

ADD1314

Adrián y Jacobo

ADD1314

Adrián y Jacobo

fecha de publicación 19/11/2013

Este docbook recoge los trabajos de la asignatura ADD1314.

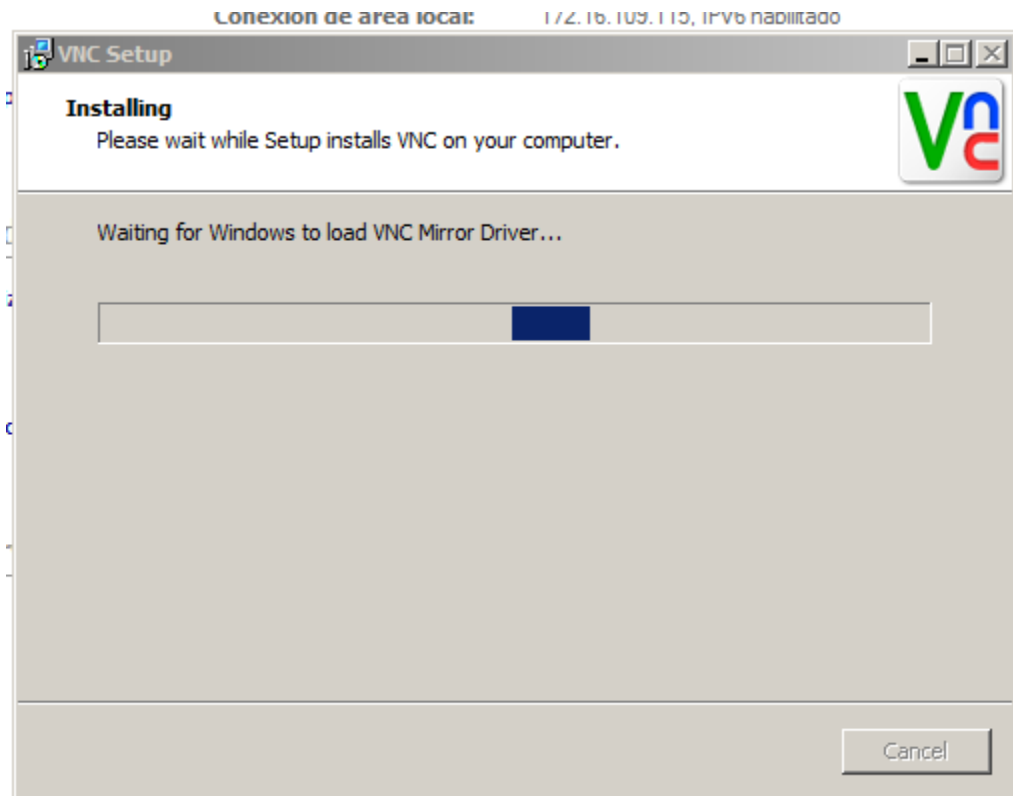
Tabla de contenidos

1. acceso remoto	1
VNC	1
Escritorio Remoto	6
Terminal Server	9
RemoteApp	11
2. clientes ligeros	13
Preparacion del servidor	13
Instalacion	13
3. SSH	14
Servidor Windows	14
Servidor (Windows 2008 Server):	14
Cliente 1 (Xubuntu):	16
Cliente 2 (Windows 7):	22
Servidor Linux	26
Instalación básica	26
Personalización del prompt Bash	28
Autenticación mediante claves públicas	29
Uso de SSH como túnel para X	30
Aplicaciones Windows nativas	31
Restricciones de uso	34

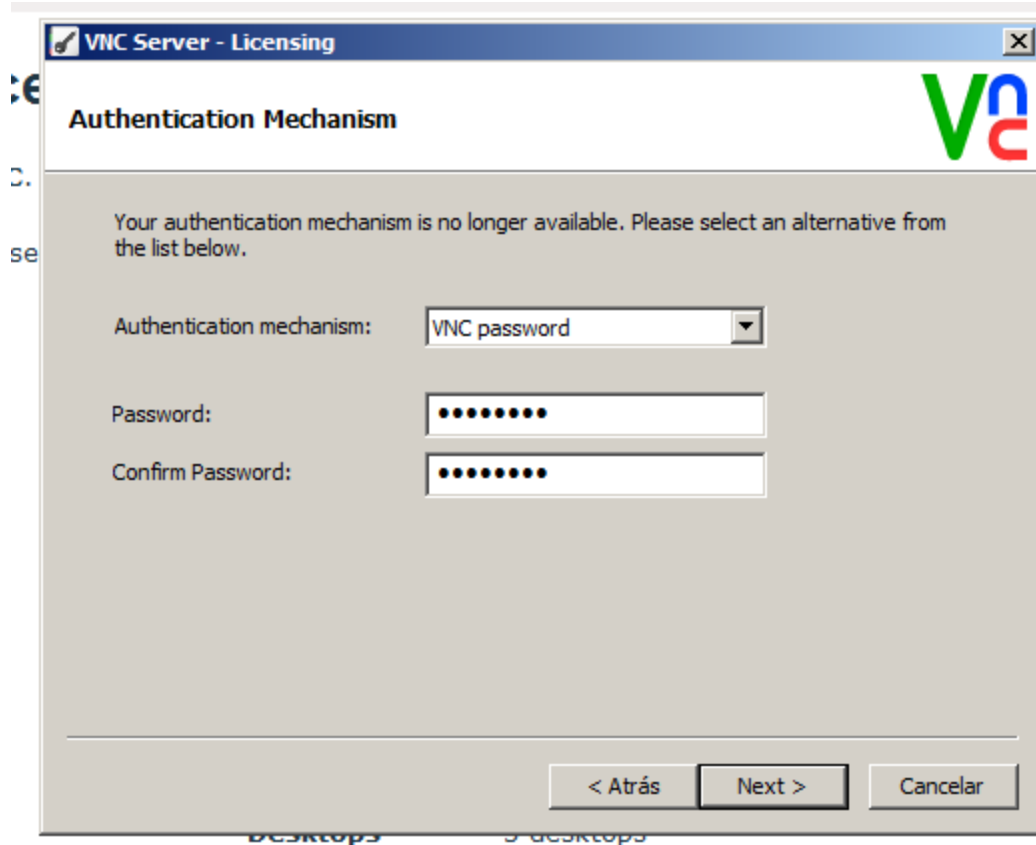
Capítulo 1. acceso remoto

VNC

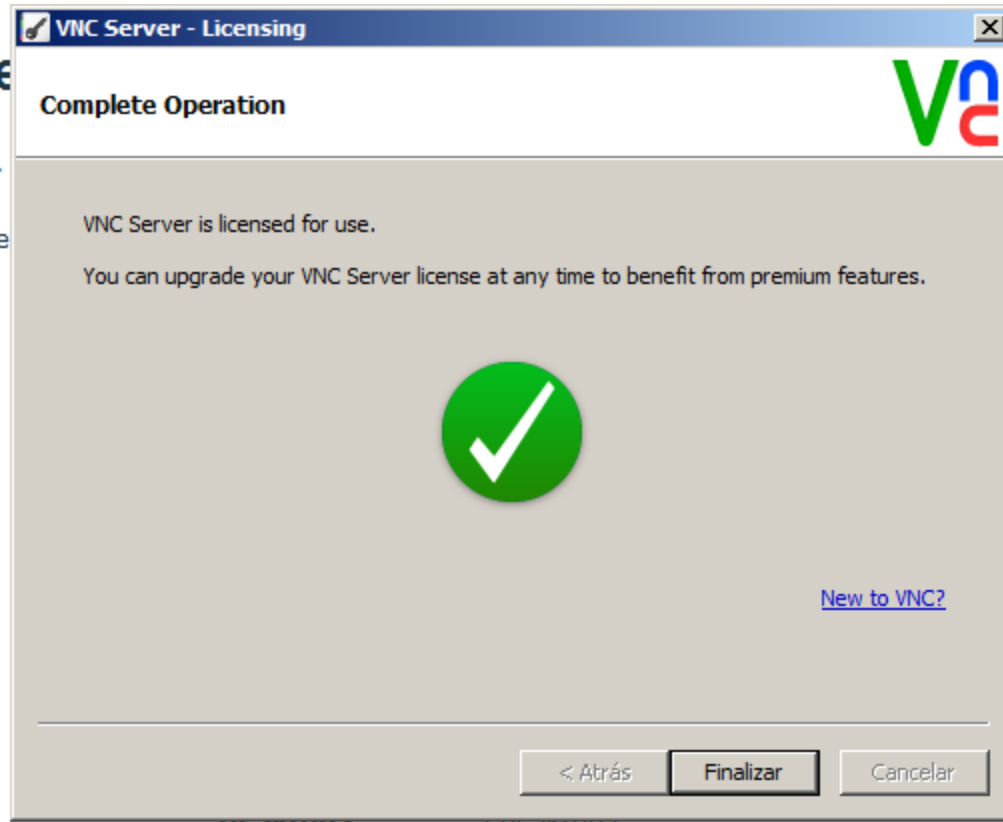
Lo primero que hicimos fue instalar el VNC para poder acceder a la máquina remota.



Configuramos el servidor VNC.

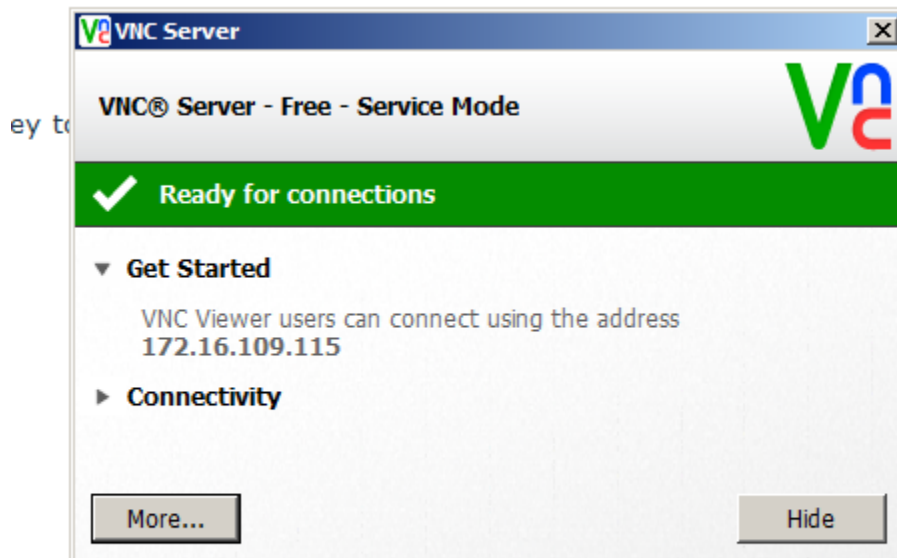


Finaliza la instalación.



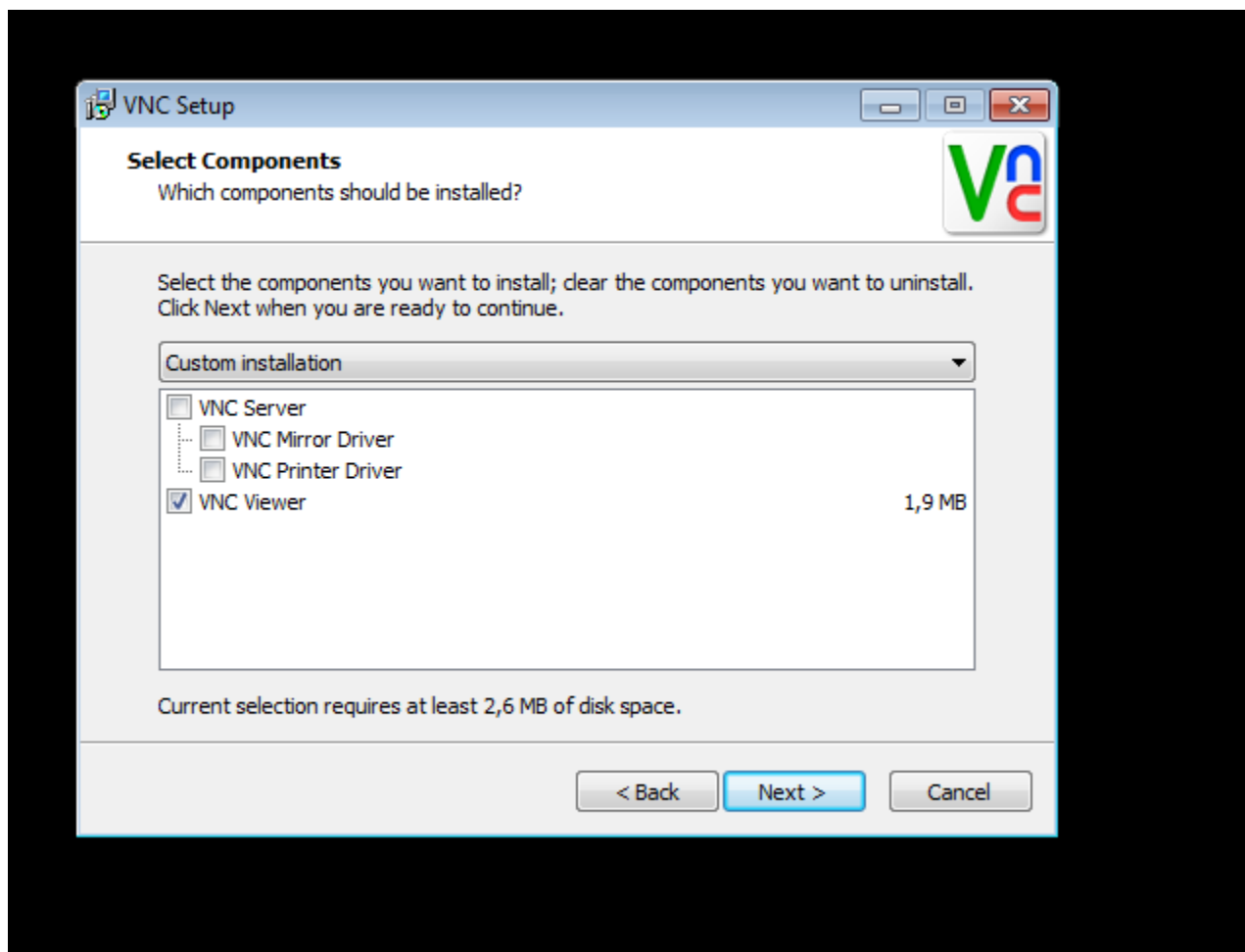
Abrimos el VNC en el servidor.

Use key

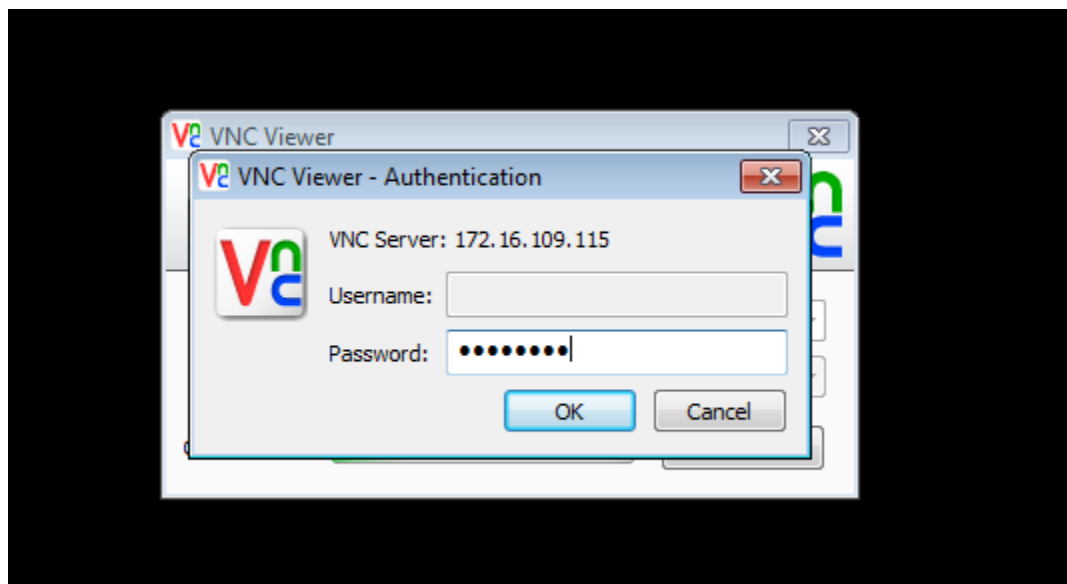


Product	VNC
License type	Free
Support	No support
Desktops	5 desktops

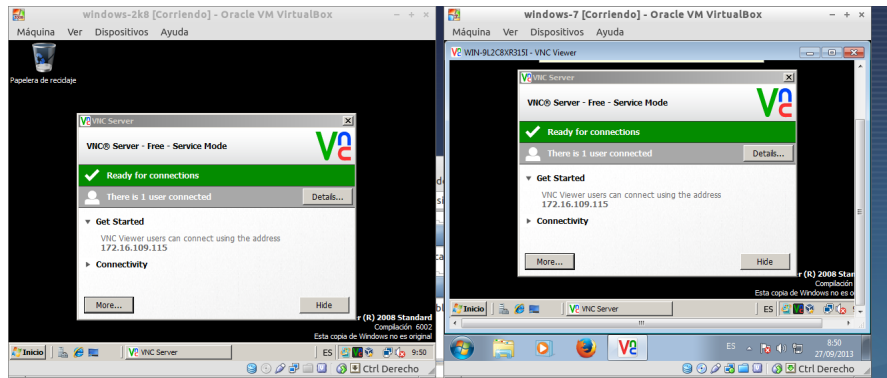
Instalamos el VNC en el cliente.



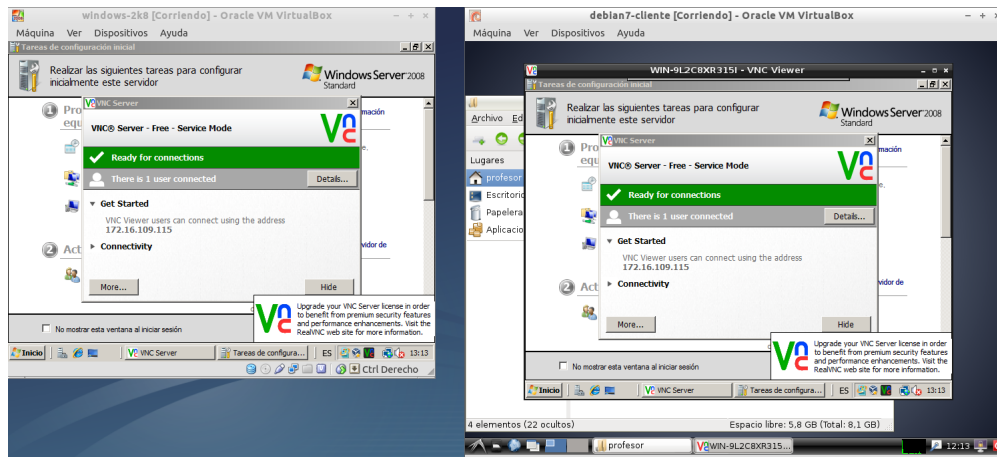
Conectamos el cliente con el servidor.



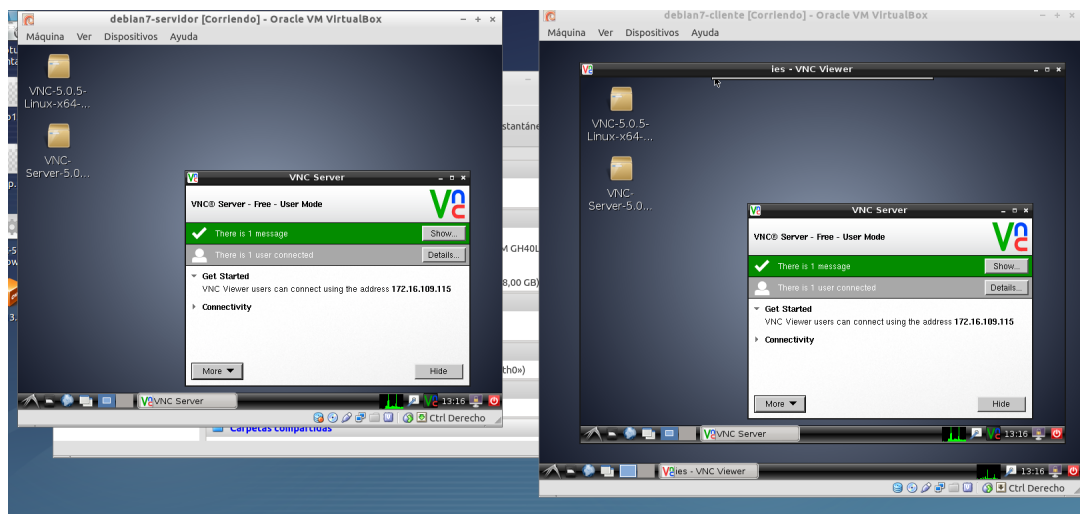
Los dos Windows conectados con VNC.



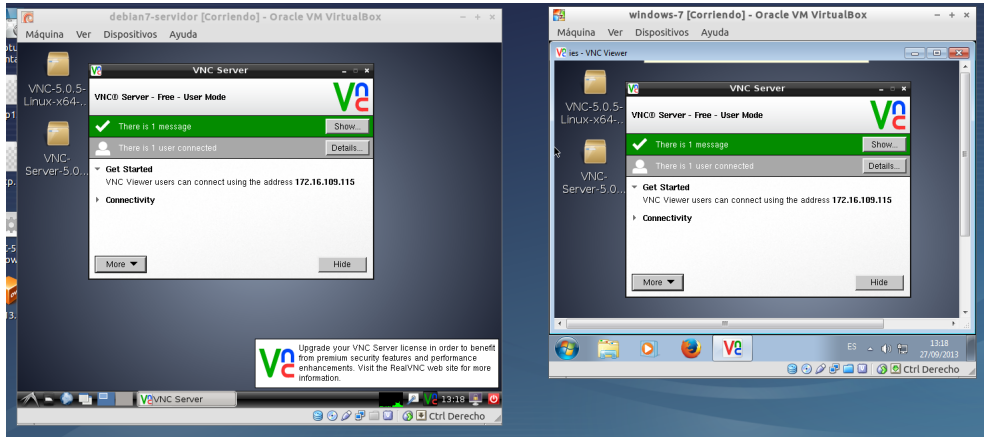
Windows y Debian conectados con VNC.



Los dos Debian conectados con VNC.

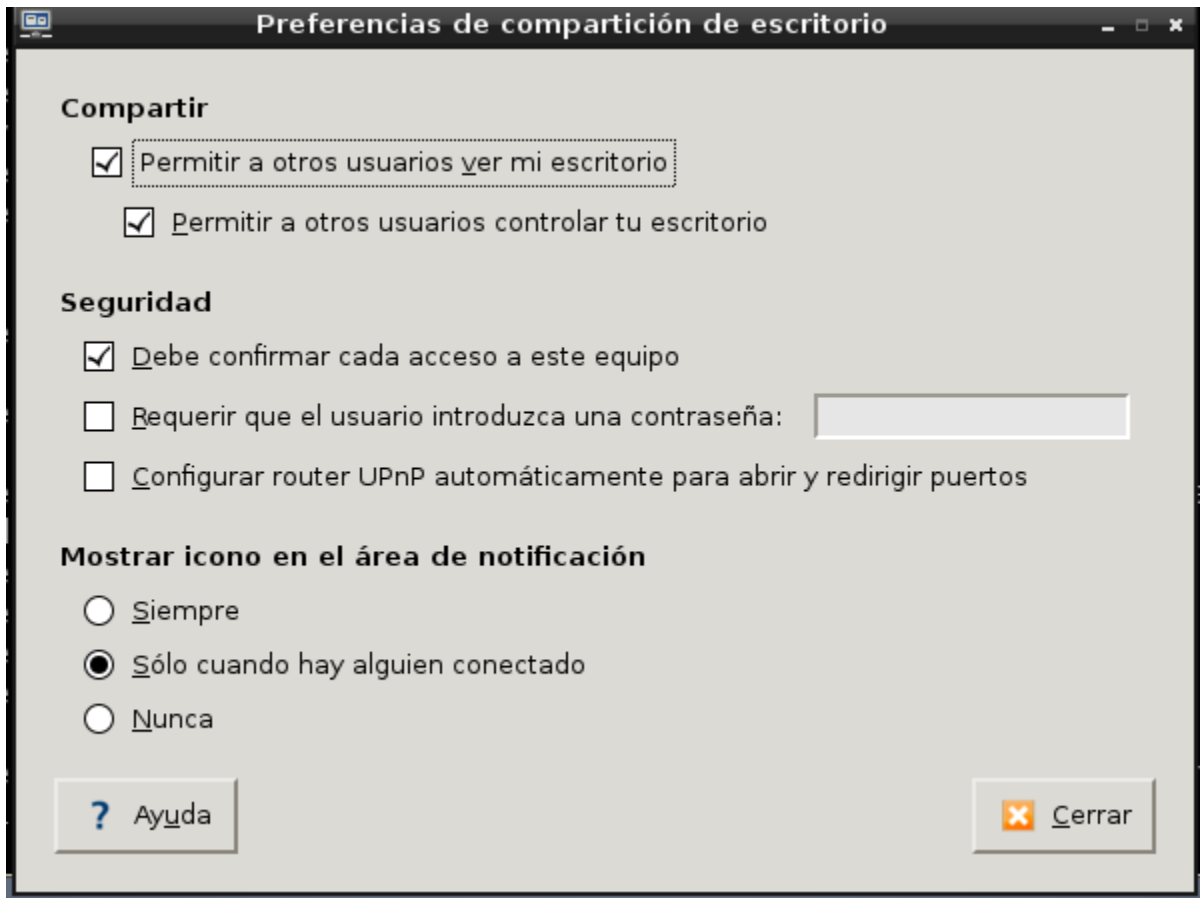


Debian y Windows conectados con VNC.

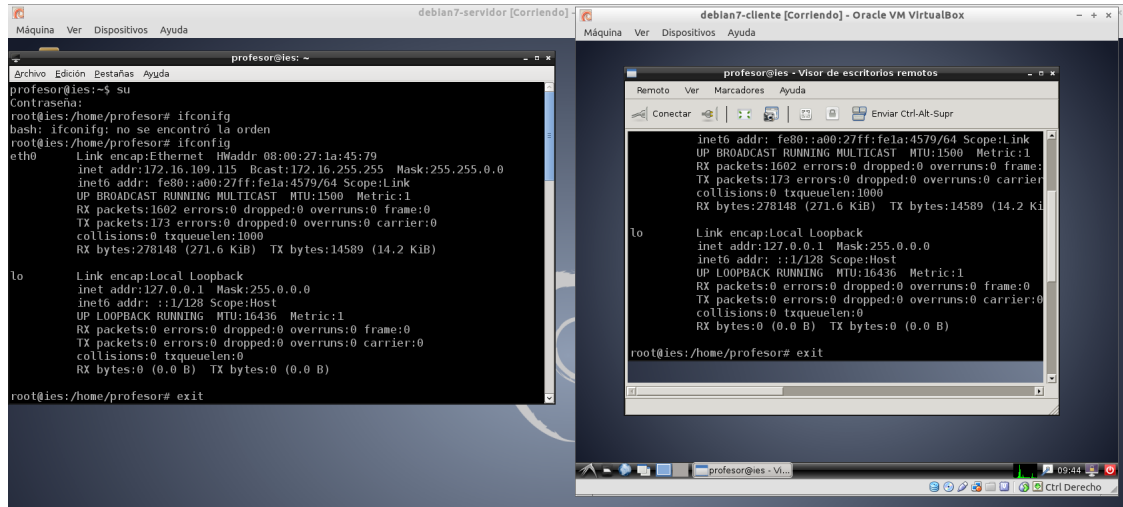


Escritorio Remoto

Configurando el escritorio remoto de Linux.



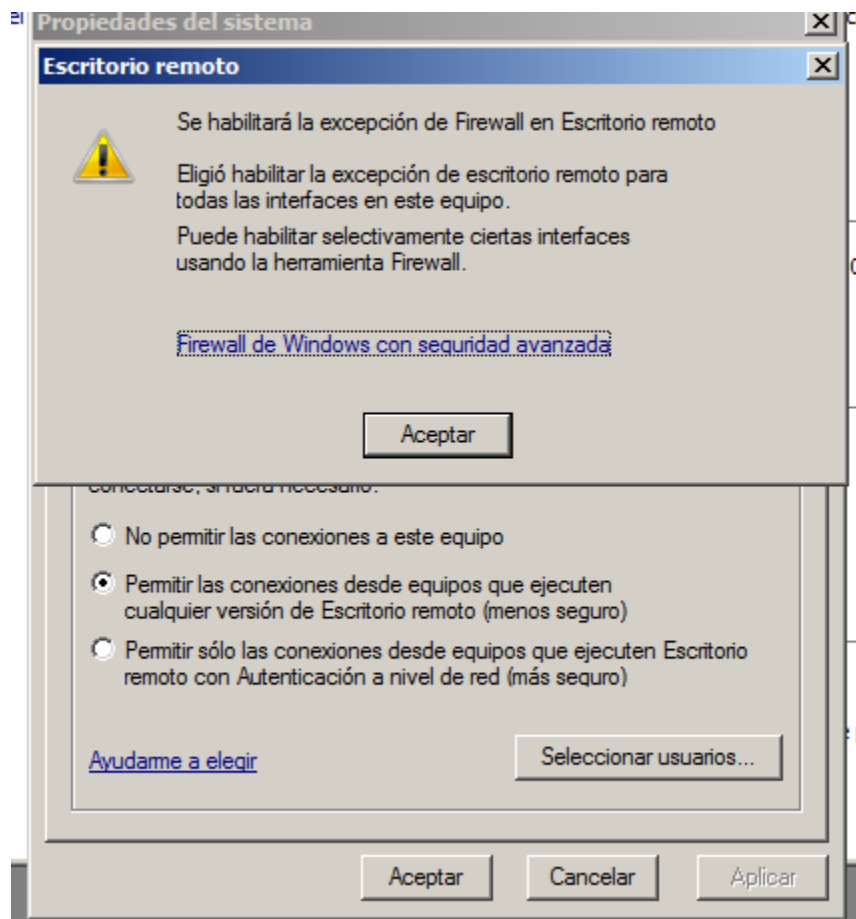
Debian y Debian conectados con Escritorio Remoto.



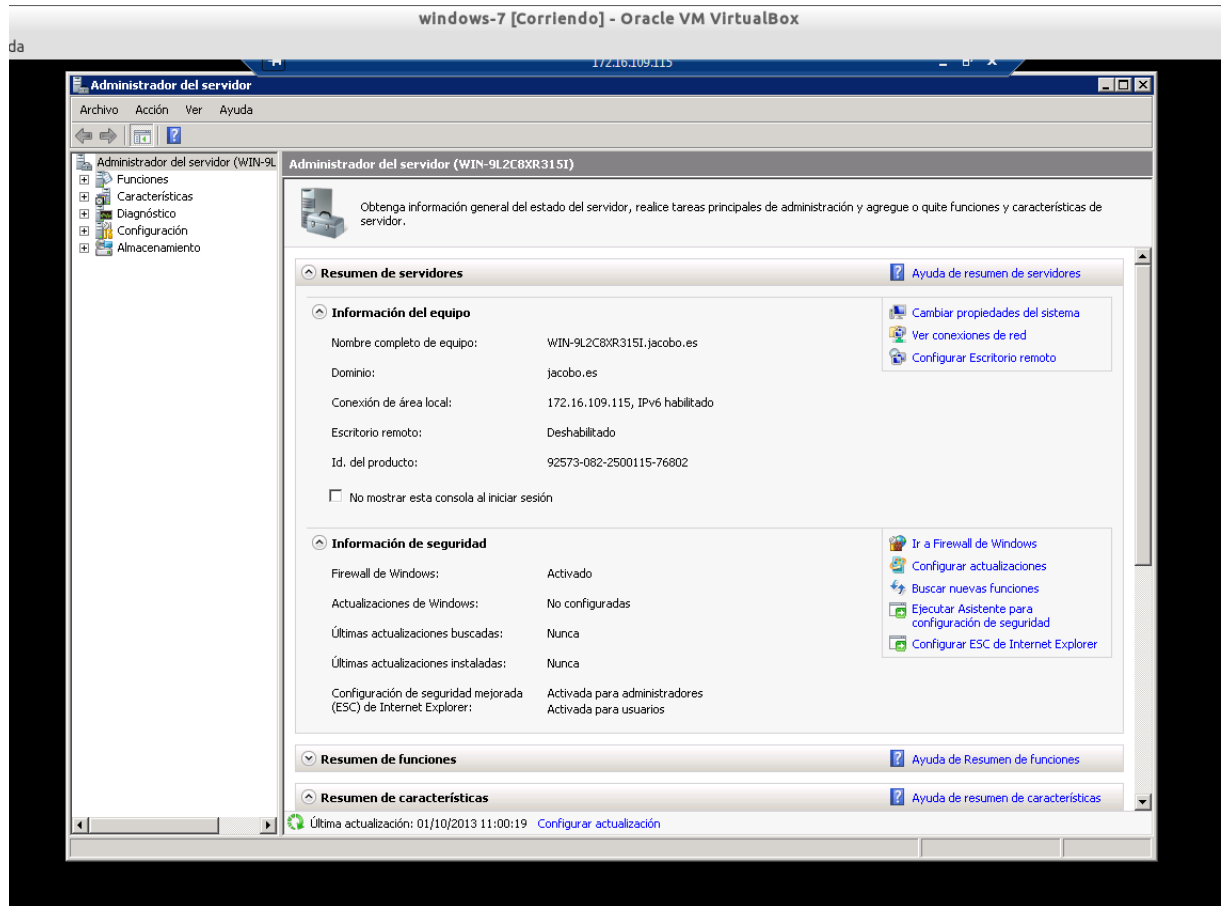
Comando para arancar el vino server.

```
/usr/lib/vino/vino-server
```

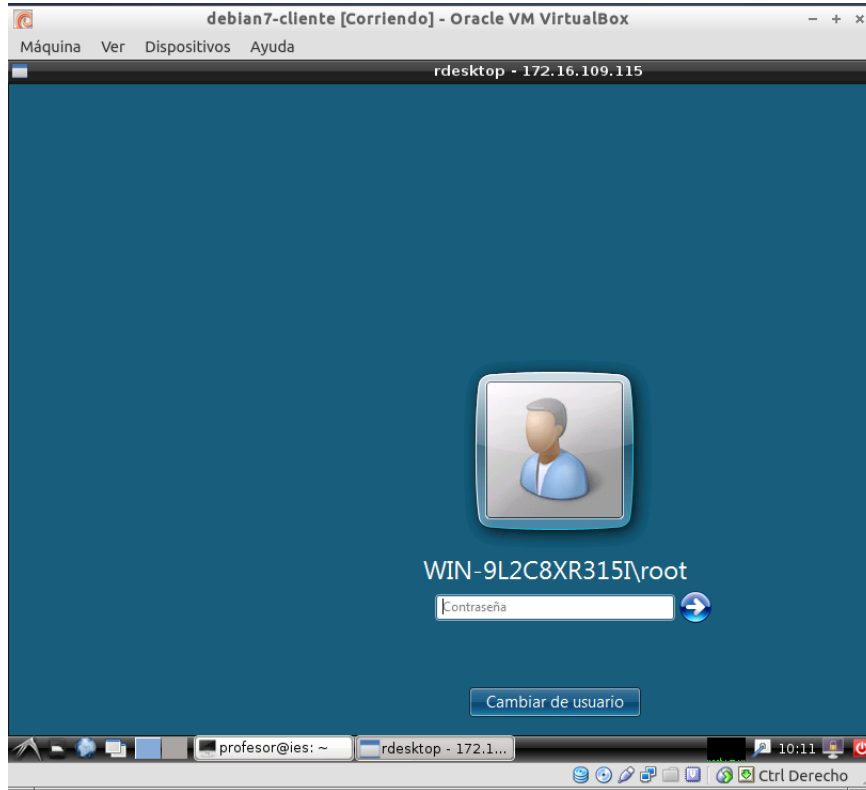
Configurando escritorio remoto en Windows.



Windows y Windows conectados con Escritorio Remoto.

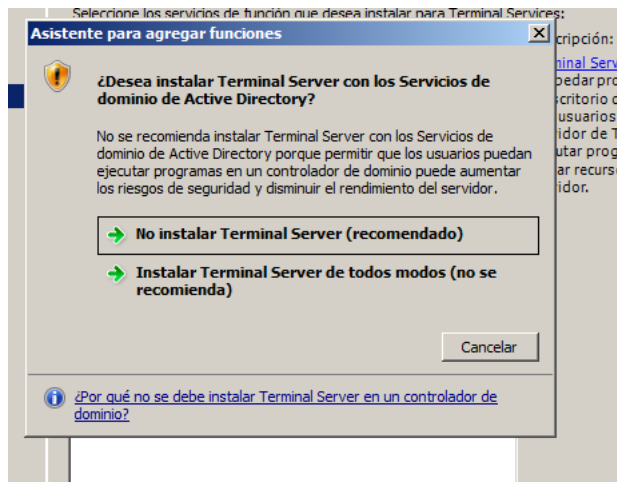


Debian y Windows conectados con Escritorio Remoto.

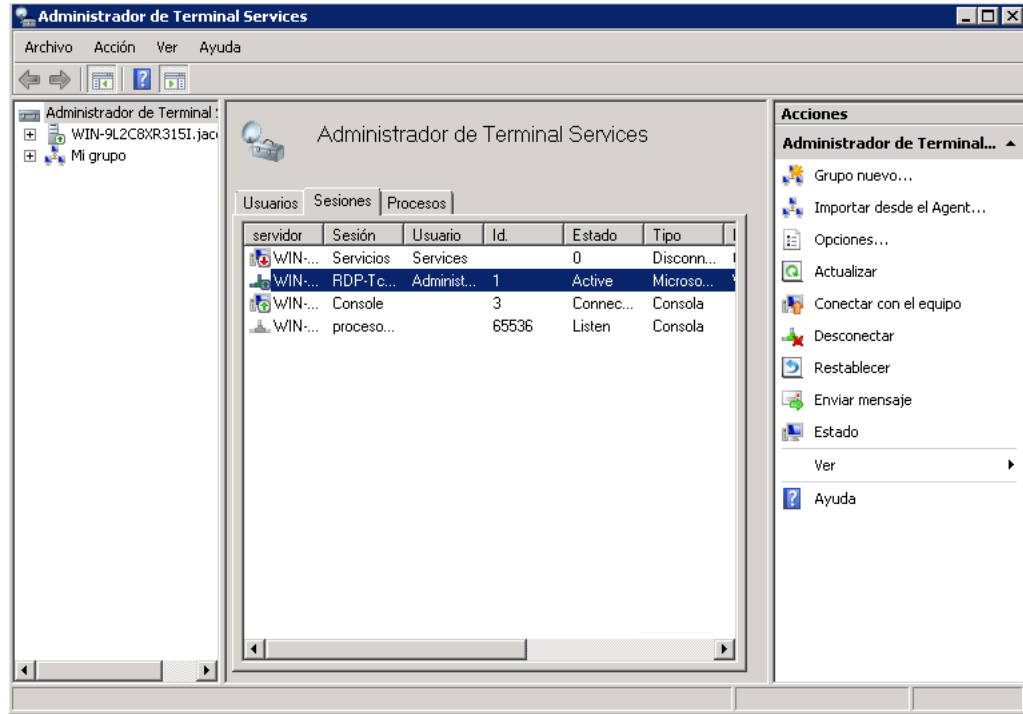


Terminal Server

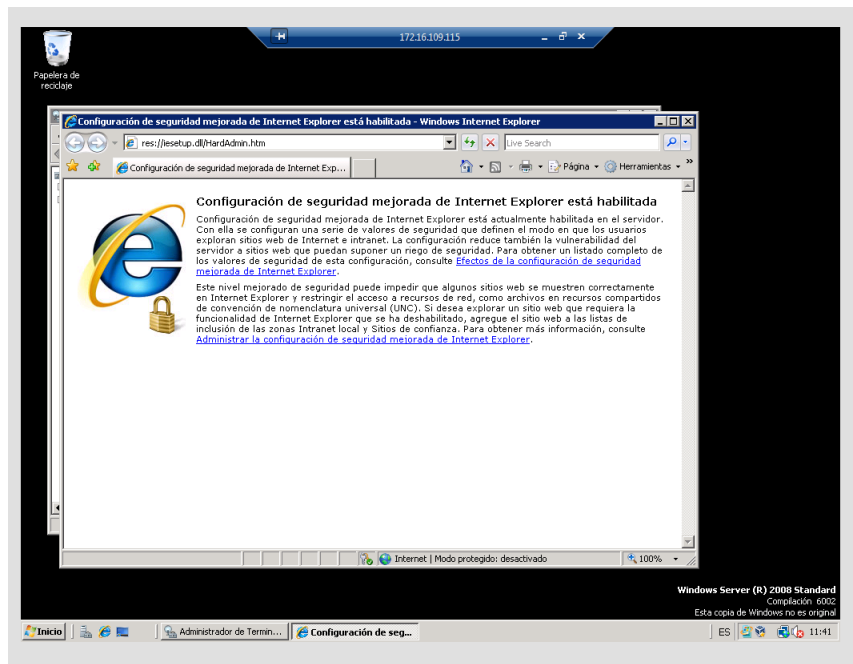
Asistencia de instalación.



Demostración de que está funcionando de manera correcta el Terminal Server.



Probando una aplicación con el Terminal Server en Windows.

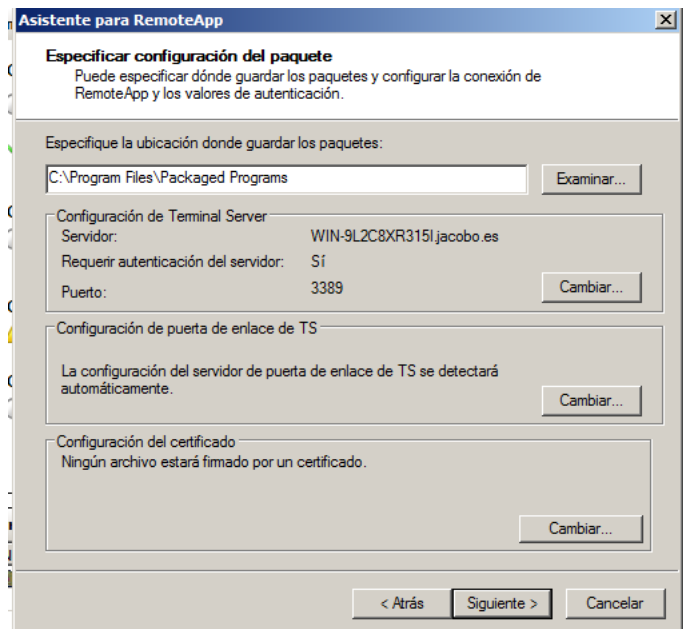


Probando una aplicación con el Terminal Server en Debian.



RemoteApp

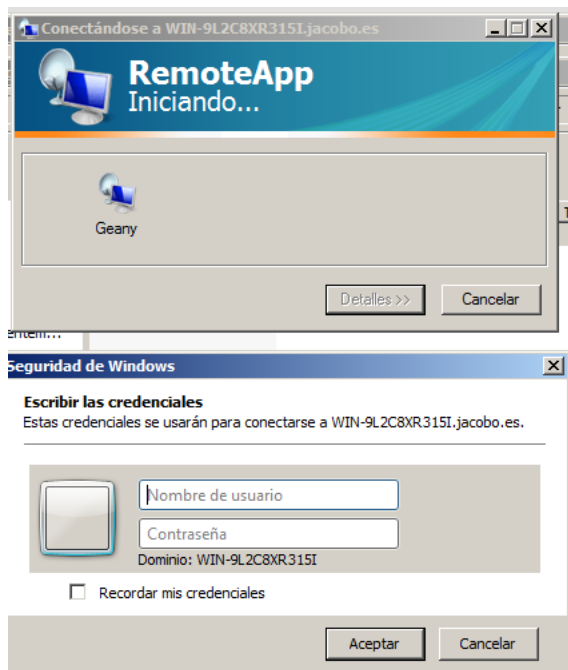
Instalación RemoteApp.



Programas añadidos para RemoteApp.

Programas RemoteApp				
Nombre	ruta\ruta de acceso	Acceso web ...	Argumentos	
Geany	C:\Program Files\Geany\bin\...	Sí	Deshabilitado	

Iniciando el RemoteApp.



Capítulo 2. clientes ligeros

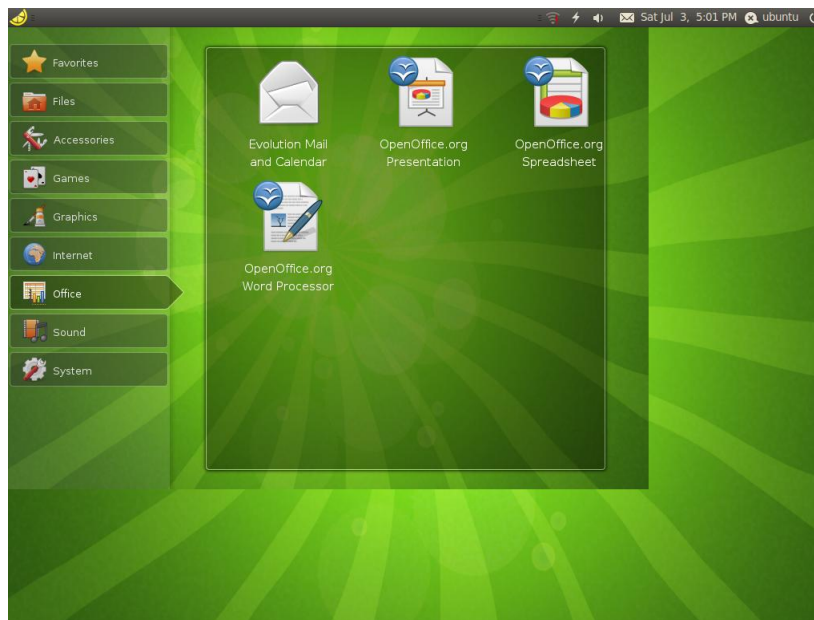
Preparacion del servidor

El servidor tendra que tener dos interfaces de red

- Una externa para comunicarnos con internet, tenemos que tenerla en adaptador puente
- Una interna para conectarnos con los clientes, su ip sera estatica, en la misma red que los clientes y la pondremos como red interna

Instalacion

Lo primero que teníamos que hacer para realizar esta practica era instalar un sistema operativo Linux. Yo personalmente escogí uno llamado Easy peasy es una distribución basada en Ubuntu pero diseñada especialmente para los netbook.



Una vez hecho la instalación teníamos que instalar unos paquetes, para ellos lanzábamos estos comandos:

```
sudo apt-get install ltsp-server-standalone
```

```
sudo ltsp-build-client
```

Una vez realizado la instalación tendríamos que configurar la ip yo a mi maquina servidora le puse la 192.168.0.1 que es la que coge por defecto el servidor dhcp para no tener que estar configurándolo.

Una ves hecho todo esto y configurado dos interfaces de red una con la ip mencionada anterior mente y otra automática solo nos quedaba conectar el cliente. Y este es el resultado final.

ir al video de la comprobación [<http://www.youtube.com/watch?v=wwTOQWQmWsw>].

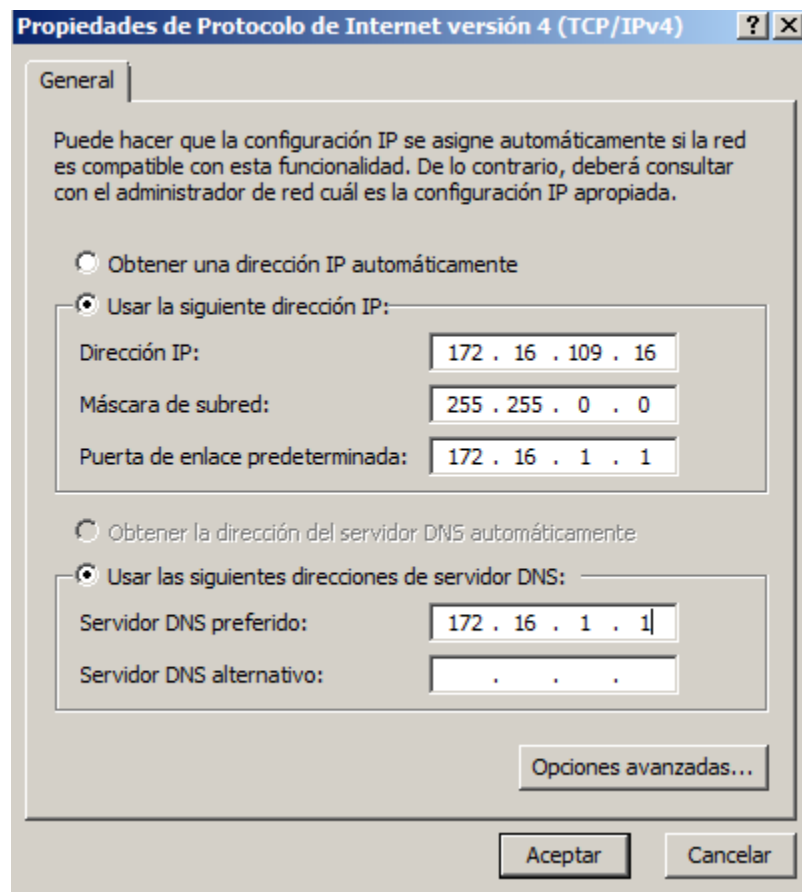
Capítulo 3. SSH

Servidor Windows

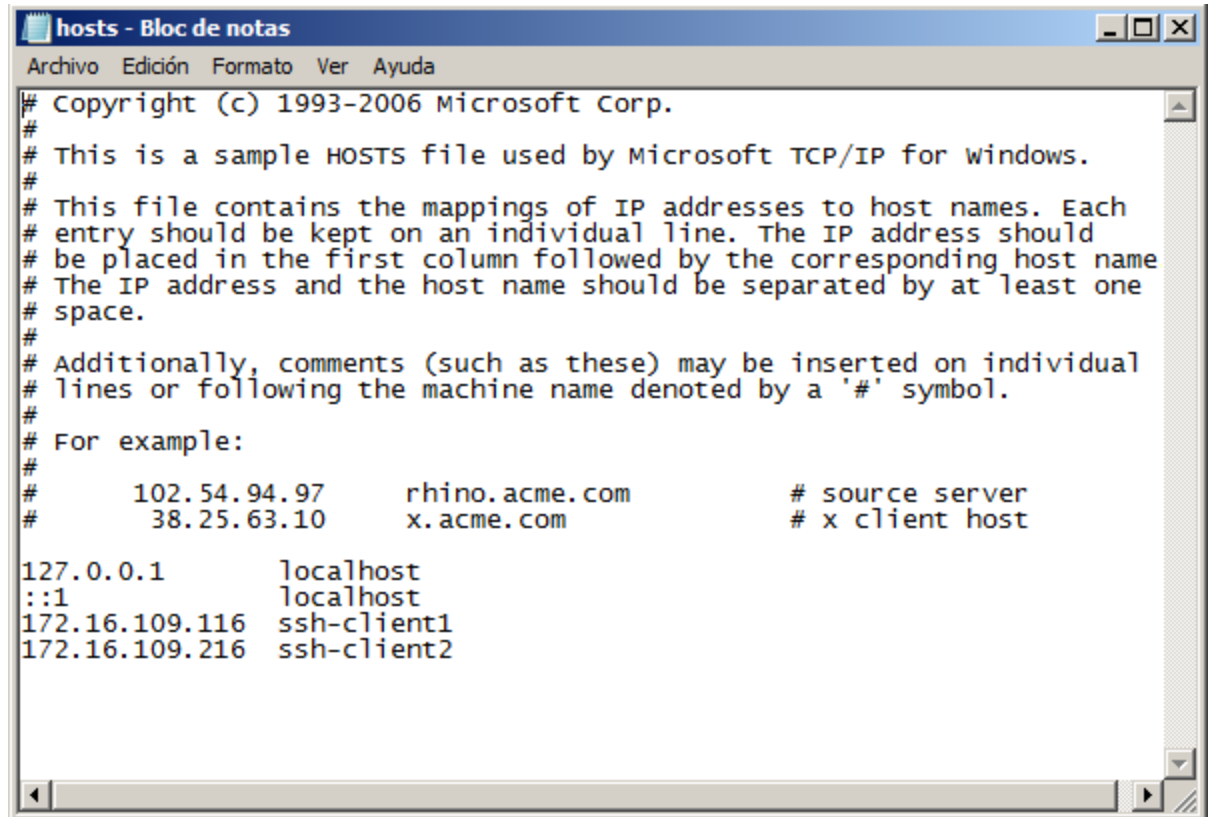
- Servidor: Windows 2008 Server.
- Cliente 1: Xubuntu.
- Cliente 2: Windows 7.

Servidor (Windows 2008 Server):

IP's del servidor Windows 2008 Server:



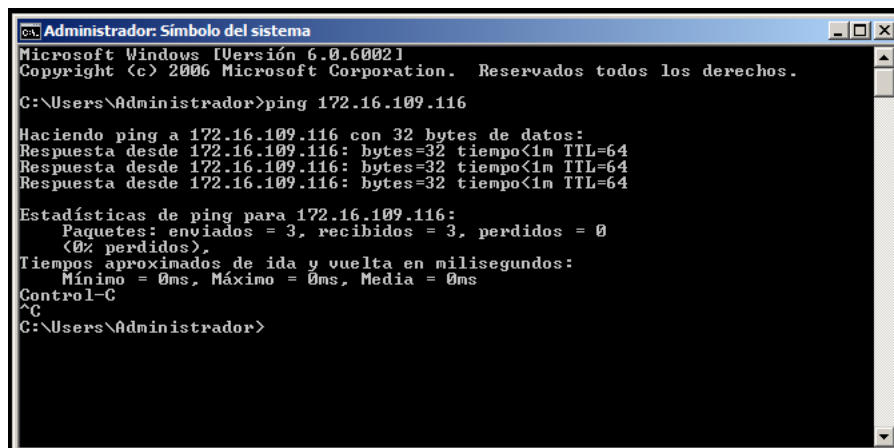
Añadimos en C:\Windows\System32\drivers\etc\hosts el equipo ssh-client1 y ssh-client2:



```
# Copyright (c) 1993-2006 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com           # source server
#       38.25.63.10       x.acme.com               # x client host

127.0.0.1        localhost
::1              localhost
172.16.109.116   ssh-client1
172.16.109.216   ssh-client2
```

Probamos hacer ping al cliente 1:

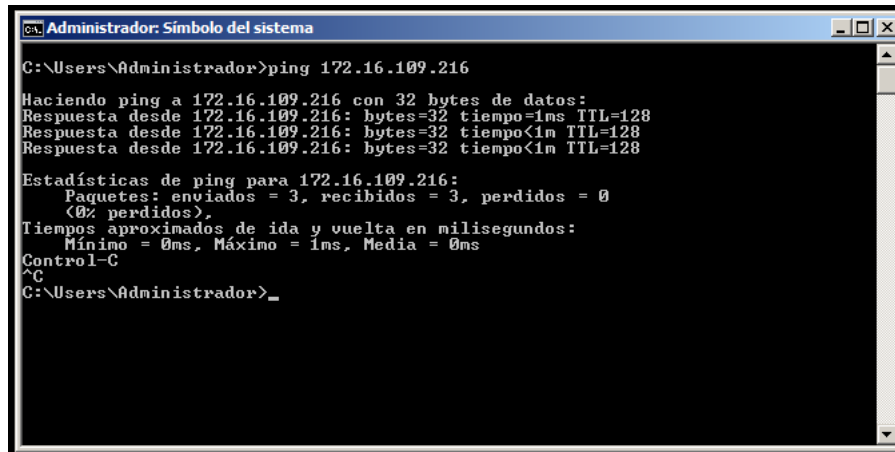


```
C:\Users\Administrador>ping 172.16.109.116

Haciendo ping a 172.16.109.116 con 32 bytes de datos:
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 172.16.109.116:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
              Mínimo = 0ms, Máximo = 0ms, Media = 0ms
Control-C
^C
C:\Users\Administrador>
```

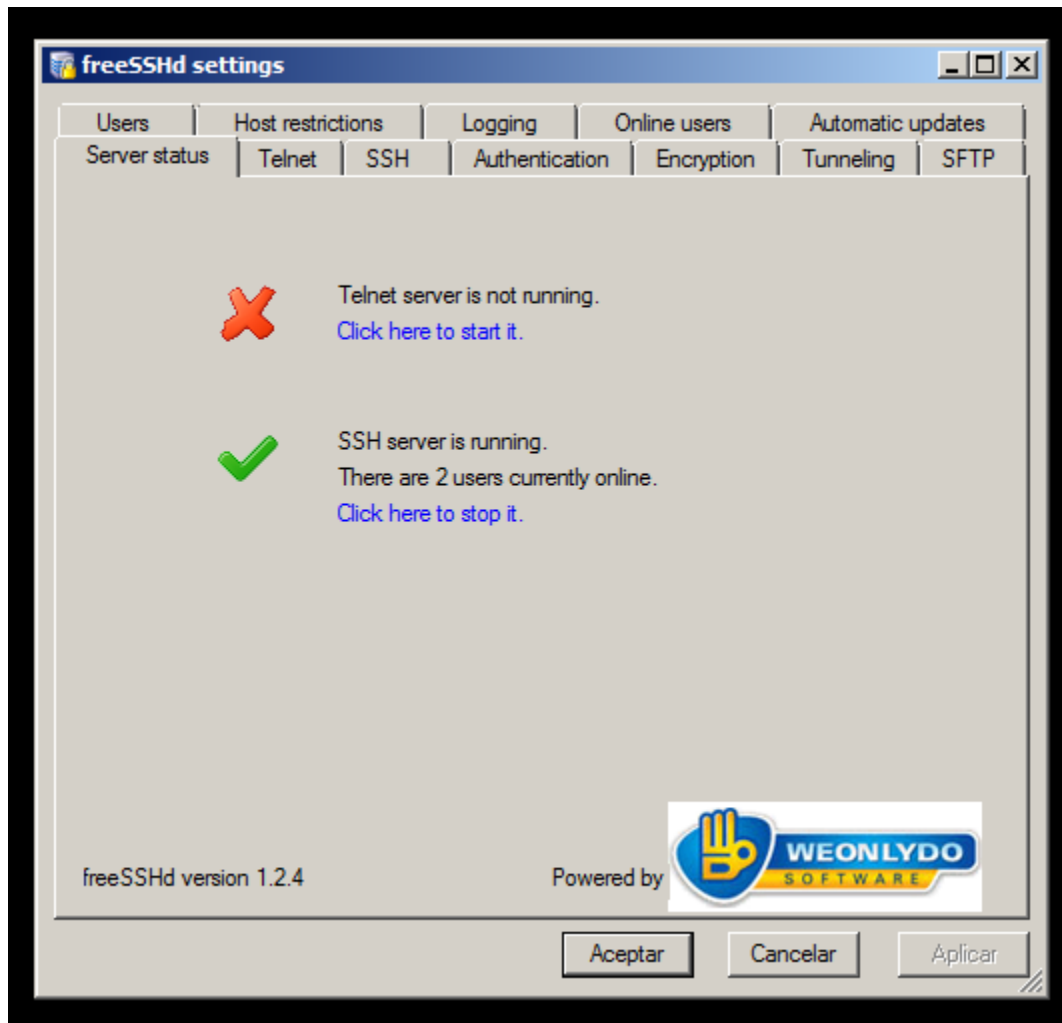
Probamos hacer ping al cliente 2:



```
C:\Users\Administrador>ping 172.16.109.216
Haciendo ping a 172.16.109.216 con 32 bytes de datos:
Respuesta desde 172.16.109.216: bytes=32 tiempo=1ms TTL=128
Respuesta desde 172.16.109.216: bytes=32 tiempo<1m TTL=128
Respuesta desde 172.16.109.216: bytes=32 tiempo<1m TTL=128

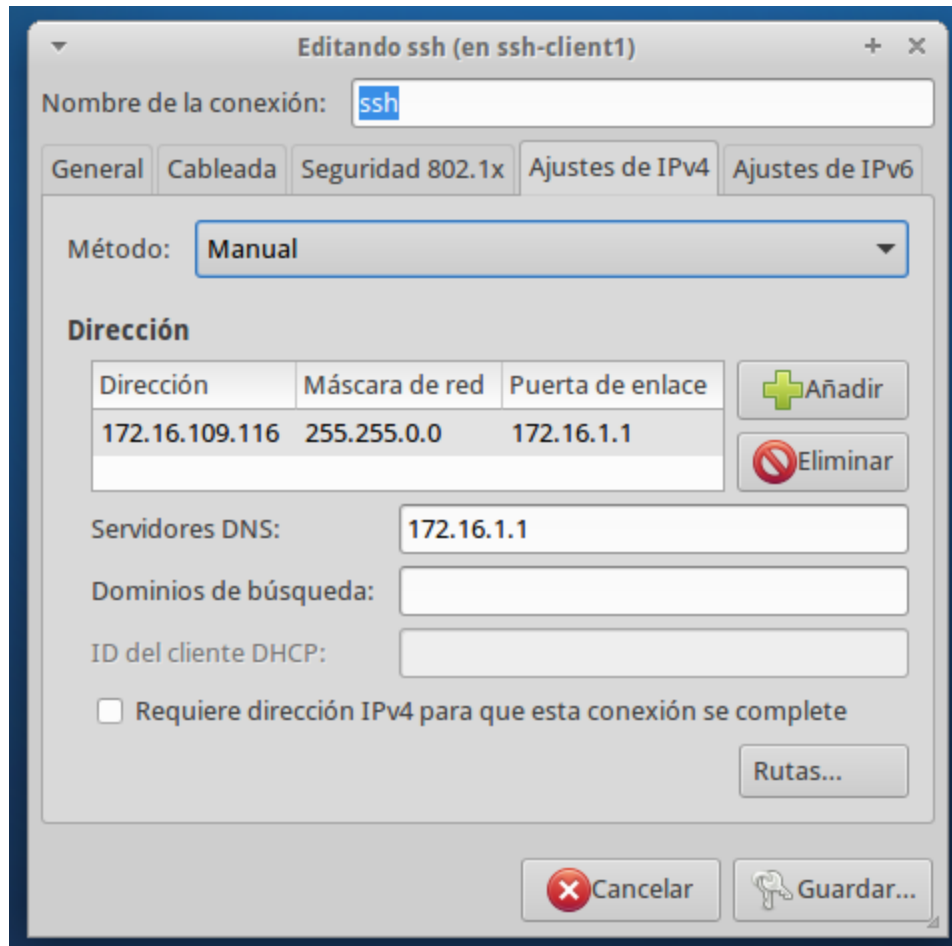
Estadísticas de ping para 172.16.109.216:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 1ms, Media = 0ms
Control-C
^C
C:\Users\Administrador>
```

Comprobamos que ambos clientes están conectados al servidor mediante SSH:



Cliente 1 (Xubuntu):

IP's del cliente1 Xubuntu:



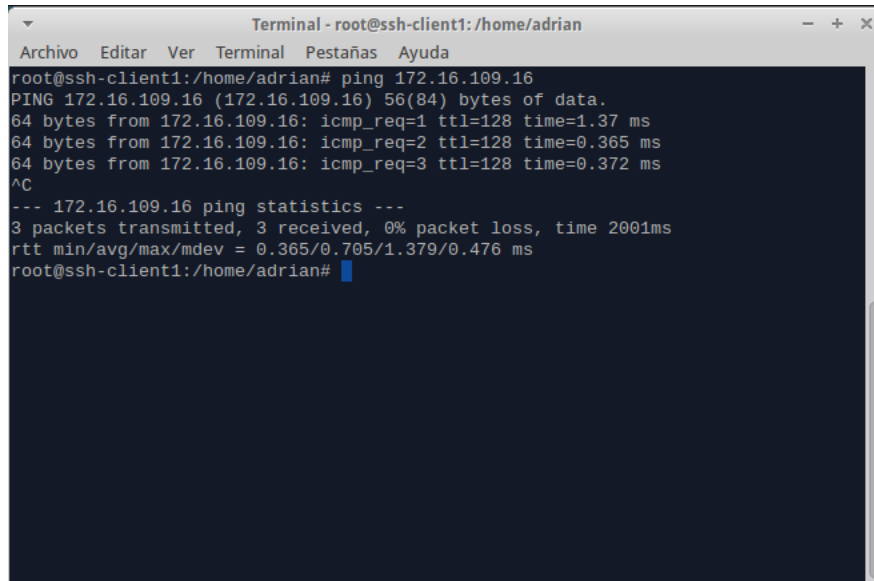
Añadimos en /etc/hosts el equipo ssh-server y ssh-client2:

```
GNU nano 2.2.6 Archivo: hosts Modificado
127.0.0.1    localhost
127.0.1.1    profesor-VirtualBox
172.16.109.16 ssh-server
172.16.109.216 ssh-client2

# The following lines are desirable for IPv6 capable hosts
::1    ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

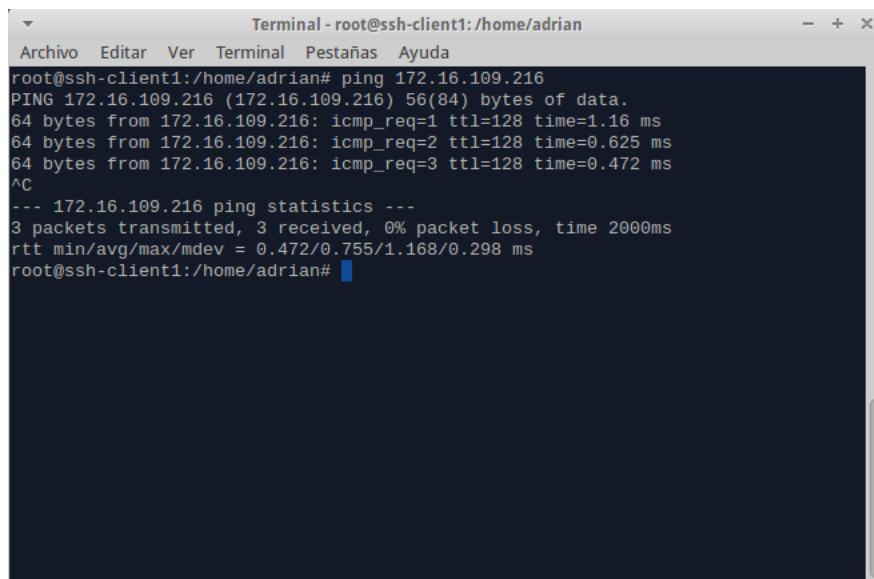
^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía
```

Probamos hacer ping al servidor:

A terminal window titled "Terminal - root@ssh-client1: /home/adrian" with a menu bar (Archivo, Editar, Ver, Terminal, Pestañas, Ayuda). The terminal shows a ping command to 172.16.109.16. The output shows three successful pings with times of 1.37 ms, 0.365 ms, and 0.372 ms. A statistics summary shows 3 packets transmitted, 3 received, 0% packet loss, and a time of 2001ms.

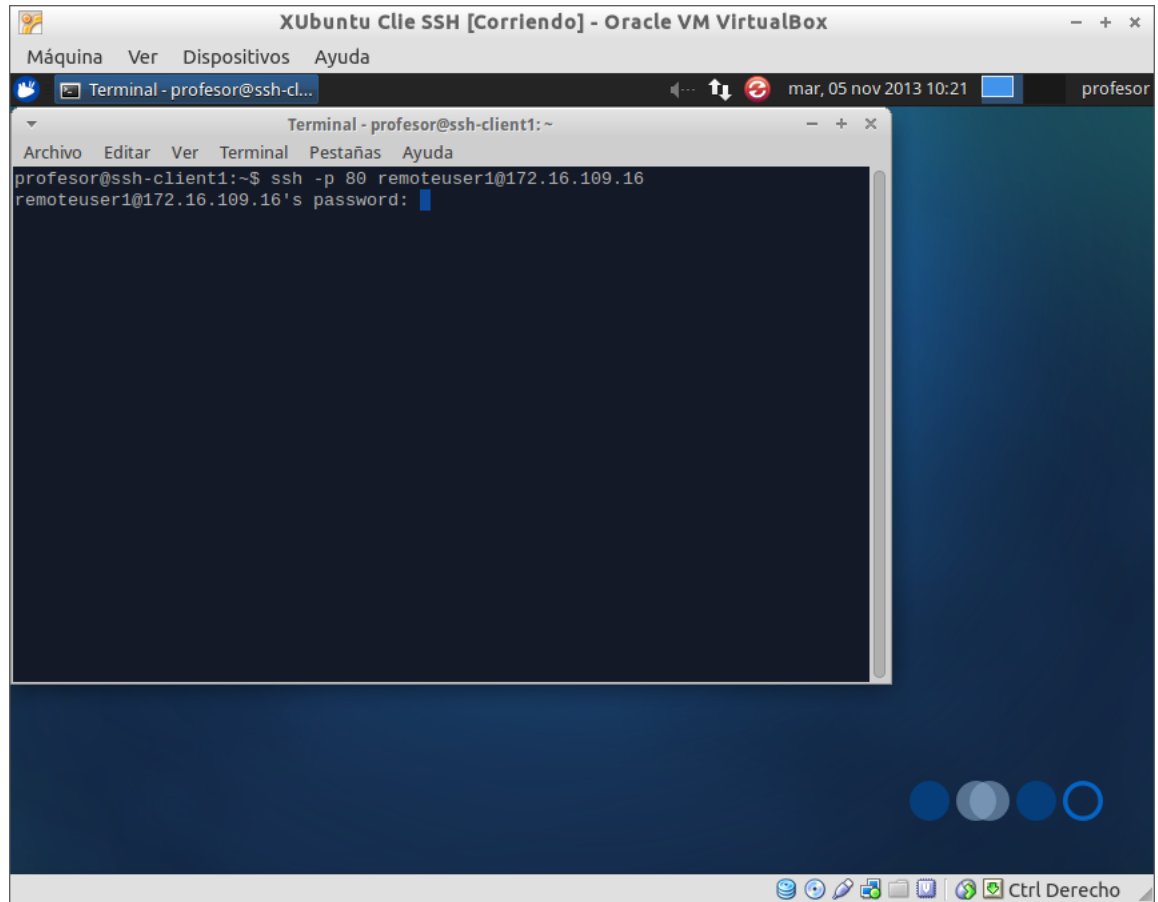
```
root@ssh-client1:/home/adrian# ping 172.16.109.16
PING 172.16.109.16 (172.16.109.16) 56(84) bytes of data.
64 bytes from 172.16.109.16: icmp_req=1 ttl=128 time=1.37 ms
64 bytes from 172.16.109.16: icmp_req=2 ttl=128 time=0.365 ms
64 bytes from 172.16.109.16: icmp_req=3 ttl=128 time=0.372 ms
^C
--- 172.16.109.16 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.365/0.705/1.379/0.476 ms
root@ssh-client1:/home/adrian#
```

Probamos hacer ping al cliente 2:

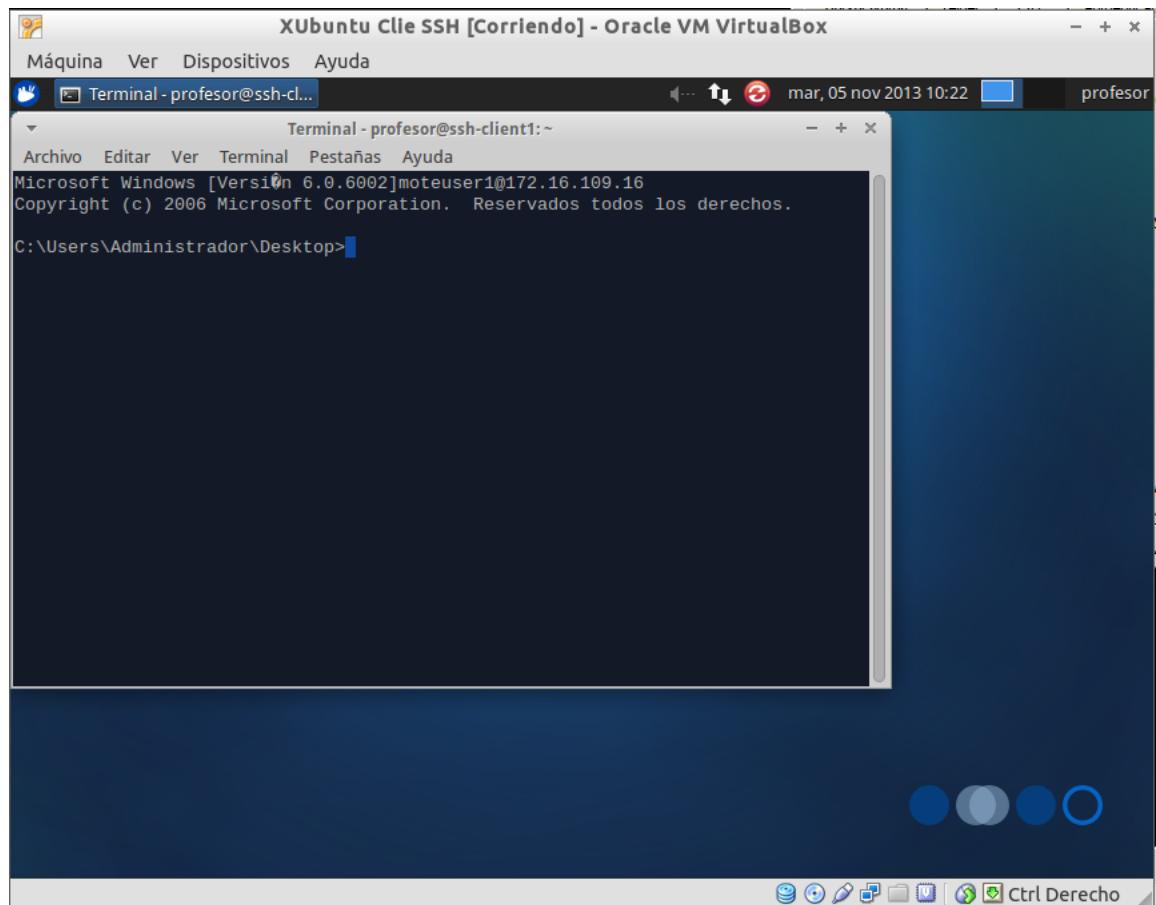
A terminal window titled "Terminal - root@ssh-client1: /home/adrian" with a menu bar (Archivo, Editar, Ver, Terminal, Pestañas, Ayuda). The terminal shows a ping command to 172.16.109.216. The output shows three successful pings with times of 1.16 ms, 0.625 ms, and 0.472 ms. A statistics summary shows 3 packets transmitted, 3 received, 0% packet loss, and a time of 2000ms.

```
root@ssh-client1:/home/adrian# ping 172.16.109.216
PING 172.16.109.216 (172.16.109.216) 56(84) bytes of data.
64 bytes from 172.16.109.216: icmp_req=1 ttl=128 time=1.16 ms
64 bytes from 172.16.109.216: icmp_req=2 ttl=128 time=0.625 ms
64 bytes from 172.16.109.216: icmp_req=3 ttl=128 time=0.472 ms
^C
--- 172.16.109.216 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.472/0.755/1.168/0.298 ms
root@ssh-client1:/home/adrian#
```

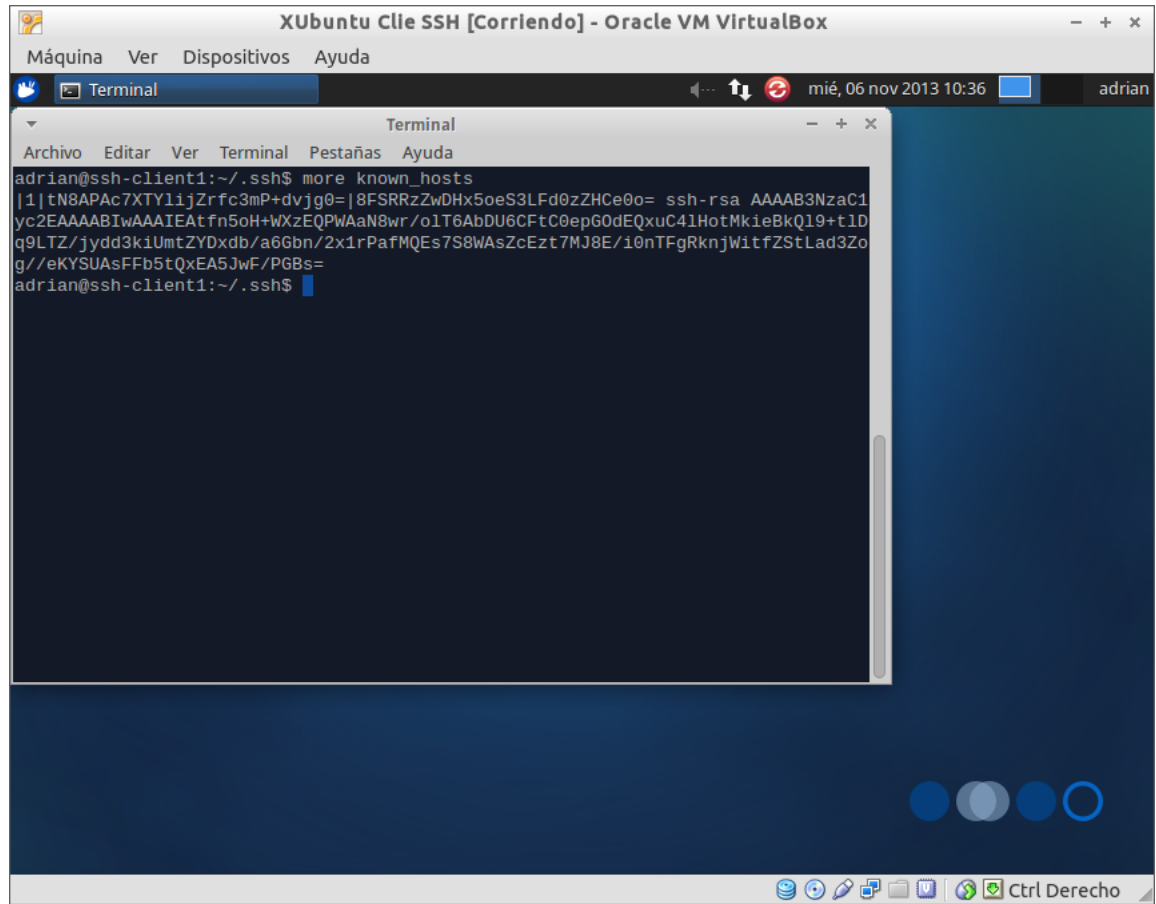
Probamos conectarnos al servidor mediante SSH:



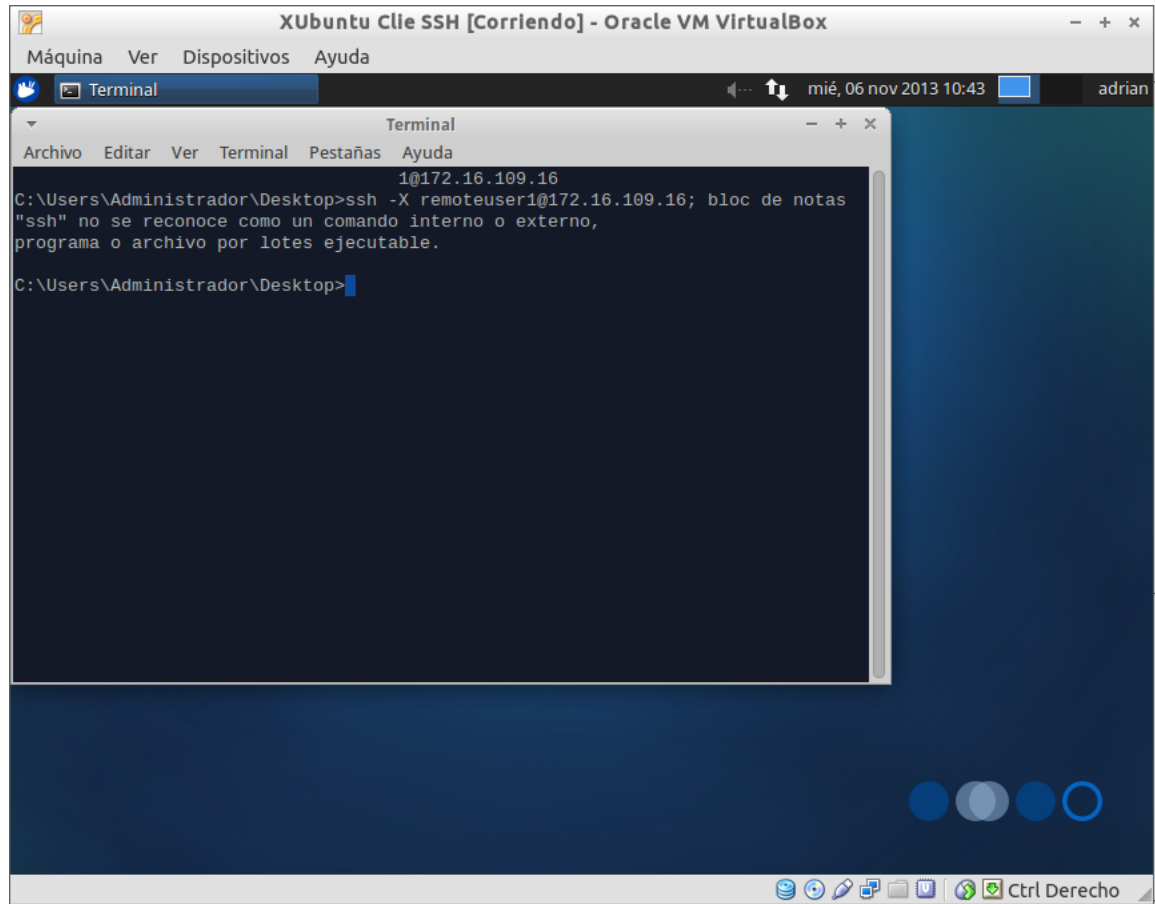
Conectados de manera correcta al servidor Windows 2008 Server:



Capturar la clave intercambiada en el primer proceso de conexión SSH:

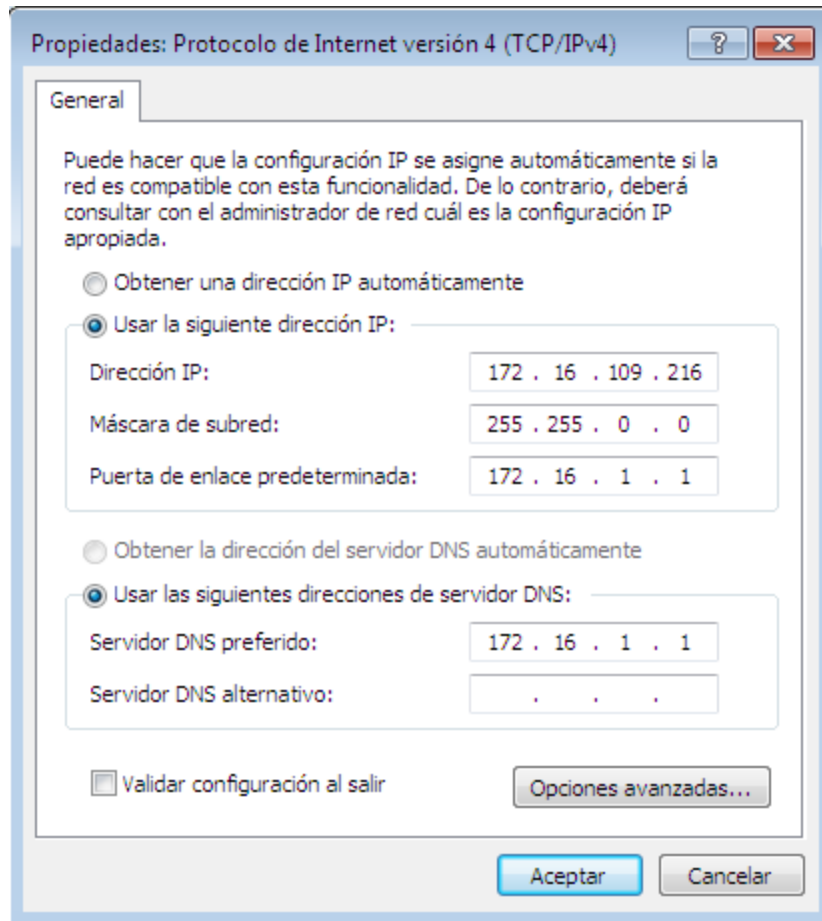


Probamos hacer los demás pasos llegados a este etapa como por ejemplo el del punto 5 (Uso de SSH como túnel para X), pero nos da error ya que no se pueden realizar con un servidor windows:

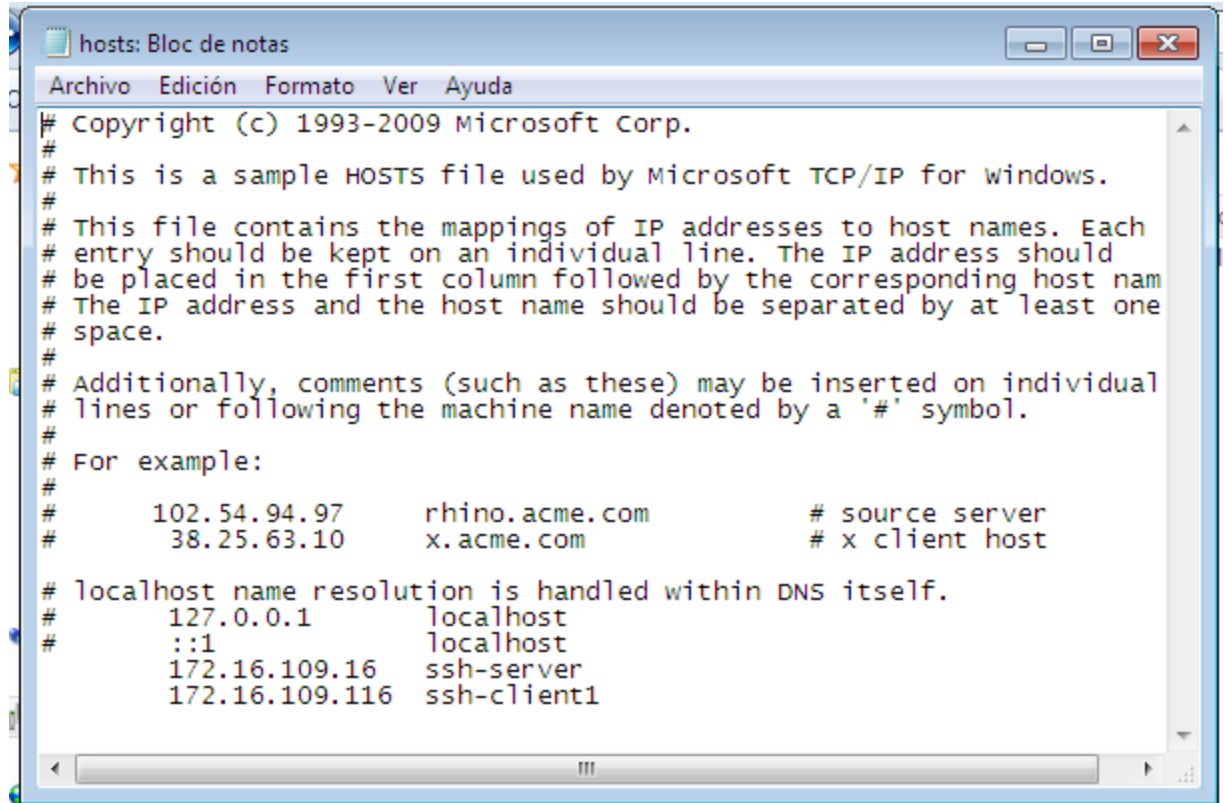


Cliente 2 (Windows 7):

IP's del cliente2 Windows 7:

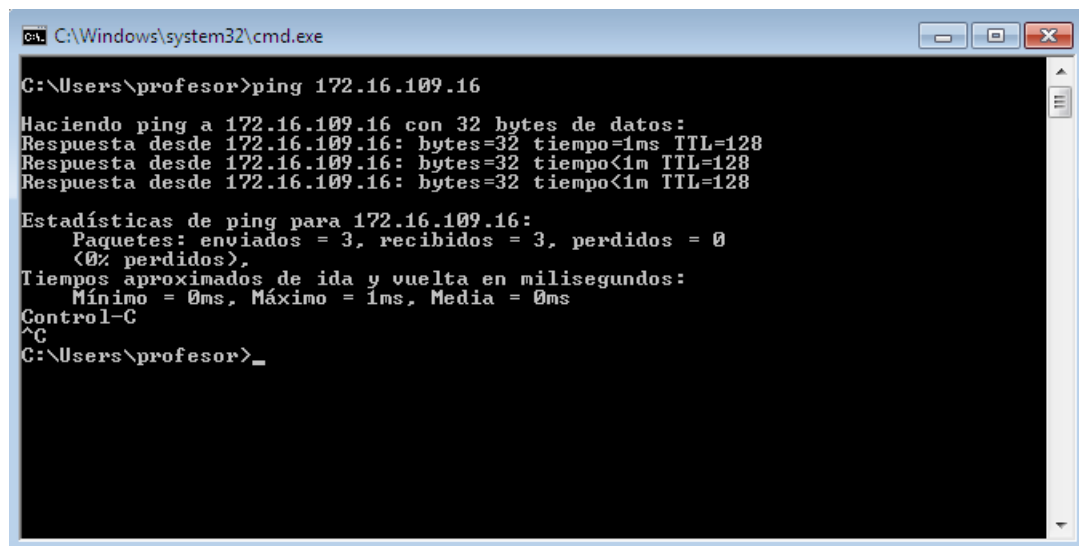


Añadimos en /etc/hosts el equipo ssh-server y ssh-client1:



```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host nam
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com           # source server
#       38.25.63.10       x.acme.com               # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
#       172.16.109.16     ssh-server
#       172.16.109.116    ssh-client1
```

Probamos hacer ping al servidor:



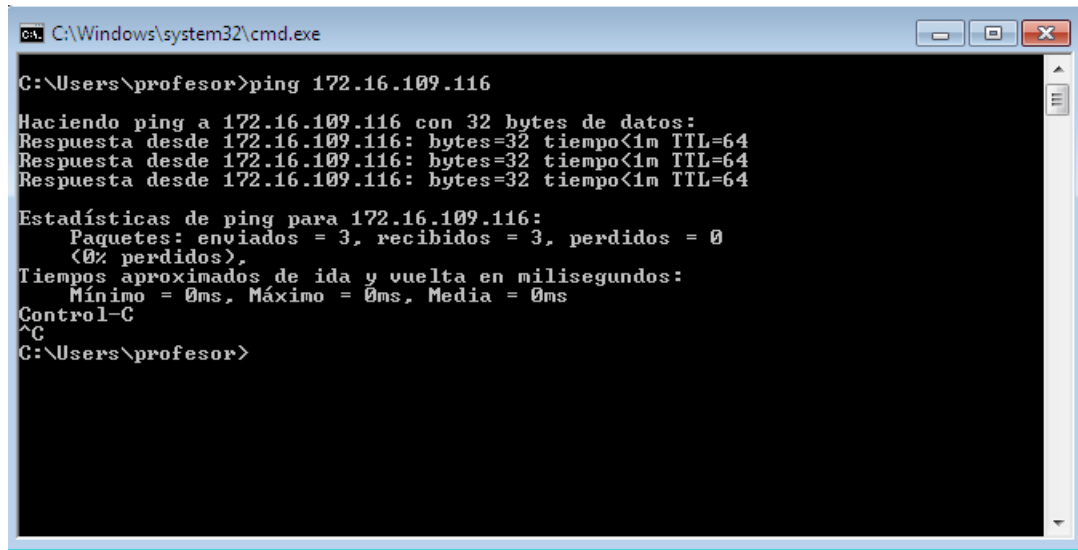
```
C:\Windows\system32\cmd.exe

C:\Users\profesor>ping 172.16.109.16

Haciendo ping a 172.16.109.16 con 32 bytes de datos:
Respuesta desde 172.16.109.16: bytes=32 tiempo=1ms TTL=128
Respuesta desde 172.16.109.16: bytes=32 tiempo<1m TTL=128
Respuesta desde 172.16.109.16: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 172.16.109.16:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
            Mínimo = 0ms, Máximo = 1ms, Media = 0ms
Control-C
^C
C:\Users\profesor>_
```

Probamos hacer ping al cliente 1:



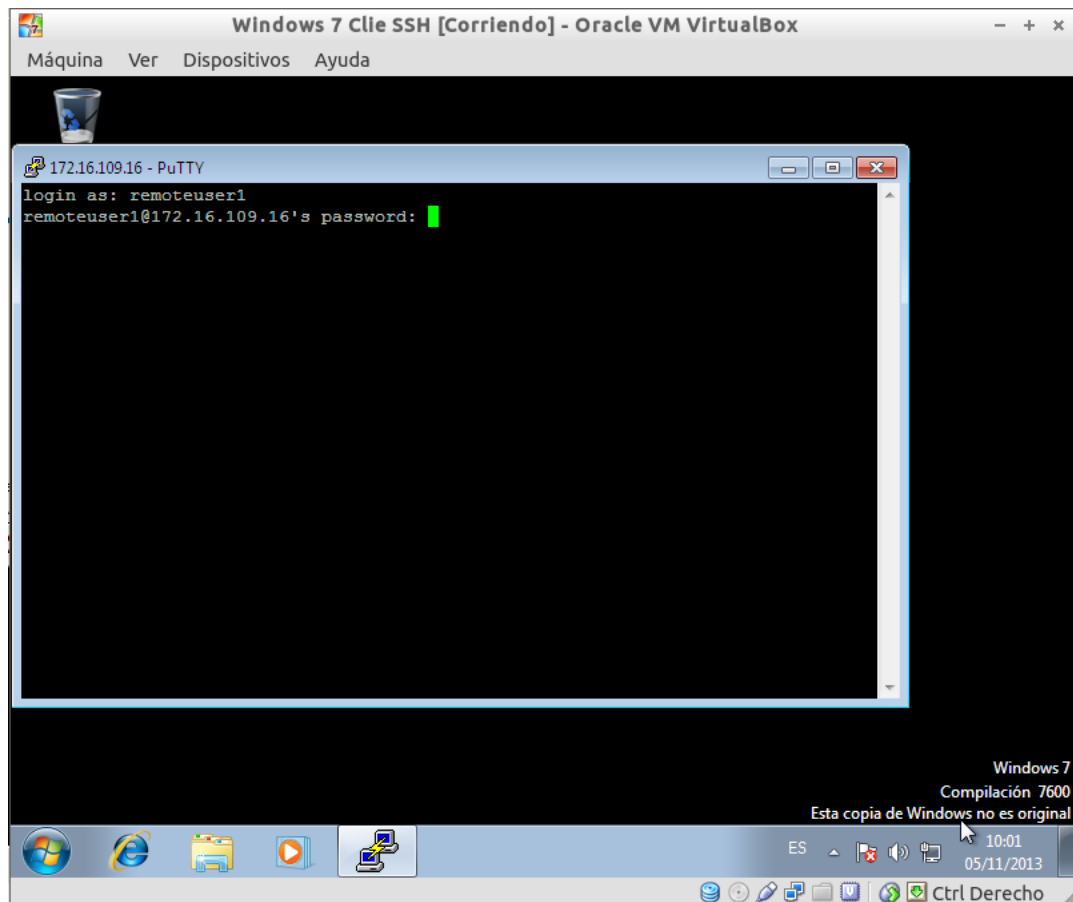
```
C:\Windows\system32\cmd.exe

C:\Users\profesor>ping 172.16.109.116

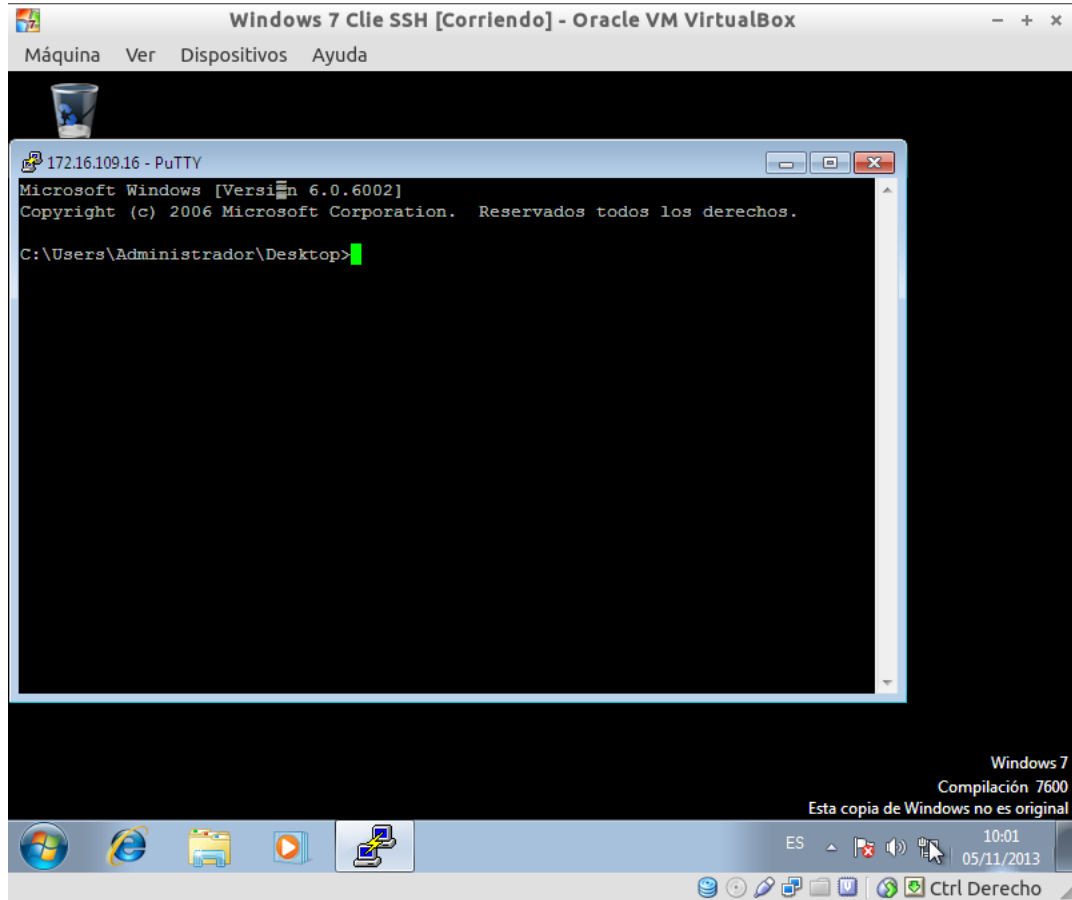
Haciendo ping a 172.16.109.116 con 32 bytes de datos:
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64
Respuesta desde 172.16.109.116: bytes=32 tiempo<1m TTL=64

Estadísticas de ping para 172.16.109.116:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
Control-C
^C
C:\Users\profesor>
```

Probamos conectarnos al servidor mediante SSH:



Conectados de manera correcta al servidor Windows 2008 Server:

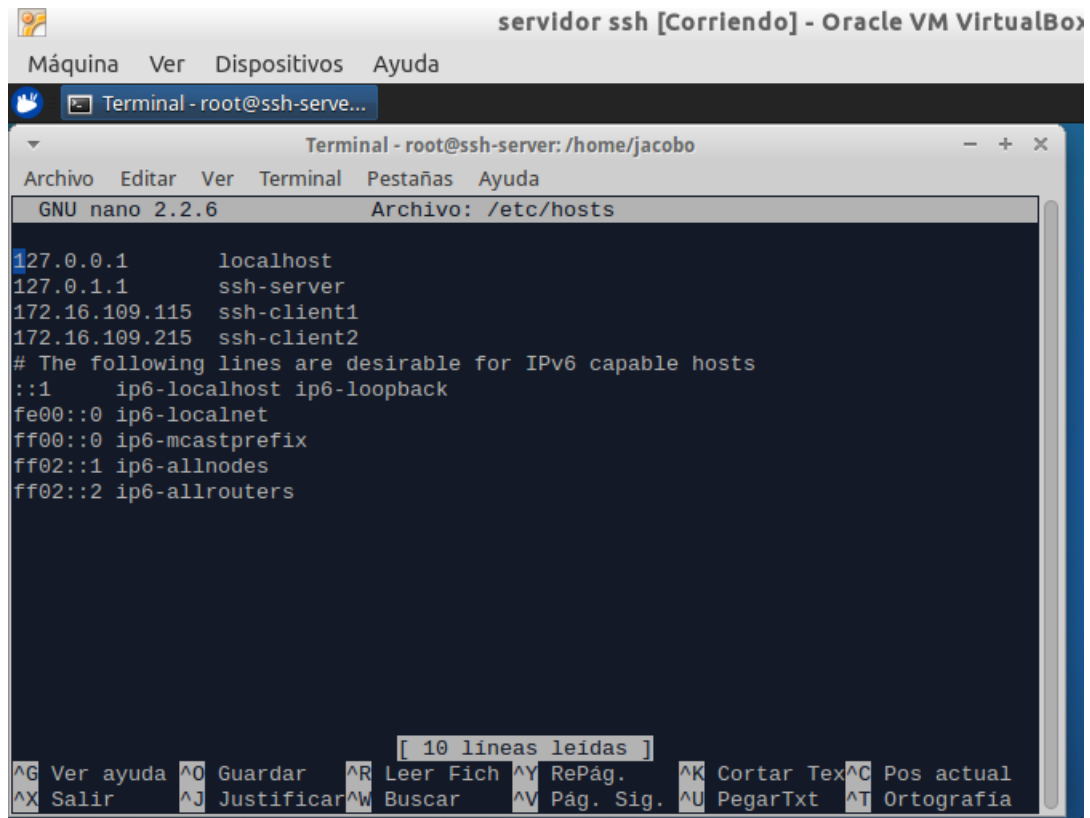


Servidor Linux

- Servidor: Xubuntu.
- Cliente 1: Xubuntu.
- Cliente 2: Windows XP.

Instalación básica

Configurar el servidor ssh:



servidor ssh [Corriendo] - Oracle VM VirtualBox

Máquina Ver Dispositivos Ayuda

Terminal - root@ssh-serve...

Terminal - root@ssh-server: /home/jacobo

Archivo Editar Ver Terminal Pestañas Ayuda

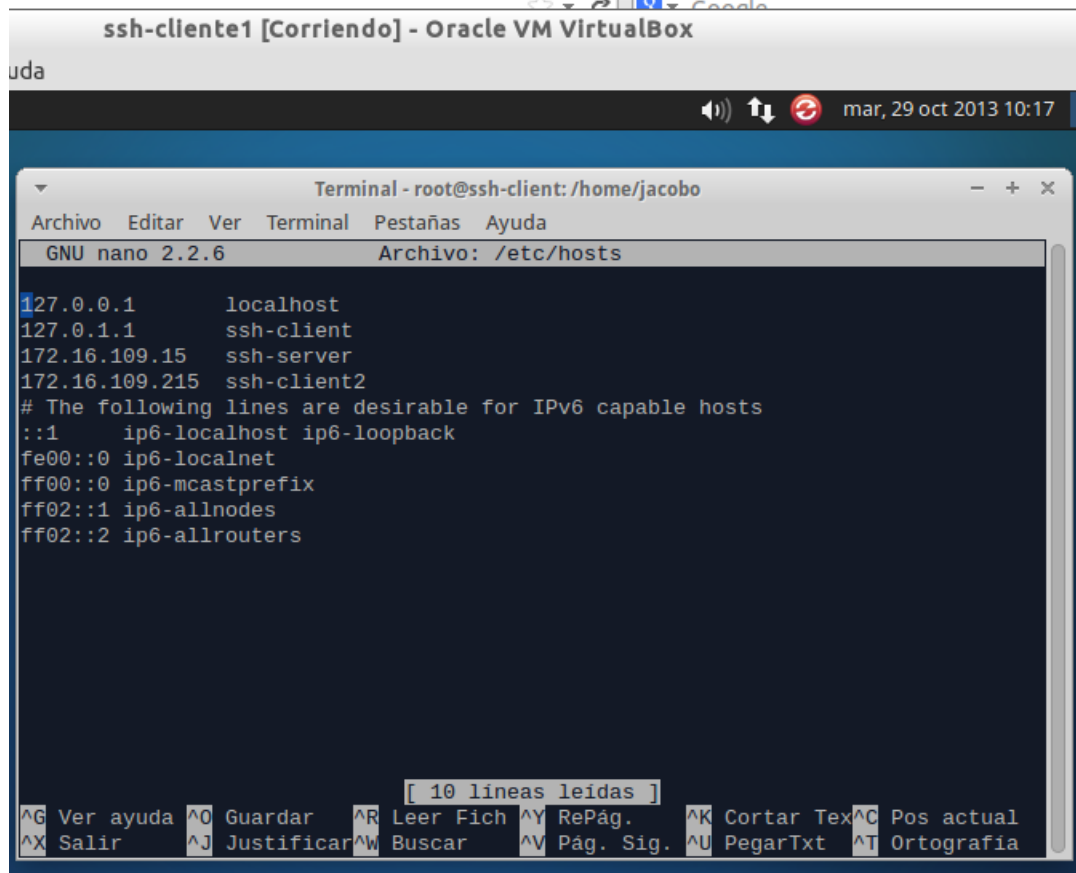
GNU nano 2.2.6 Archivo: /etc/hosts

```
127.0.0.1    localhost
127.0.1.1    ssh-server
172.16.109.115 ssh-client1
172.16.109.215 ssh-client2
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

[10 líneas leídas]

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía

Configurar el cliente1 (Xubuntu) ssh:



The screenshot shows a terminal window titled "ssh-client1 [Corriendo] - Oracle VM VirtualBox". The terminal is running a nano editor to edit the file "/etc/hosts". The content of the file is as follows:

```
127.0.0.1    localhost
127.0.1.1    ssh-client
172.16.109.15 ssh-server
172.16.109.215 ssh-client2
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

The terminal window also shows a status bar at the bottom with various keyboard shortcuts and a message "[10 líneas leídas]".

En Windows lo único que tenemos que hacer para configurar el ssh es instalar la aplicación putty.

Personalización del prompt Bash

Para configurar el prompt bash tenemos que dirigirnos al servidor ssh y escribir la siguiente línea de comandos:

```

Terminal - root@ssh-server: /home/jacobo
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
GNU nano 2.2.6  Archivo: /home/remoteuser1/.bashrc

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

if [ -n "SSH-CLIENT" ]; then text=" ssh"
fi
export PS1='\[\e[0;31m\]\u@h:\w${text}$\[\e[m\]'

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág. Sig. ^U PegarTxt ^T Ortografía

```

Ahora nos vamos al cliente Xubuntu y nos conectamos mediante ssh al servidor y vemos que el prompt ha cambiado el color a rojo:

```

Terminal - root@ssh-client: /home/jacobo
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
jacobos@ssh-client:~$ sudo su
[sudo] password for jacobos:
root@ssh-client:/home/jacobo# sshr remoteuser1@ssh-server
No se ha encontrado la orden «sshr», quizás quiso decir:
  La orden «sshm» del paquete «sshm» (universe)
  La orden «ssh» del paquete «openssh-client» (main)
  La orden «sshd» del paquete «openssh-server» (main)
sshr: no se encontró la orden
root@ssh-client:/home/jacobo# ssh remoteuser1@ssh-server
remoteuser1@ssh-server's password:
Welcome to Ubuntu 13.04 (GNU/Linux 3.8.0-19-generic i686)

 * Documentation:  https://help.ubuntu.com/

New release '13.10' available.
Run 'do-release-upgrade' to upgrade to it.

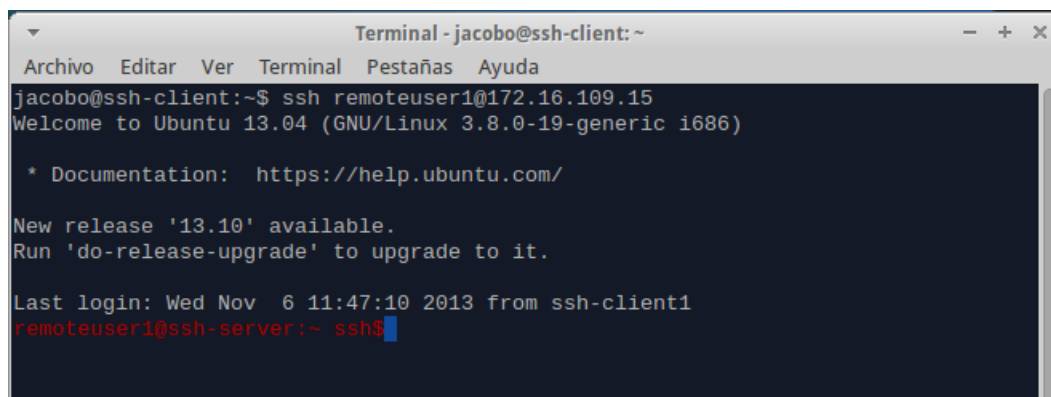
Last login: Tue Oct 29 10:30:46 2013 from ssh-client1
remoteuser1@ssh-server:~ ssh$

```

Autenticación mediante claves públicas


```
root@ssh-server:/home/jacobo# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
57:90:f0:69:09:f3:9d:1a:6f:1c:38:6b:18:63:53:d0 root@ssh-server
The key's randomart image is:
+--[ RSA 2048 ]-----+
|      =+o.      |
|      *E*      |
|      = X =     |
|      . * 0 .   |
|      S = +     |
|      o .       |
|                |
+-----+
root@ssh-server:/home/jacobo#
```

Modificamos distintos archivos del servidor para que nuestros clientes sean autenticados por claves públicas y no nos pida una contraseña a la hora de conectarnos:



```
Terminal - jacobo@ssh-client: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
jacobo@ssh-client:~$ ssh remoteuser1@172.16.109.15
Welcome to Ubuntu 13.04 (GNU/Linux 3.8.0-19-generic i686)

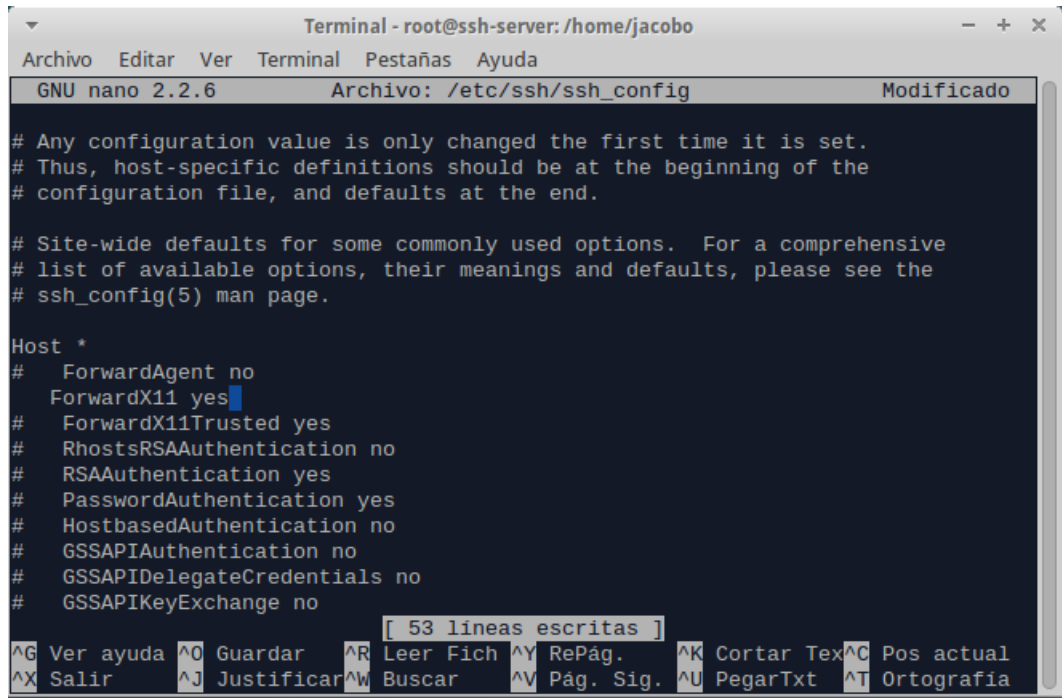
 * Documentation:  https://help.ubuntu.com/

New release '13.10' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Nov  6 11:47:10 2013 from ssh-client1
remoteuser1@ssh-server:~ ssh$
```

Uso de SSH como túnel para X

Añadimos una línea más al archivo de configuración ssh-config como se ve en la siguiente imagen:



```
Terminal - root@ssh-server: /home/jacobo
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
GNU nano 2.2.6      Archivo: /etc/ssh/ssh_config      Modificado

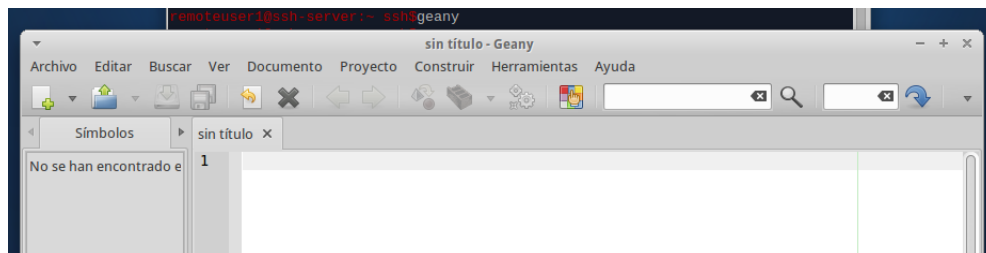
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for some commonly used options.  For a comprehensive
# list of available options, their meanings and defaults, please see the
# ssh_config(5) man page.

Host *
#   ForwardAgent no
#   ForwardX11 yes
#   ForwardX11Trusted yes
#   RhostsRSAAuthentication no
#   RSAAuthentication yes
#   PasswordAuthentication yes
#   HostbasedAuthentication no
#   GSSAPIAuthentication no
#   GSSAPIDelegateCredentials no
#   GSSAPIKeyExchange no

[ 53 líneas escritas ]
^G Ver ayuda  ^O Guardar  ^R Leer Fich  ^Y RePág.  ^K Cortar Tex  ^C Pos actual
^X Salir      ^J Justificar  ^W Buscar    ^V Pág. Sig.  ^U PegarTxt   ^T Ortografia
```

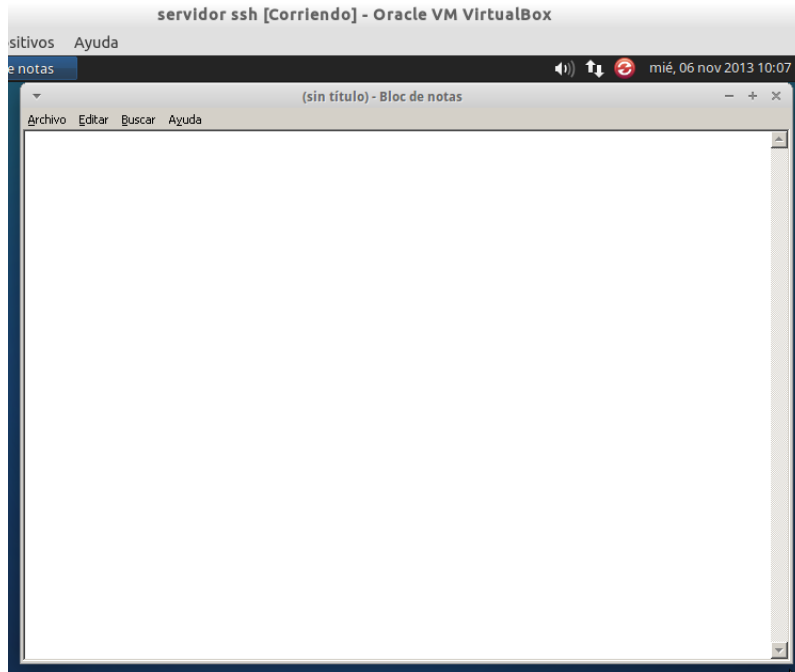
Probamos una aplicación que está instalada en el servidor pero no en el cliente, para ello nos conectamos mediante ssh al servidor y escribimos el siguiente comando `ssh -X remoteuser1@ssh-server; geany`:



Aplicaciones Windows nativas

Nos conectamos con el cliente linux al servidor y ejecutamos el notepad:

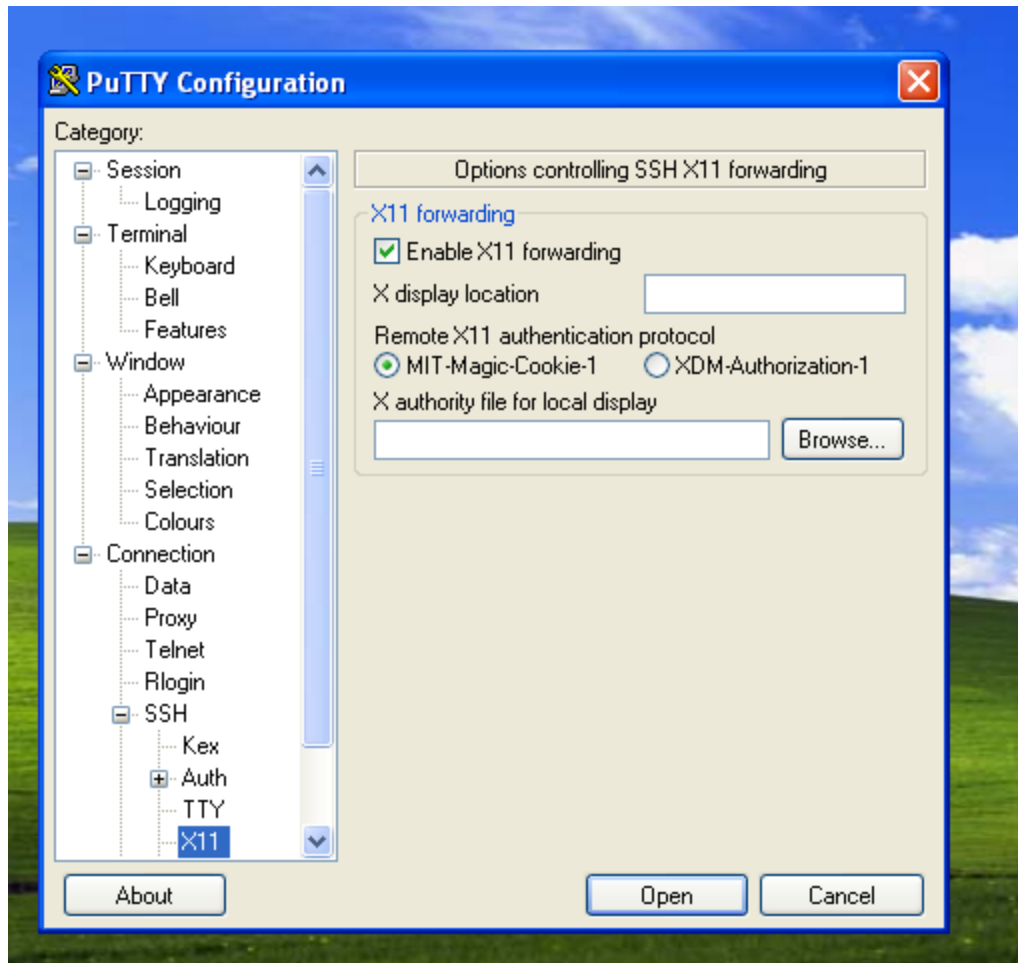
Para poder ejecutar aplicaciones de windows nativas debemos instalar el wine:



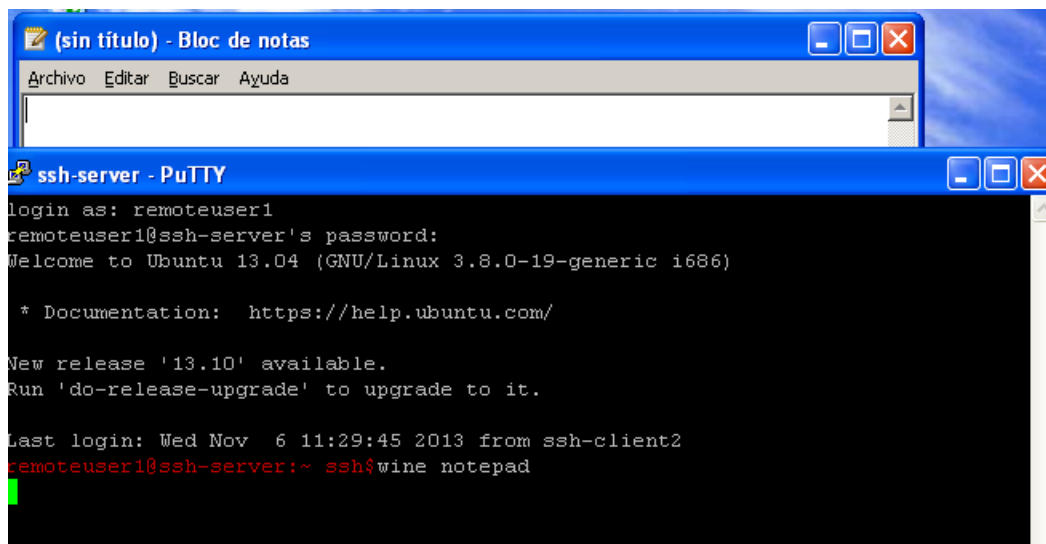
Para poder ejecutar aplicaciones de windows nativas desde un cliente windows tenemos que instalar el programa xming:



Una vez instalado el programa anterior configuramos el putty como se ve en la siguiente imagen:



Aquí vemos funcionando el notepad por ssh:



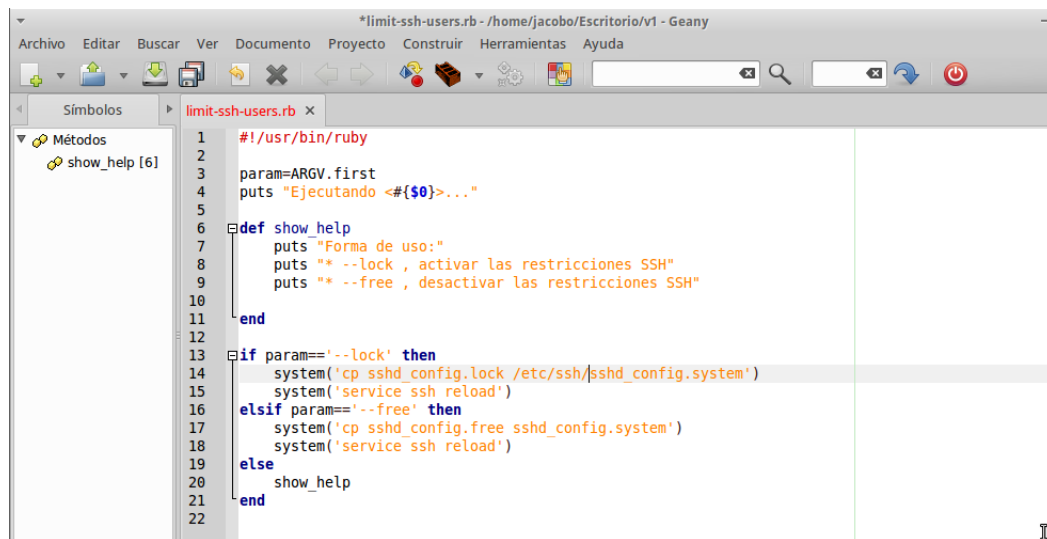
Restricciones de uso

La restricción de tipo 1 no hay que hacer nada.

Para aplicar la restricción tipo 2, tenemos que dirigirnos al archivo `/etc/sshd_config` al final añadimos la línea `DenyUsers remoteuser2` y queda aplicar la restricción para que dicho usuario no pueda conectarse por ssh:

```
jacobo@ssh-client:~$ ssh remoteuser2@172.16.109.15
remoteuser2@172.16.109.15's password:
Permission denied, please try again.
```

Para aplicar la restricción tipo 3, tenemos que modificar el siguiente script:



```
*limit-ssh-users.rb - /home/jacobo/Escritorio/v1 - Geany
Archivo  Editar  Buscar  Ver  Documento  Proyecto  Construir  Herramientas  Ayuda

limit-ssh-users.rb x
Métodos
  show_help [6]

1  #!/usr/bin/ruby
2
3  param=ARGV.first
4  puts "Ejecutando <#{$0}>..."
5
6  def show_help
7    puts "Forma de uso:"
8    puts "** --lock , activar las restricciones SSH"
9    puts "** --free , desactivar las restricciones SSH"
10 end
11
12
13 if param=='--lock' then
14   system('cp sshd_config.lock /etc/ssh/sshd_config.system')
15   system('service ssh reload')
16 elsif param=='--free' then
17   system('cp sshd_config.free sshd_config.system')
18   system('service ssh reload')
19 else
20   show_help
21 end
22
```

Lo ejecutamos pero nos dará error, ya que tenemos que usar una de las variables que se nos indica (`--lock`):

```
root@ssh-server:/home/jacobo/Escritorio/v1# ./limit-ssh-users.rb
```

Los escribiremos de manera correcta `./limit-ssh-users.rb --lock`:

```
root@ssh-server:/home/jacobo/Escritorio/v1# ./limit-ssh-users.rb
```

Y lo ejecutaremos:

```
root@ssh-server:/home/jacobo/Escritorio/v1# ./limit-ssh-users.rb --lock
Ejecutando <./limit-ssh-users.rb>...
root@ssh-server:/home/jacobo/Escritorio/v1#
```