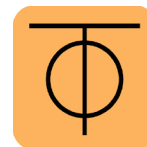


# Centro de Procesamiento de Datos



## Práctica 5. Acceso remoto

### Objetivo:

En esta práctica estudiaremos diversas formas de comunicaciones entre nodos. En particular, el uso de ZeroTier y túneles SSH.

Debe presentarse en un documento pdf, con la siguiente información:

- Captura del escritorio VNC creado (Apartado III).
- Captura de ping a máquinas remotas (Apartado IV)
- Acceso SSH a la máquina del compañero mediante ssh (modo terminal) (Apartado V)
- Acceso VNC a la máquina virtual del compañero (Apartado VI)

### Rúbrica:

Capturas de pantalla	40%
Descripción del trabajo desarrollado	30%
Entrega en plazo	30%

### Desarrollo:

**Antes de empezar, vamos a modificar ligeramente nuestro fichero Vagrantfile**

**Ampliando la memoria a 2GB, reencaminamos puerto 22 (host 2522) y asignamos ip:**

```
# -*- mode: ruby -*-
# vi: set ft=ruby :
Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/jammy64"
  config.vm.network "forwarded_port", guest: 22, host: 2522
  config.vm.network "private_network", ip: "192.168.56.11"
  config.vm.provider "virtualbox" do |vb|
    vb.memory = "2048"
    vb.cpus = 2
  end
end
```

Accedemos a la máquina virtual:

```
vagrant up
vagrant ssh
```

En la máquina, usuario *vagrant*, password *vagrant*

El acceso con password está desactivado por defecto en `/etc/ssh/sshd_config`. Para activarlo, modificar la línea:

*PasswordAuthentication yes*

## Reiniciar el servicio ssh

```
sudo service ssh restart
```

## I) Uso de ZeroTier <https://www.zerotier.com/>

Esta red permite crear VPN multiplataforma (Windows, Linux, MAC, Android, ...) de forma sencilla y segura. Podemos crear múltiples redes y de forma gratuita hasta 25 nodos por red.

Para la instalación de los cliente seguimos los pasos definidos en:

<https://www.zerotier.com/download/>

Dentro de la máquina Ubuntu, conectaremos todas las máquinas virtuales entre sí.

Para conectar a una red con el identificador indicado en clase.

```
zerotier-cli join XXXXXXXXXX
```

Ej: 14d718456ce156a6 Para conectar/salir a una red:

```
zerotier-cli join 14d718456ce156a6  
zerotier-cli leave 14d718456ce156a6
```

Comprobar la lista de redes

```
zerotier-cli listnetworks
```

Comprobamos las conexiones entre compañeros haciendo ping.

## II) Instalación del entorno gráfico en la máquina virtual

Instalamos el escritorio en nuestra máquina virtual:

```
sudo apt -y update && sudo apt -y upgrade  
sudo apt -y install tasksel
```

Podemos instalar Lubuntu-desktop

```
sudo tasksel install lubuntu-desktop  
sudo apt -y install lxsession  
sudo apt -y install tightvncserver
```

Ejecutar vncpasswd para definir la clave de acceso a vnc

```
vncpasswd
```

Crear el fichero *./vnc/xstartup*

```
/usr/bin/startlxde  
lxterminal &  
/usr/bin/lxsession -s LXDE &
```

**Importante:** el fichero `./vnc/xstartup` tiene que tener permisos de ejecución `x` y pertenece al usuario `vagrant` (no debe pertenecer al usuario `root` ya que da problemas de permisos de ejecución).

```
chmod u+x ./vnc/xstartup
chown vagrant:vagrant ./vnc/xstartup
```

### III) Acceso mediante SSH al escritorio

Levantamos un escritorio remoto en la máquina virtual al que podremos acceder desde el host

```
vncserver :1
```

Desde nuestro equipo (HOST) accedemos a la máquina virtual:

```
ssh -L5901:localhost:5901 vagrant@192.168.56.11
```

Con la opción `-L` reencaminamos nuestro puerto local TCP 5901 con el puerto que “ve” el nodo al que accedemos por SSH, por lo que “localhost” es la propia máquina virtual y el último 5901 es el puerto al que queremos conectar.

Comprobemos el reencaminamiento que también tenemos en localhost, puerto 2522 hacia la máquina virtual.

Instalamos el cliente de vnc en nuestro ordenador. Puede ser `realvnc`, o `ultravnc`, o `tightvnc`.  
Accedemos a nuestro escritorio remoto.

### IV) Acceso desde escritorio

Una vez dentro del escritorio remoto, abrimos un terminal y podemos acceder a cualquier equipo en la red. Ej: ping.

**Por parejas:**

### V) Acceso por SSH entre máquinas virtuales

Instalamos ZeroTier dentro de las máquinas virtuales. Comprobar que podemos entrar de máquina desde nuestra máquina virtual a la máquina virtual de nuestro compañero por SSH.

### VI) Comprobar que mediante VNC podemos acceder a la máquina virtual de nuestro compañero.

Es decir, si los nodos alfa1 y alfa2 son los nodos host, y beta1 y beta2 son cada una de las máquinas virtuales, queremos hacer un reencaminamiento SSH alfa1-Beta1, que, aunque no tengamos instalado ZeroTier en las máquinas alfa, desde el VNC de alfa1 se pueda acceder a la máquina virtual Beta2.

### VII) Reencaminamiento automático con ProxyJump

Podemos simplificar el acceso entre nodos utilizando el `portForthaedin`

Modificamos nuestro fichero `$HOME/.ssh/config`

```
Host mialfa
  Hostname 192.168.56.11
```

```
User vagrant
```

```
Host beta2
```

```
Hostname 192.168.191.3
```

```
ProxyJump mialfa
```

Donde es la IP de zerotier del nodo de nuestro compañero. De esta forma, al ejecutar desde nuestro host:

```
ssh beta2
```

Podemos acceder a la máquina virtual de nuestro compañero. Por lo que si añadimos el reencaminamiento del puerto:

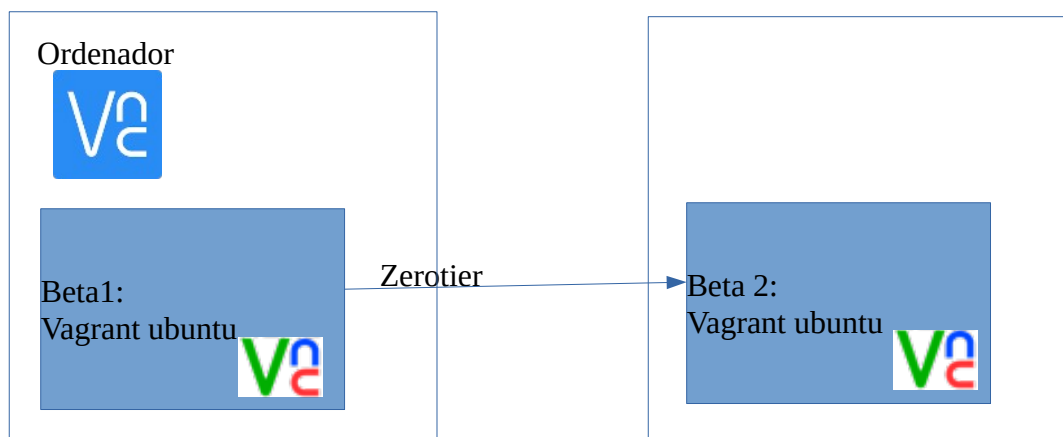
```
ssh -L5901:localhost:5901 beta2
```

Se accede al escritorio remoto de nuestro compañero.

Crear un reencaminamiento automático con portForwarding (creando una entrada en el fichero .ssh)

alfa1

alfa2



Podemos añadir también dentro del fichero .ssh/config los reencaminamientos de puertos que queramos:

```
LocalForward 5901 localhost:5901
```

**Los números no tienen por qué ser iguales, simplemente indicamos puerto origen y puerto remoto.**