Configuración avanzada de redes en GNU/Linux

Adrian Boubeta <root@boube.es>

Introducción

- La libreta
- Parar network-manager / wicd
- Crear un usuario minino con clave minino
- Instalar vim
- Importante: ir practicando los comandos a medida que los vamos viendo.

Redes Inalámbricas

- iwconfig
  - iwconfig
  - iwlist ath0 scan
  - iwconfig ath0 essid $RED key off
  - iwconfig ath0 essid $RED key s:$CLAVE_ASCII
  - iwconfig ath0 essid $RED key $CLAVE_HEXA

Ifconfig

- ifconfig
  - ifconfig
  - ifconfig ath0
  - ifconfig ath0 $ip
  - ifconfig ath0 $ip netmask $netmask
  - route add default gw $gw
  - dhclient $interrfaz

Ifup / ifdown I

- /etc/network/interfaces

```
iface ath0 inet static
    address $IP
    netmask $NETMASK
    gateway $GW
    wpa-ssid $RED | wireless-essid $RED
    wpa-psk $CLAVE | wireless-key $CLAVE
```

Ifup / ifdown II

```
iface $INTERFAZ inet dhcp
    wpa-ssid $RED | wireless-essid $RED
    wpa-psk $CLAVE | wireless-key $CLAVE
```

Ifup / ifdown III

- Ifup
- ifdown

hosts / hostname

- /etc/hosts
- /etc/hostname
- hostname

Diagnóstico

- ping
- route -n
- netcat
- netstat -punta
- traceroute
- mtr
- nmap

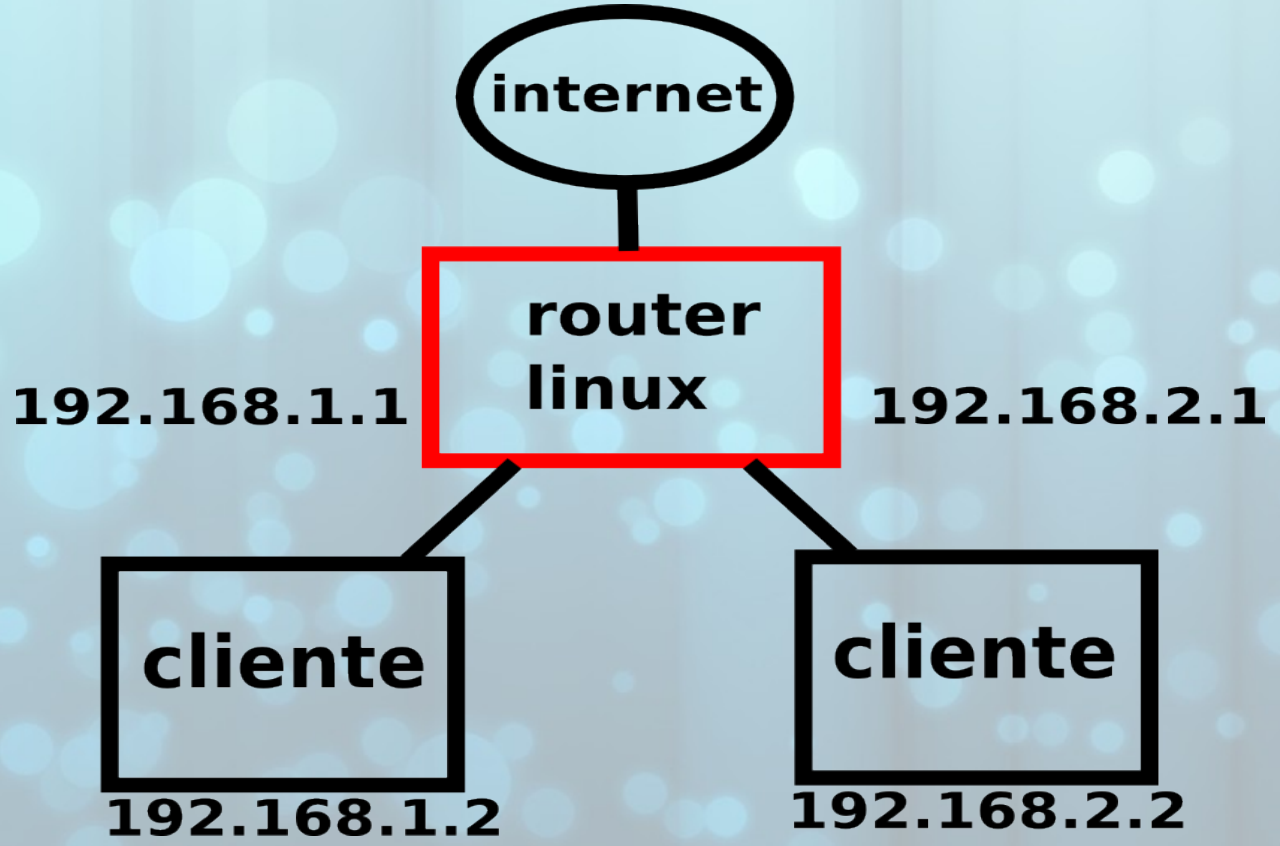
NMAP

- nmap $IP (Puertos abiertos)
- nmap -O $IP (Puertos+Sistema Operativo)
- nmap -sP $RANGO.* (Maquinas encendidas)

Routing

echo 1 > /proc/sys/net/ipv4/ip_forward

/etc/sysctl.conf: net.ipv4.ip_forward=1



Iptables

- Sólo nivel local
  - 2 tablas: INPUT, OUTPUT
  - 4 acciones: ACCEPT, DROP, REJECT, LOG

- Definir politicas (-P)

```
iptables -P OUTPUT ACCEPT
iptables -P INPUT DROP
```

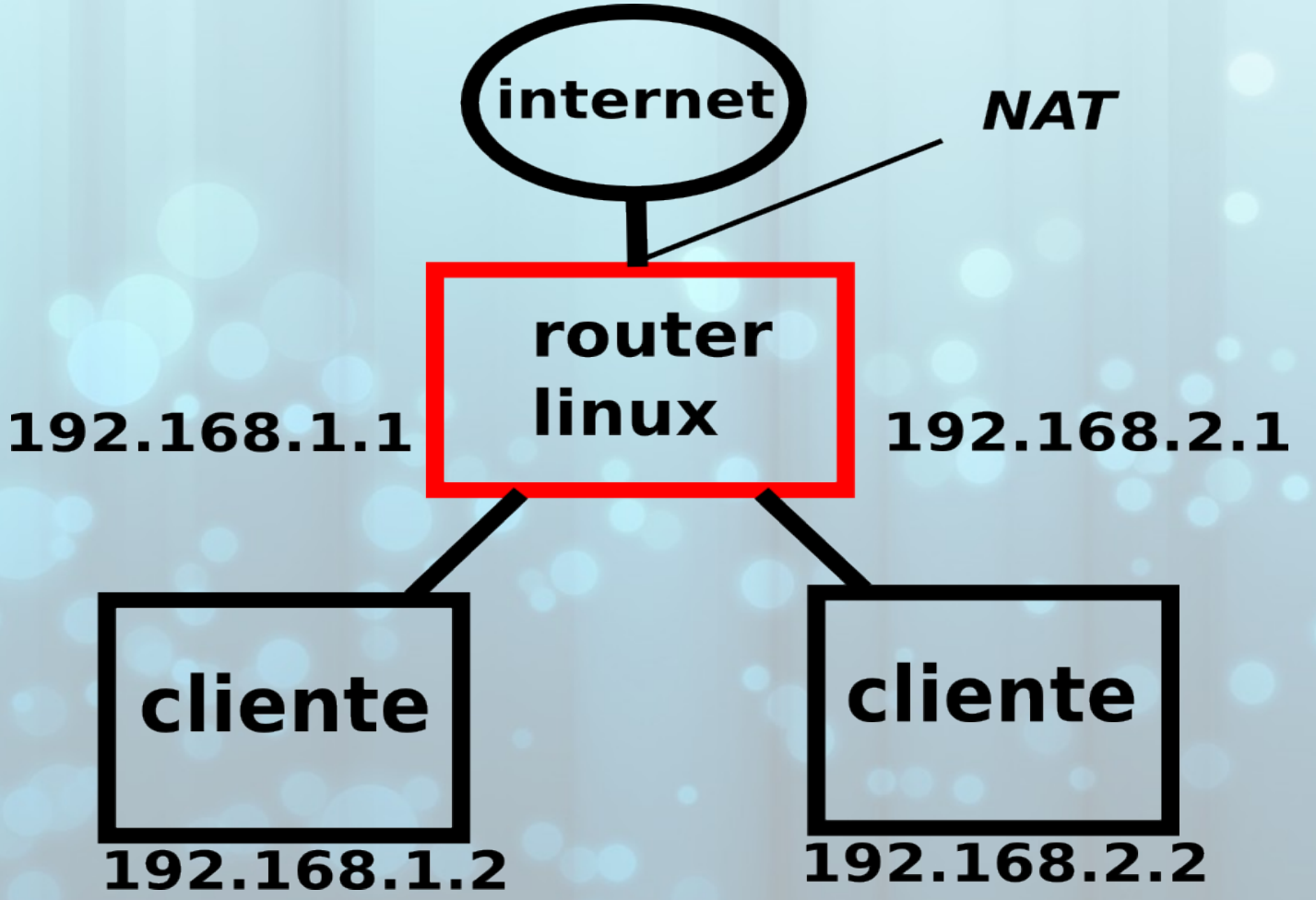
Iptables

Ejemplos de reglas:

```
iptables -A INPUT -p tcp -d 10.1.1.1 -j ACCEPT
iptables -A INPUT -p tcp --sport 80  -j ACCEPT
```

NAT

- Network Address Translation



NAT

```
iptables -t nat -A POSTROUTING -s \
192.168.1.0/24  -d 0.0.0.0/0 -j MASQUERADE
```

Iptables

```
#!/bin/bash
echo  -n "Arrancando el firewall ... "
# FLUSH de reglas
 iptables -F
 iptables -t nat -F
 # Establecemos politica por defecto
 iptables -P INPUT DROP
 iptables -P OUTPUT ACCEPT
 # Permitimos ya establecidas y relacionadas
 iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
```

Iptables

```
 # Del localhost   aceptamos todo
 iptables -A INPUT -i lo -j ACCEPT
## TCP INPUT
 iptables -A INPUT  -p tcp --dport 22  -j ACCEPT
 iptables -A INPUT  -p tcp --dport 25  -j ACCEPT
 iptables -A INPUT  -p tcp --dport 80  -j ACCEPT
## UDP INPUT
 iptables -A INPUT -p udp --dport 53  -j ACCEPT
echo ". . OK"
```

Bridges

- El bridge convierte dos tarjetas de red conectadas a un GNU/Linux en un switch
- apt-get install bridge-utils
- brctl addbr br0
- ifconfig eth0 up
- ifconfig ath0 up
- brctl addif br0 ath0
- brctl addif br0 eth0
- dhclient br0 / ifconfig br0 $IP ; route add default gw $GW

SSH

ssh $usuario@$maquina

ssh $usuario@$maquina “$comando”

ssh -X $usuario@$maquina

scp $usuario@$maquina:/ruta/fichero /ruta/dest

scp -r $usuario@$maquina:/ruta/direc /ruta/dest

SSH - Eliminar fingerprints

- ssh minino@localhost
- ssh-keygen -q -f /etc/ssh_host_rsa_key -N -t rsa
- ssh-keygen -q -f /etc/ssh_host_dsa_key -N -t dsa
- sed -i 1d .ssh/know_hosts

SSH – Asegurando sshd

- /etc/ssh/sshd_config:
  - Port $puerto
  - PermitRootLogin no
  - AllowUsers/AllowGroups $usuario/$grupo
  - PasswordAuthentication yes/no
  - PubKeyAuthentication yes/no
  - X11Forwarding yes/no

SSH - .ssh/config

- $HOME/.ssh/config:

  Host $nombre_arbitrario

    HostName $nombre_de_dominio / $IP

    Port $puerto

    User $usuario

SSH - Keys

- ssh-keygen
  - Keys sin clave
  - Keys con clave
- ssh-copy-id -i .ssh/id_rsa.pub usuario@maquina
- La clave de usuario no viaja por la red, ni siquiera cifrada.

SSH - Parallel

apt-get install pssh

ssh-agent bash

ssh-add

parallel-ssh -h $LISTA -o $SALIDA $COMANDO

parallel-ssh -h lista.txt -o /tmp/out/ “cat /proc/cpuinfo”

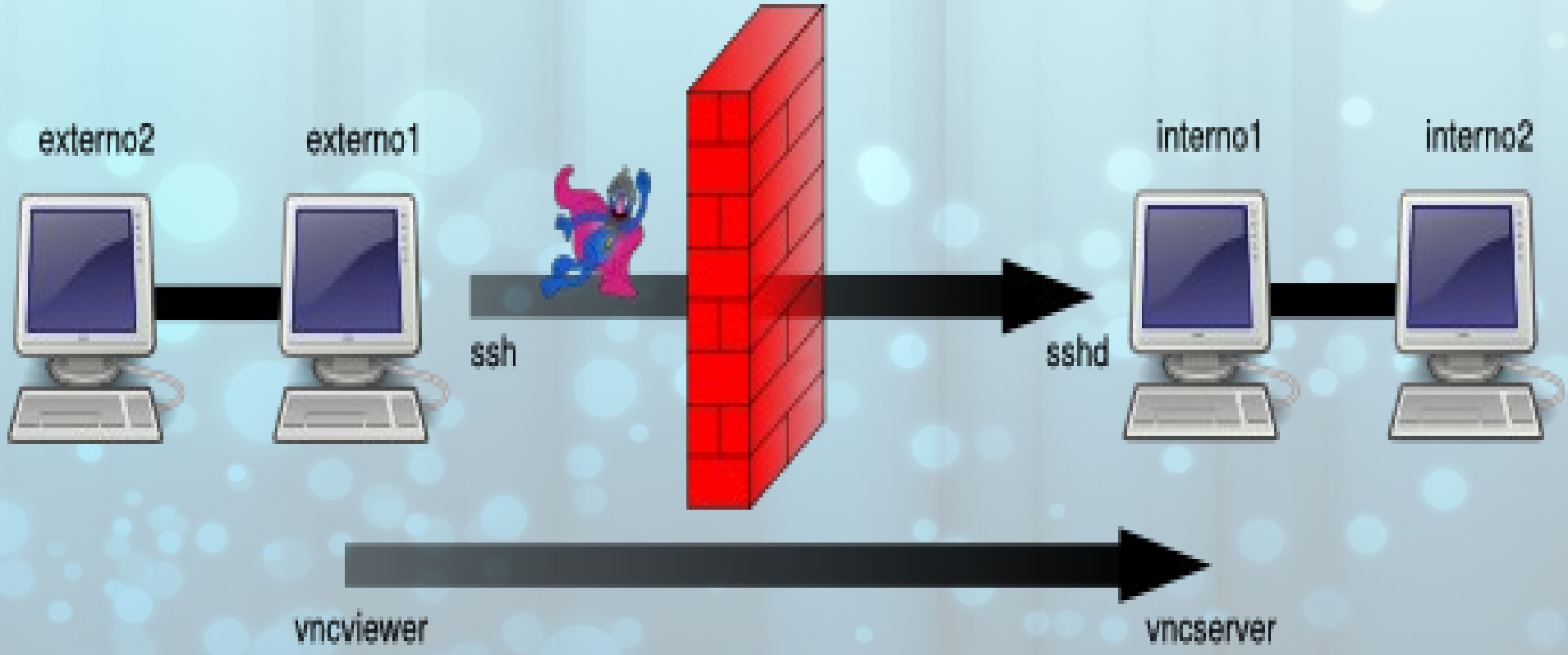
SSH - sshfs

apt-get install sshfs fuse-utils && modprobe fuse

sshfs $IP:directorio_remoto directorio_local -o allow_other

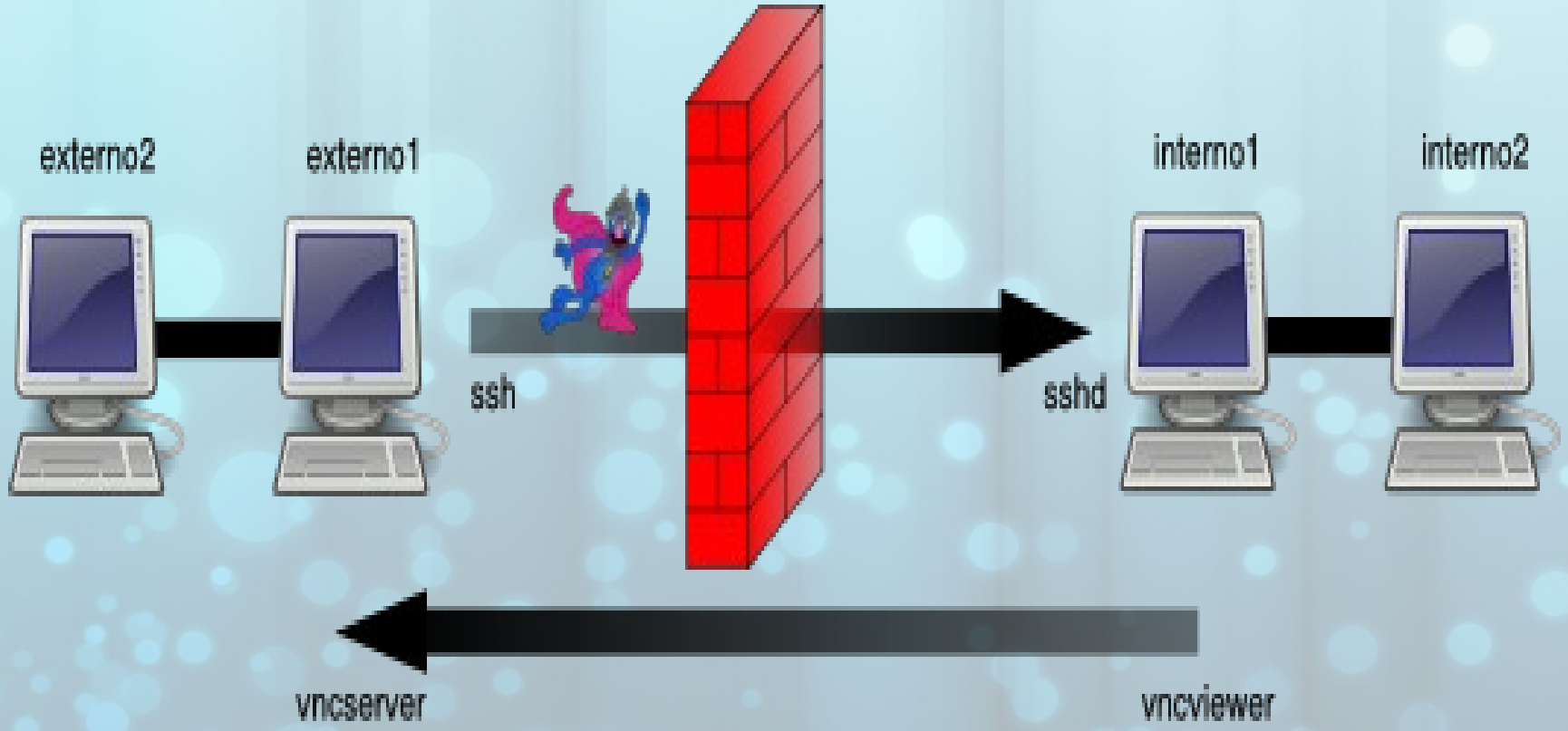
Tunel Local

ssh -L 7900:localhost:5900 usuario@interno1



Tunel Local Inverso

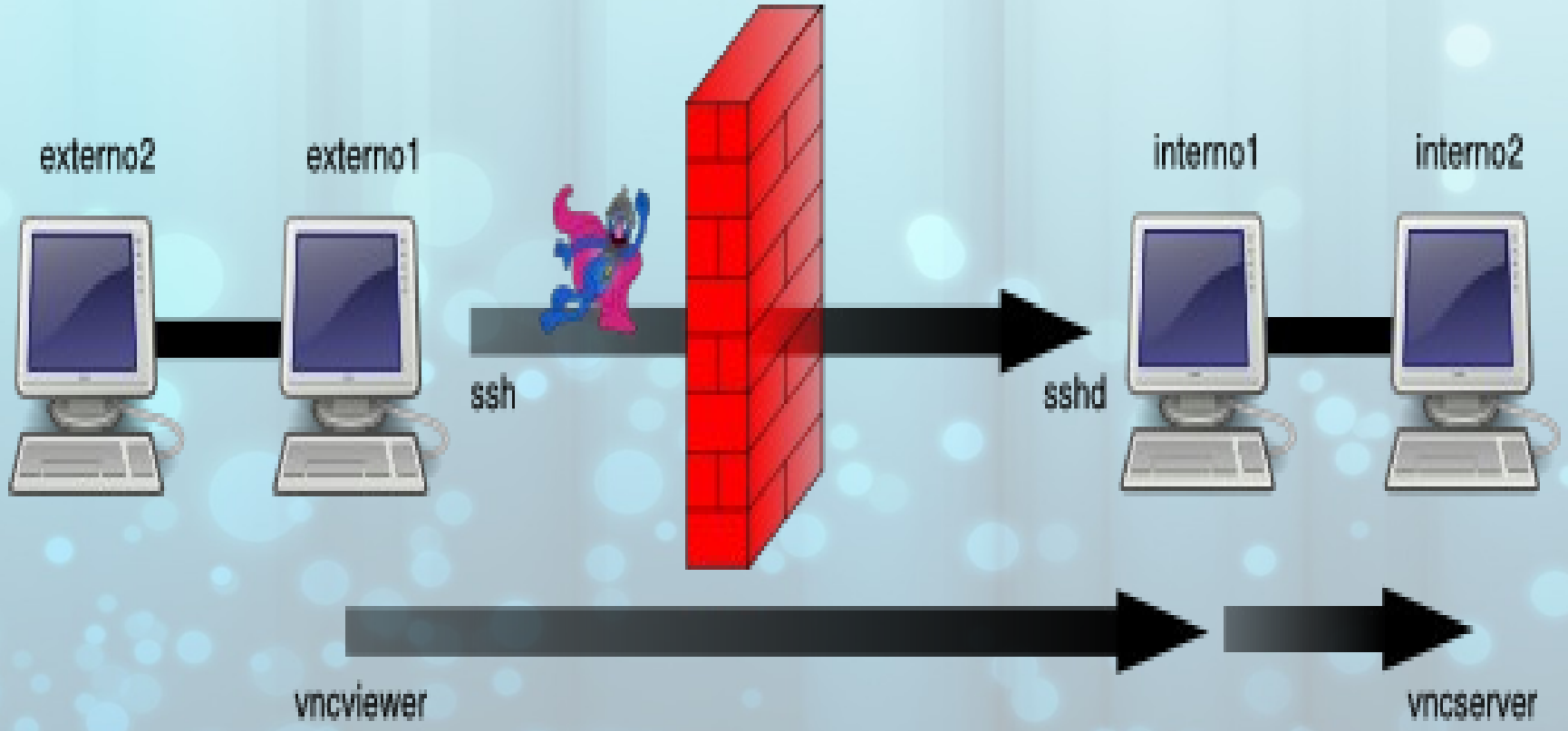
- ssh -R 7900:localhost:5900 usuario@interno1



vicente-navarro.com

Tunel Remoto

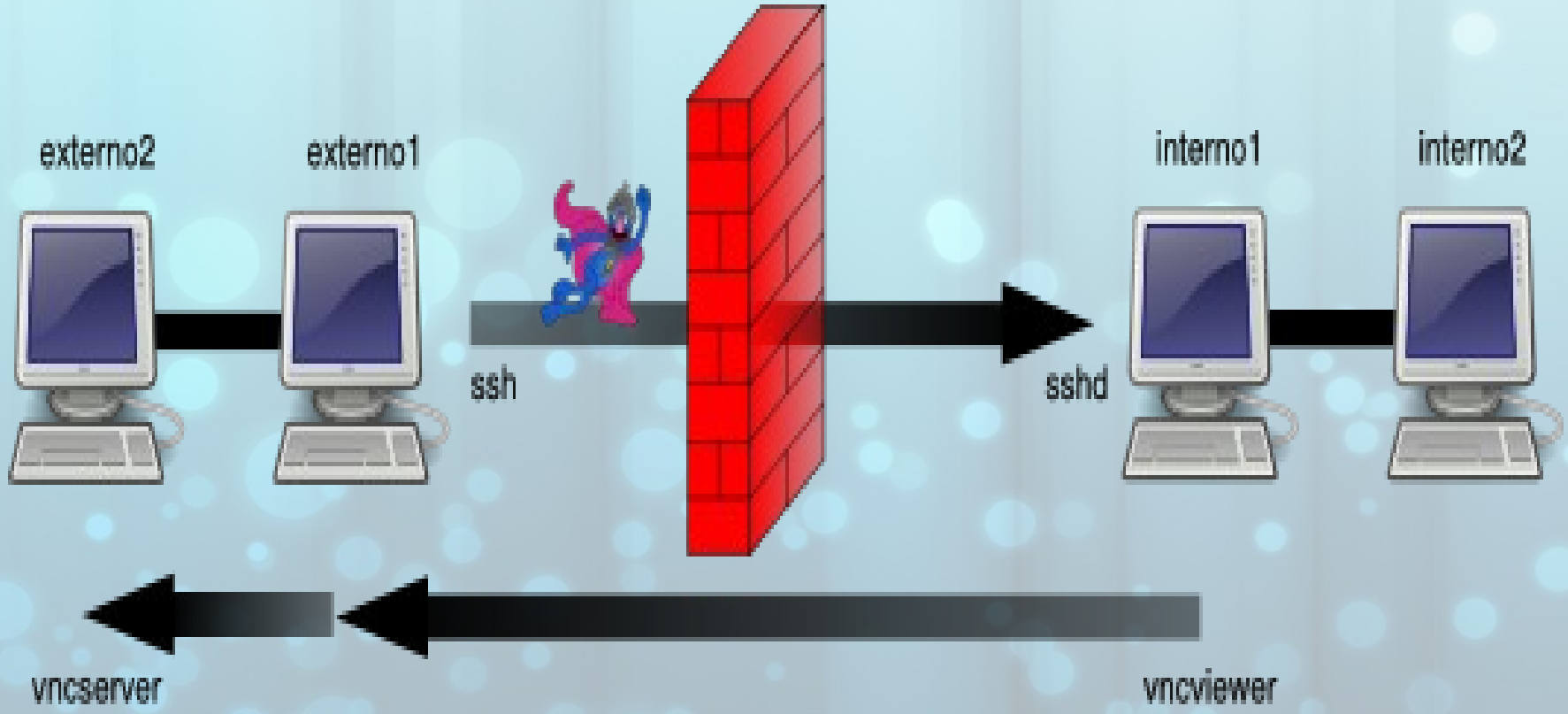
- ssh -L 7900:interno2:5900 usuario@interno1



vicente-navarro.com

Tunel Remoto Inverso

- ssh -R 7900:externo2:5900 usuario@interno1



Socks

- **apt-get install iceweasel**
- ssh -D 8080 usuario@maquina
- **Configurar iceweasel solo con proxy socks**

Neatx

- apt-get install python-software-properties
- deb http://ppa.launchpad.net/freenx-team/ppa/ubuntu lucid main
- apt-get update
- apt-get install neatx-server
- http://www.nomachine.com/download-client-linux.php

Gracias a Miguel Bouzada, Guillermo Cordeiro, Vicente Navarro,
Oscar Casal, TEGNIX, . . .