
```

Con esto ha finalizado la configuración del puente. Ahora OpenVPN (cuando este eventualmente configurado) podrá comunicarse con la red privada, en este ejemplo, 'slacknet' sin problemas, incluso si solo tiene 'nadie' como nombre de usuario.

Configuración de OpenVPN

Así que esta es la parte en la que puedo descansar tranquilamente 'sobre los hombros de gigantes' por así decirlo. Por favor, configure su PKI para un servidor OpenVPN siguiendo guía en los howto de OpenVPN. Siga toda la sección 5 para tener un conjunto de claves públicas y privadas para el servidor. No te preocupes por crear el archivo de configuración para hacer referencias a ellos, ya que lo haremos aquí.

¿Estas preparado? bien.

Mi /etc/openvpn/server.conf es un archivo pequeño y un poco más simple que el howto de OpenVPN. Se han eliminado los comentario por brevedad y también la parte de enrutamiento. Usted puede siempre chequear las opciones en la pagina del manual de OpenVPN si lo desea, pero la mayoría se explican por si misma.

```
port 443
proto udp      # o tcp
dev tap0     # o tun
ca /etc/openvpn/certs/ca.crt
cert /etc/openvpn/certs/vpn.crt
```

```
key /etc/openvpn/keys/vpn.key
dh /etc/openvpn/certs/dh2048.pem
tls-server
keepalive 10 120
daemon
log-append  /var/log/openvpn.log
verb 3
```

Yo removí la palabra clave para vpn.key (recuerda que es equivalente de 'server1.key' en el howto de OpenVPN). Eh evitado renunciar a root. Usted ejecutaría una configuración más segura si este fuera un sistema de producción al comprender las opciones en la página de manual de OpenVPN, esto es solo para que pueda trabajar.

Ahora usted puede volver a atrás a los otros artículos y leer Configuración del servidor OpenVPN. Ahora usted puede saltar a los scripts que hablan son rc.openvpn, para asegurarte de tener un script que inicie OpenVPN.

Client Gateway

Configuración de VirtualBox

Al igual que la máquina 'puerta de enlace privado' ('Private Gateway'), esta máquina puede tener dos NICs. 'Adaptador 1', sin embargo, puede ser NAT en lugar de puenteado, ya que nada tiene que ser capaz de encontrarlo. El 'Adaptador 2' puede estar conectado a la red privada 'slacknet'. Yo hice le asigne el mismo nombre que en la otra máquina del servidor, ya que se conectarán entre sí, pero no importa como se llame, siempre que el 'Cliente Privado' use el mismo nombre.

Instalación de Slackware

Me temo que una vez más, necesitarás instalar tunctl, crear el dispositivo tap0 y habilitar el puente exactamente como sucedió con la puerta de enlace del servidor. Una vez que tap0, y br0 son creadas en el arranque usted puede configurar OpenVPN.

Configuración de OpenVPN

Una vez más, pongo en juego el otro howto para la configuración PKI del lado del cliente. Sin embargo, usted puede saltar la creación del archivo rc.openvpn ya que no estaremos ejecutando OpenVPN como un demonio. El archivo de configuración del cliente es un simple archivo(/etc/openvpn/client.conf):

```
client
dev tap0
proto udp
remote vpn.localnet 443
```

```
ca /etc/openvpn/certs/ca.crt
cert /etc/openvpn/certs/client.crt
key /etc/openvpn/keys/client.key
dh /etc/openvpn/certs/dh2048.pem
remote-cert-tls server
verb 3
```

Ahora debería poder conectarse a la 'Server Gateway' con este comando como root:

```
openvpn /etc/openvpn/client.conf
```

Cliente Privado

Al igual que el 'Servidor privado', esta máquina tendrá una configuración similar de VirtualBox. Esta tendrá un solo NIC conectado a una red llamada 'slacknet'.

Iba a usar Slackware para esto, pero decidí hacer trampa y usar Slax (distribución de CD de inicio) para ahorrar tiempo en la instalación. Slax solicita una dirección IP mediante DHCP, por lo que realmente tiene todo lo que necesita para probar el puente (solicitud DHCP enviada, respuesta DHCP recibida). Si en el arranque va todo bien, obtendrás una dirección IP como 10.0.0.50 (o al menos en el rango de direcciones proporcionadas por el servidor Dnsmasq que configuramos en el 'Servidor privado').

Pruebas de extremo a extremo

Entonces, una vez que las cuatro máquinas virtuales (¡sí!) estén en funcionamiento y corriendo, debería ser posible (si todo funciona) iniciar el Cliente privado y obtener una dirección IP en el rango correcto, es decir, entre 10.0.0.50 y 10.0.0.70. Preste mucha atención a /var/logs/openvpn.log sobre la 'Server Gateway' y, por supuesto, los mensajes mostrados cuando se ejecuta OpenVPN en la 'Client Gateway', ya que deberían informarle lo que está mal. OpenVPN es bastante buena. El paranoico también querrá verificar /var/state/dnsmasq/dnsmasq.leases en el 'Servidor privado' para asegurarse de que todo está bien y que en realidad estamos hablando a través del puente.

Consolidación

Como explique en la introducción, es posible probar todo esto sin el 'Servidor privado', y combina la función de 'Servidor privado' y 'Servidor de puerta de enlace' en una simple máquina. Sigue leyendo para descubrir cómo.

Lo primero que debe hacer es apagar el 'servidor privado'.

A continuación, debemos asignar una dirección IP a nuestro puente Ethernet en el 'Servidor de puerta de enlace'.

```
# ifconfig br0 10.0.0.1 netmask 255.255.255.0
```

¡Asignamos una dirección IP a el puente y no a eth1!

Ahora solamente necesitamos correr una instancia de dnsmasq sobre la interfaz br0, para servir solicitudes a la VPN. Como la configuración del 'Servidor privado', /etc/dnsmasq.conf debería verse así:

```
interface=br0
dhcp-range=10.0.0.50,10.0.0.70,12h
```

Entonces es solo cuestión de ejecutar dnsmasq:

```
# chmod 755 /etc/rc.d/rc.dnsmasq
# /etc/rc.d/rc.dnsmasq start
```

En este punto, rebooteando el 'Servidor privado' debería obtener una dirección IP del servidor consolidado, y eso es todo. Sin embargo, también cambié el **old** 'Servidor privado' a dhcp para confirmar que también podría obtener una dirección IP desde 'Server Gateway'.

Fuentes

- Escrito originalmente por User bifferos
- Traducido por — *rramp 2019/02/18 01:03 (UTC)*

howtos, ethernet, bridging, author bifferos