

ssh tunel inverso (o reverso)

método SSH

sean:

- A - ordenador al que me quiero conectar desde «donde sea»
- B - ordenador con acceso «pleno»
- C - cualquier ordenador
- A y C pueden estar sin acceso público SSH

el método sería:

1. conecto desde A a B con:

```
ssh -R 12345:localhost:22 usuario_B@B
```

2. conecto desde C a B con SSH (como siempre)

```
$ ssh usuario_B@B
```

3. una vez en B, conecto con A con:

```
ssh -p 12345 usuario_A@localhost
```

- desde A establezco una conexión SSH haciendo que B escuche en 12345 y se lo envíe a A
- desde C conecto a B en primera instancia y desde ahí conecto con A usando la conexión ya abierta
- puedo usar `nohup ssh -N -f -R 12345:localhost:22 usuario_B@B` para que quede la conexión «activa» aunque haga logout
- este método tiene la ventaja que si escanean con `nmap -p 12340-12350 -sV @B` no hay puertos en uso

/via: [acceso SSH a ordenador tras Firewall desde un segundo](#)

método PORT_FORWARDING

sean:

- A - ordenador con página web tras firewall (en este caso)
- B - ordenador en internet
- C - ordenador que quiere acceder a A

método:

- activar: `# vim /etc/ssh/sshd_config`

```
GatewayPorts clientspecified
```

- Reiniciar el servicio SSH:

```
$ service ssh reload
```

- Luego desde el A ejecutar el siguiente comando:

```
ssh -N -f -R B:6677:localhost:80 user@B
```

- Y una vez hecho esto ya estaría el túnel creado. Para conectarse desde el C habría que escribir lo siguiente en el navegador:

```
http://172.10.5.36:6677
```

/via: [Usar SSH para dar acceso a ordenador:puerto en red privada a través de un servidor externo sin comprometer contraseñas](#)

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/linux:ssh:reverse?rev=1478974331>

Last update: **12/11/2016 10:12**

