

# Interfaces de túnel

Si está buscando establecer un túnel SSH entre dos redes y tratar el túnel como una interfaz, esto puede ayudar.

\* Primero recomendaría habilitar `rc.ip_forward` en `/etc/rc.d/` (en máquinas locales y de destino)

```
root@darkstar:~# chmod +x /etc/rc.d/rc.ip_forward
```

o para un uso temporal (se pierde al reiniciar)

```
root@darkstar:~# echo "1" > /proc/sys/net/ipv4/ip_forward
```

- Casi se me olvida que tendrá que habilitar el túnel en `/etc/ssh/sshd_conf`. Encuentre esta parte `# PermitTunnel` no descomentar y cambie a sí (en la máquina de destino) puede usar `vi` para editar o probar el siguiente código:

```
root@darkstar:~# /usr/bin/sudo /bin/sed -e "s/#PermitTunnel\
no/PermitTunnel\ yes\ #changed\ `date '+%Y%m%d'` \
`\ by\ `/bin/whoami`/" -i.stock_slackware-`/bin/awk '{print $2}'
/etc/slackware-version` /etc/ssh/sshd_config
```

- A continuación, estoy usando `autossh` que se encuentra en [slackbuilds.org](http://slackbuilds.org). El siguiente script lo requiere (hay un método alternativo mencionado en la parte inferior. Reemplace la línea 20).
- También querrá tener configuradas conexiones ssh no interactivas. [Cómo utilizar las claves SSH para conectarse sin una contraseña.](#)
- También asumiendo que tiene privilegios de `sudo` y usuario en la máquina remota (usando la opción `NOPASSWD: ALL` en `/etc/sudoers` )  
Ejemplo: como tipo de raíz " `visudo` " y agregue a tu usuario le gusta esto

```
rich ALL=(ALL) NOPASSWD: ALL
```

- A continuación estoy usando un simple script de bash que explicaré después de echarle un vistazo.

## [rc.tunnel](#)

```
#!/bin/bash

#####
#
#Enter the ip of the target you wish to make a tunnel with.
#By ip address or hostname
#
target=74.79.121.210
port=22
#
#####
# suggestions contact rich at lehcar.no-ip.org
# with thanks to Billy T (for idea and assistance)
```

```
#####  
#load module  
/usr/bin/sudo /sbin/modprobe tun  
#load remote module  
/usr/bin/ssh -p $port $target "/usr/bin/sudo /sbin/modprobe tun"  
/bin/sleep 1  
  
/usr/bin/sudo /usr/bin/autossh -p $port -M 0 -o  
"ServerAliveInterval 60" -o "ServerAliveCountMax 3" -fw 0:0  
$target /bin/true  
/bin/sleep 4  
/usr/bin/ssh -p $port $target "/usr/bin/sudo /sbin/ifconfig tun0  
192.168.5.2 pointopoint 192.168.5.1 netmask 255.255.255.252  
broadcast 192.168.5.3"  
/usr/bin/sudo /sbin/ifconfig tun0 192.168.5.1 pointopoint  
192.168.5.2 netmask 255.255.255.252 broadcast 192.168.5.3  
/usr/bin/sudo /usr/sbin/iptables -t nat -A POSTROUTING -o eth0 -j  
MASQUERADE  
/usr/bin/sudo /usr/sbin/iptables -A FORWARD -i eth0 -o tun0 -m  
state --state RELATED,ESTABLISHED -j ACCEPT  
/usr/bin/sudo /usr/sbin/iptables -A FORWARD -i tun0 -o eth0 -j  
ACCEPT  
  
/usr/bin/ssh -p $port $target "/usr/bin/sudo /usr/sbin/iptables -t  
nat -A POSTROUTING -o eth0 -j MASQUERADE"  
/usr/bin/ssh -p $port $target "/usr/bin/sudo /usr/sbin/iptables -A  
FORWARD -i eth0 -o tun0 -m state --state RELATED,ESTABLISHED -j  
ACCEPT"  
/usr/bin/ssh -p $port $target "/usr/bin/sudo /usr/sbin/iptables -A  
FORWARD -i tun0 -o eth0 -j ACCEPT"
```

- Para dirigir el tráfico sobre el túnel intente:

```
user@darkstar:~$ sudo /sbin/route add -net 74.125.131.0 netmask  
255.255.255.0 dev tun0
```

- Para sustituir ssh normal en lugar de autossh, sustituya esta línea a continuación por una con autossh (línea 20)

```
/usr/bin/sudo /usr/bin/ssh -p $port -o "ServerAliveInterval 60" -o  
"ServerAliveCountMax 3" -fw 0:0 $target /bin/true
```

Para recapitular algunas de las opciones: " -fw 0: 0 " coloca el proceso ssh en segundo plano/abre un túnel y " 0: 0 " selecciona las interfaces locales y remotas (es decir, tun0 y tun0 ). Las opciones " -o " se utilizan para especificar parámetros para openSSH.

## Resultados

Ahora puede acceder a la computadora remota usando " 192.168.5.2 " y sus comunicaciones pasarán por el túnel ssh. En el script anterior, es posible que desee modificar la red y la máscara para que se ajusten a sus necesidades. Elegí " 192.168.5 " pero su configuración probablemente será diferente.

No podré elaborar las reglas de iptables todavía un poco verdes allí, pero funcionan.

## Fuentes

- Escrito originalmente por [ricky\\_cardo](#)
- Traducido por: [Victor](#) 2019/02/14 18:47 (UTC)

[howtos](#), [tunnel](#), [tun0](#), [author ricky cardo](#)

From:  
<https://docs.slackware.com/> - **SlackDocs**

Permanent link:  
[https://docs.slackware.com/es/howtos:network\\_services:tunnel\\_interfaces](https://docs.slackware.com/es/howtos:network_services:tunnel_interfaces)

Last update: **2019/02/14 18:49 (UTC)**

