

TAREA 10 – ADMINISTRACIÓN DE REDES GNU/LINUX.

CONTENIDO

PREVIAMENTE	2
Actividad 1. Configuración de la Red	3
Actividad 2. Servicio SSH. Simétrico	7
Actividad 3. Servicio SSH. Asimétrico	12
Actividad 4. Servidor Web.....	15
Actividad 5. VS Code + SSH + Apache2.....	18
Recursos	21
Criterios de Corrección	21

PREVIAMENTE

Es importante que tengáis claro como vais a configurar los adaptadores de red en vuestro hipervisor para cada una de las maquinas.

En Ubuntu Server la configuración de la red se realiza a través de la herramienta `netplan`. No dudéis en preguntar.

En **server01** (Ubuntu Sever 22.04):

- 1 adaptador de red – Adaptador puente (IP Dinámica)
- 2 adaptador de red – Red interna (IP **192.168.10.1/24**)
- Crea el usuario `tu-nick`¹. Añádalo al grupo “Administradores” (sudoers) del equipo.
- Instala el servicio ssh y confirma que este operativo.

En **cliente01**: (Ubuntu Desktop 22.04)

- 1 adaptador de red – Red interna (IP **192.168.10.2/24**)
- Crea el usuario `tu-nombre-pila`. Añádalo al grupo “Administradores” (sudoers) del equipo.

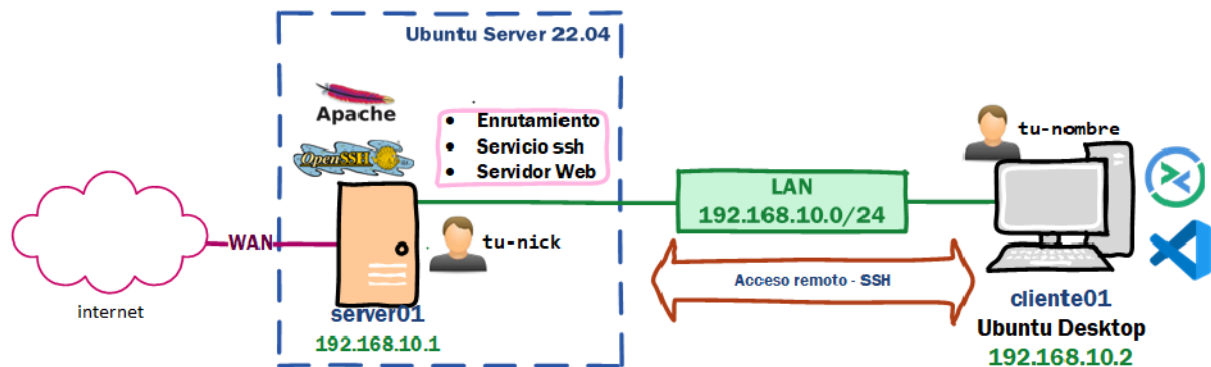
NOTA: Configurar la subred, las IPs y los usuarios **ESTRICTAMENTE** tal y como se indica.

NOTA: No es necesario documentar este aparatado. Se entiende que lo sabéis hacer y se confía en que lo hagáis correctamente para el buen desarrollo de la tarea.

¹ Corresponde con tu nick de educantabria, en mi caso es fcuadradoa01

ACTIVIDAD 1. CONFIGURACIÓN DE LA RED

La tarea que nos atañe está basada en el diagrama de red adjunto.



1. Confirma que **server01** tiene la IP especificada (LAN) y tiene acceso a internet.

Configuramos el fichero netplan, a través del editor de consola nano y nos debe de quedar de la siguiente forma:

```
GNU nano 6.2 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernet:
    ens33:
      dhcp4: true
    ens34:
      addresses:
        - 192.168.10.1/24
  version: 2
```

Ejecutamos el comando netplan apply para que aplique la nueva configuración y ya con mediante el comando ip a podemos ver como queda configurada las interfaces.

```
instalador@cabuerniga:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:19:d4:7c brd ff:ff:ff:ff:ff:ff
    altnames enp2s1
    inet 192.168.128.128/24 metric 100 brd 192.168.128.255 scope global dynamic ens33
        valid_lft 1798sec preferred_lft 1798sec
    inet6 fe80::20c:29ff:fe19:d47c/64 scope link
        valid_lft forever preferred_lft forever
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:19:d4:86 brd ff:ff:ff:ff:ff:ff
    altnames enp2s2
    inet 192.168.10.1/24 brd 192.168.10.255 scope global ens34
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe19:d486/64 scope link
        valid_lft forever preferred_lft forever
instalador@cabuerniga:~$
```

La primera interfaz ens33 es la que tiene acceso a internet y la segunda ens34 esta en una red interna con la ip que nos indica el diagrama. Si además hacemos un ping www.google.es

```

instalador@cabuerniga:~$ ping www.google.es
PING www.google.es (142.250.200.131) 56(84) bytes of data.
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=1 ttl=128 time=16.3 ms
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=2 ttl=128 time=16.0 ms
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=3 ttl=128 time=15.8 ms
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=4 ttl=128 time=20.7 ms
^C
--- www.google.es ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 15.770/17.213/20.714/2.030 ms
instalador@cabuerniga:~$

```

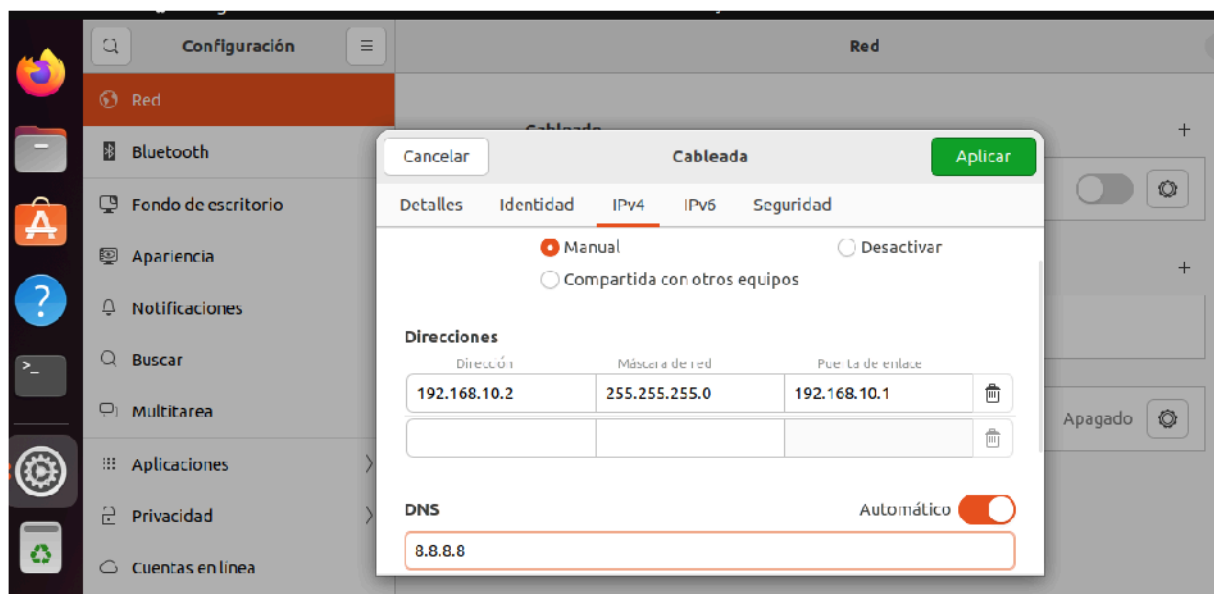
Para confirmar que nuestra puerta de enlace en la interfaz ens33 ejecutamos el comando `ip route show`

```

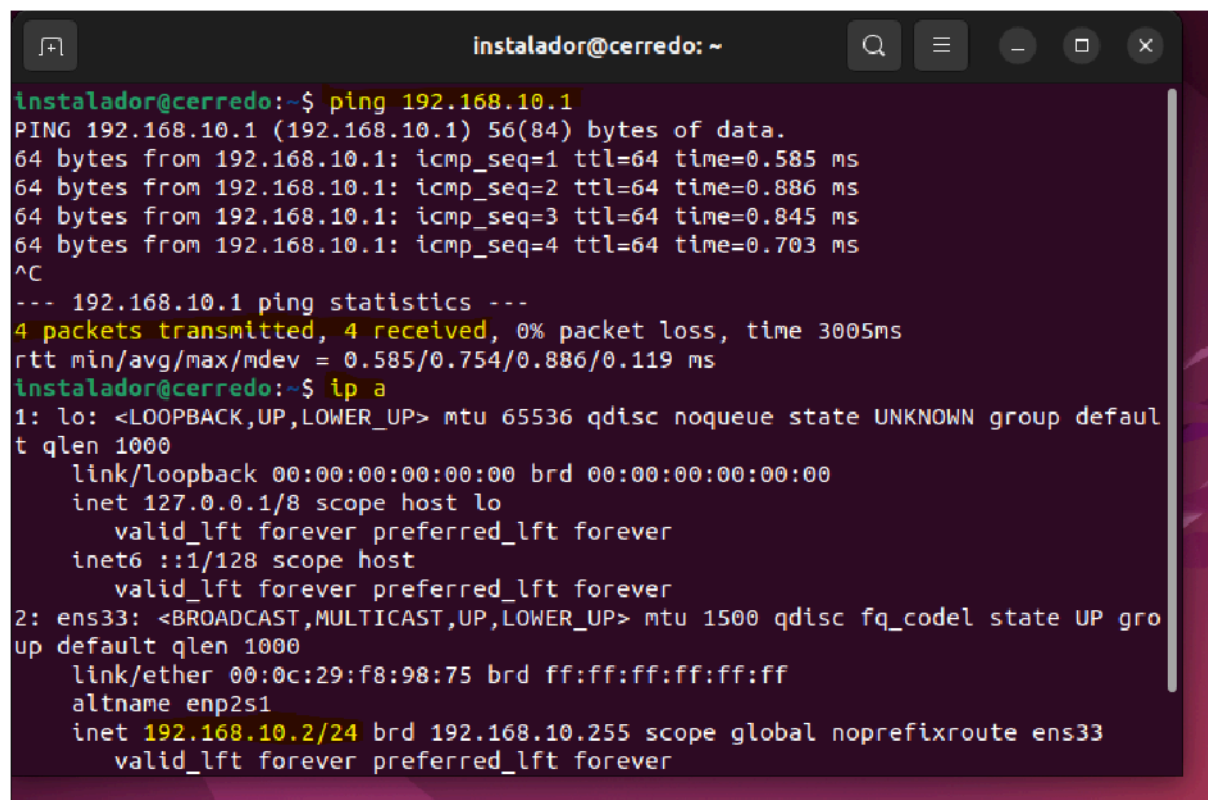
instalador@cabuerniga:~$ ip route show
default via 192.168.128.2 dev ens33 proto dhcp src 192.168.128.128 metric 100
192.168.10.0/24 dev ens34 proto kernel scope link src 192.168.10.1
192.168.128.0/24 dev ens33 proto kernel scope link src 192.168.128.128 metric 100
192.168.128.2 dev ens33 proto dhcp scope link src 192.168.128.128 metric 100
instalador@cabuerniga:~$

```

La IP del cliente la configuramos de forma gráfica. Y confirmamos que ambas maquinas tienen conectividad entre ellas:



Y un ping entre maquinas...

A terminal window titled 'instalador@cerredo: ~' with standard window controls. It shows the execution of 'ping 192.168.10.1' and 'ip a'. The ping command shows four successful packets with varying round-trip times. The 'ip a' command shows details for the loopback interface 'lo' and the ethernet interface 'ens33'.

```
instalador@cerredo:~$ ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=0.585 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=0.886 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=64 time=0.845 ms
64 bytes from 192.168.10.1: icmp_seq=4 ttl=64 time=0.703 ms
^C
--- 192.168.10.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.585/0.754/0.886/0.119 ms
instalador@cerredo:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:f8:98:75 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.10.2/24 brd 192.168.10.255 scope global noprefixroute ens33
        valid_lft forever preferred_lft forever
```

Aunque lo piden en el apartado siguiente, la puerta de enlace del cliente01 será la IP del server01, que es la máquina que le permite salir a internet. El DNS tenemos varias opciones, por simplificar dejamos el 8.8.8.8

2. Habilita enrutamiento en **server01** para que el **cliente01** tenga acceso a internet a través de él. Es decir, **server01** funcione como la puerta de enlace de **cliente01**. Decide tú que DNS configurar en **cliente01**.

En primer lugar, debemos permitir que server01 permita hacer **forward**, es decir, dejar pasar paquetes y conexiones a través de él y que no son para él.

Editamos el fichero: **/etc/sysctl.conf** y descomentamos la línea en cuestión:

```

GNU nano 6.2 /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#

#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lun.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

[ Wrote 68 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste       ^J Justify    ^_ Go To Line  M-E Redo

```

Ahora habilitamos NAT en la interfaz que funciona como WAN, la que tiene acceso a internet, en nuestro caso es **ens33**

```
sudo iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE
```

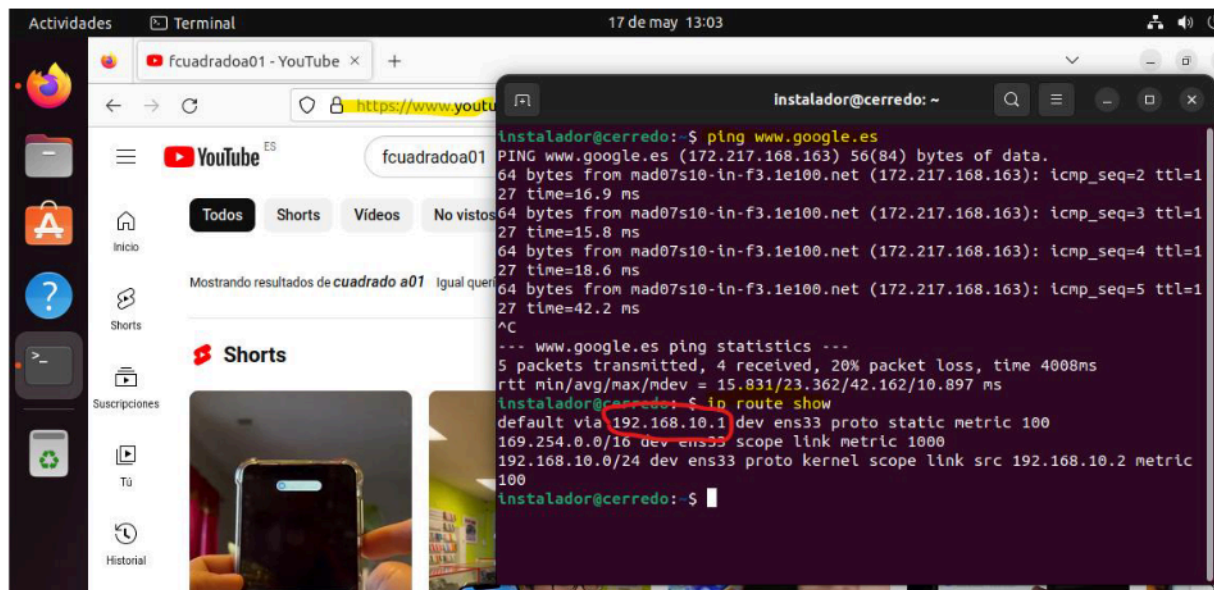
```

instalador@cabuerniga:~$ sudo iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE
[sudo] password for instalador:
instalador@cabuerniga:~$ _

```

Y con esto ya es suficiente. Existen más configuraciones posibles, pero no nos interesa ahondar más.

3. Confirma que **cliente01 tiene la IP especificada (LAN) y tiene acceso a internet.**



4. Confirma que ambas máquinas están en la misma red haciendo un ping entre ellas.

Ya está realizado.

ACTIVIDAD 2. SERVICIO SSH. SIMÉTRICO

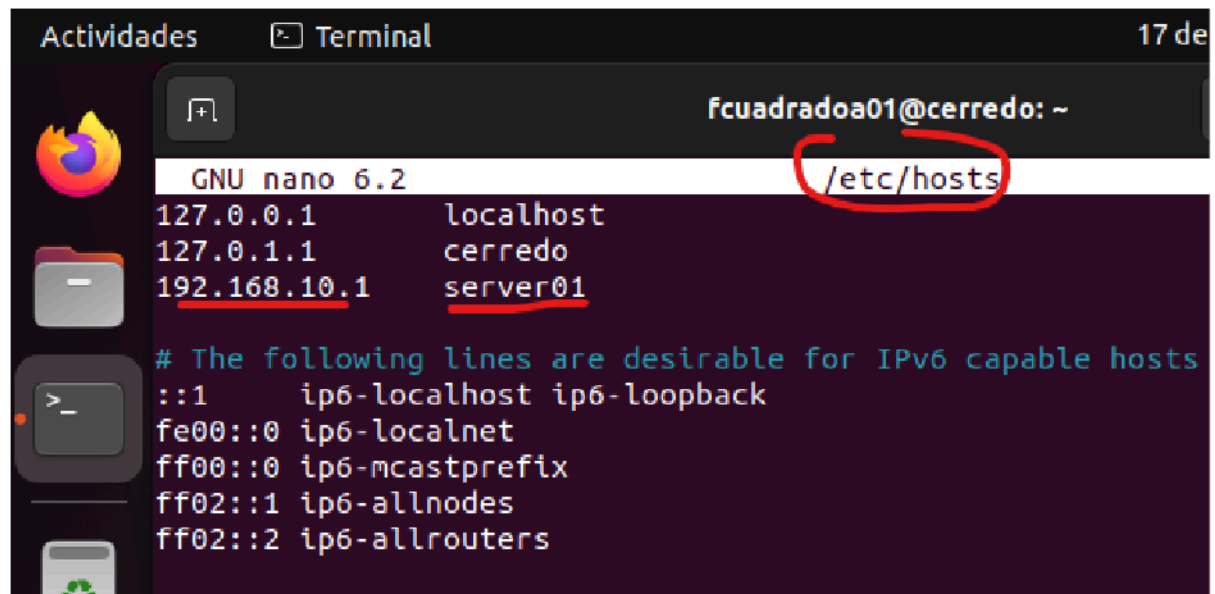
Desde **cliente01**. Iniciamos sesión con el usuario **tu-nombre-de-pila**.

Abrimos la terminal (consola/bash):

1. Nos conectamos remotamente (ssh) a **server01** con el usuario **tu-nick**².

Para no estar escribiendo la ip del servidor constantemente, añadimos una entrada al fichero que contiene el DNS local: **/etc/hosts**

² Es importante que se diferencie entre el usuario con el que estoy trabajando en el cliente y el usuario con el que me voy a conectar al servidor, que debe de ser un usuario local del servidor.



The screenshot shows a terminal window titled 'Actividades' with a 'Terminal' tab. The user is logged in as 'fcuadradoa01@cerredo: ~'. The terminal displays the contents of the `/etc/hosts` file, which is being edited with GNU nano 6.2. The file contains the following entries:

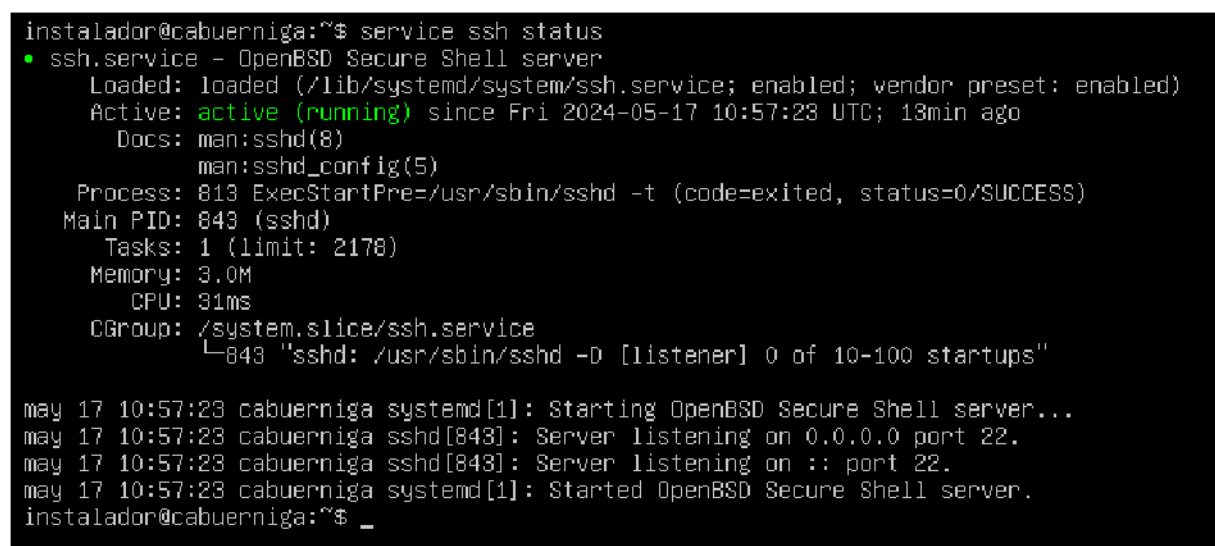
```
127.0.0.1    localhost
127.0.1.1    cerredo
192.168.10.1  server01

# 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 path `/etc/hosts` is circled in red in the original image.

Y añadimos la entrada como se muestra en la imagen anterior.

En el servidor, confirmamos que el servicio ssh está corriendo:



The screenshot shows a terminal window with the following output:

```
instalador@cabuerniga:~$ service ssh status
• ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2024-05-17 10:57:23 UTC; 13min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Process: 813 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 Main PID: 843 (sshd)
   Tasks: 1 (limit: 2178)
  Memory: 3.0M
    CPU: 31ms
  CGroup: /system.slice/ssh.service
          └─843 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

may 17 10:57:23 cabuerniga systemd[1]: Starting OpenBSD Secure Shell server...
may 17 10:57:23 cabuerniga sshd[843]: Server listening on 0.0.0.0 port 22.
may 17 10:57:23 cabuerniga sshd[843]: Server listening on :: port 22.
may 17 10:57:23 cabuerniga systemd[1]: Started OpenBSD Secure Shell server.
instalador@cabuerniga:~$ _
```

Ahora conectarnos vía ssh es muy fácil:


```
fcuadradoa01@cerredo:~$ ssh instalador@server01
instalador@server01's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of vie 17 may 2024 11:21:27 UTC

System load:  0.0               Processes:            209
Usage of /:   29.0% of 18.53GB   Users logged in:     1
Memory usage: 18%              IPv4 address for ens33: 192.168.128.128
Swap usage:   0%               IPv4 address for ens34: 192.168.10.1
```

2. Creamos una carpeta y algún archivo, confirmando que estamos trabajando sobre **server01**.

```
instalador@cabuerniga:~$ mkdir bocarte
instalador@cabuerniga:~$ cd bocarte/
instalador@cabuerniga:~/bocarte$ touch muyricos.txt
instalador@cabuerniga:~/bocarte$ ls
muyricos.txt
instalador@cabuerniga:~/bocarte$
```

3. Cerramos la sesión remota.

Tecleamos **exit**

```
instalador@cabuerniga:~/bocarte$ exit
logout
Connection to server01 closed.
fcuadradoa01@cerredo:~$
```

4. Copia un archivo desde **cliente01** a **server01** utilizando el comando **scp**.

```
fcuadradoa01@cerredo:~$ touch sardinas.txt
fcuadradoa01@cerredo:~$ ls
Descargas  Escritorio  Música      Público      snap
Documentos Imágenes    Plantillas  sardinas.txt Videos
fcuadradoa01@cerredo:~$ scp sardinas.txt instalador@server01:bocarte
instalador@server01's password:
sardinas.txt                                100%   0    0.0KB/s   00:00
fcuadradoa01@cerredo:~$ █
```

Ahora si nos conectamos remotamente, confirmamos que se ha copiado el archivo en la carpeta:

```
instalador@cabuerniga:~$ cd bocarte/  
instalador@cabuerniga:~/bocarte$ ls  
muyricos.txt  sardinas.txt  
instalador@cabuerniga:~/bocarte$
```

Es **IMPORTANTE** tener claro que el comando **scp** copia archivos directamente al home del usuario que se conecta.

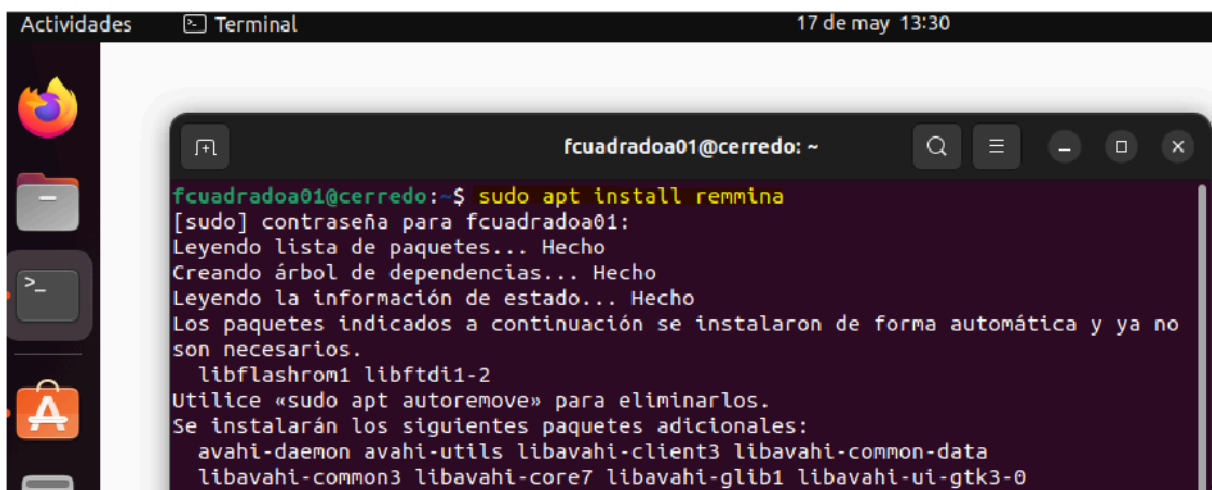
5. ¿Te puedes conectar por ssh desde tu equipo de trabajo a **server01**? ¿A qué IP te conectas?

Sí. A la IP de la WAN. La que me haya asignado el hipervisor.

Desde el entorno gráfico:

6. Instalamos el software **Remmina**.

Aunque se puede instalar directamente desde la “tienda” de Ubuntu, a nosotros nos encanta la terminal, así que:



```
Actividades Terminal 17 de may 13:30  
fcuadradoa01@cerredo: ~  
fcuadradoa01@cerredo:~$ sudo apt install remmina  
[sudo] contraseña para fcuadradoa01:  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias... Hecho  
Leyendo la información de estado... Hecho  
Los paquetes indicados a continuación se instalaron de forma automática y ya no  
son necesarios.  
  libflashrom1 libftdi1-2  
Utilice «sudo apt autoremove» para eliminarlos.  
Se instalarán los siguientes paquetes adicionales:  
  avahi-daemon avahi-utils libavahi-client3 libavahi-common-data  
  libavahi-common3 libavahi-core7 libavahi-glib1 libavahi-ui-gtk3-0
```

7. Desde Remmina, creamos una conexión ssh a **server01**, la guardamos y nos conectamos con el usuario **tu-nick** haciendo uso de ella.

Remote Connection Profile

Nombre:

Grupo:

Protocolo:

Básico | Avanzado | Behavior | Túnel SSH | Notes

Servidor:

Tipo de autenticación:

Nombre de usuario:

Contraseña de usuario:

Archivo de identidad SSH:

SSH certificate file:

Contraseña para desbloquear la clave privada:

OpenSSH command:

Cancelar | Guardar como predeterminado | Guardar | Conectar | Guardar y conectar

Y la nos conectamos:

```

server01
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of vie 17 may 2024 11:33:42 UTC

System load:  0.080078125      Processes:            212
Usage of /:   29.0% of 18.53GB Users logged in:       1
Memory usage: 18%              IPv4 address for ens33: 192.168.128.128
Swap usage:   0%               IPv4 address for ens34: 192.168.10.1

El mantenimiento de seguridad expandido para Applications está desactivado

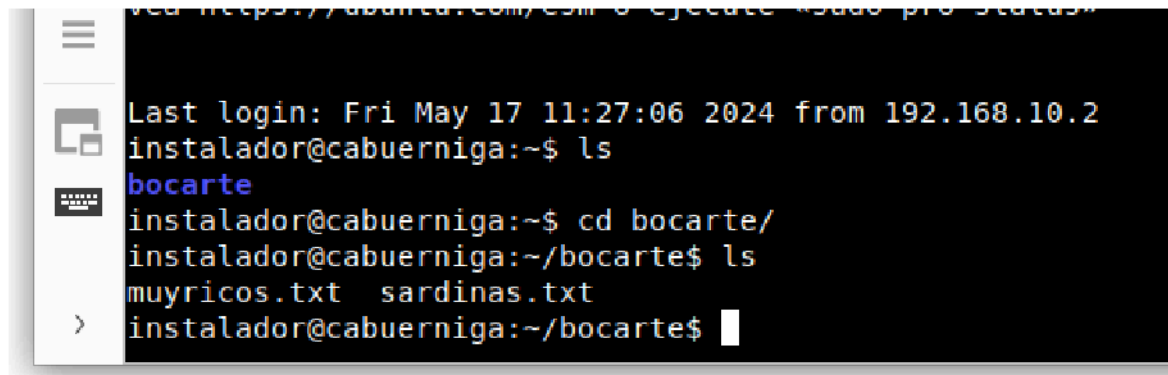
Se pueden aplicar 135 actualizaciones de forma inmediata.
85 de estas son actualizaciones de seguridad estándares.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

Last login: Fri May 17 11:27:06 2024 from 192.168.10.2
instalador@cabuerniga:~$

```

8. Confirmamos que los archivos y carpeta creados en el paso 2 están en **server01**.



```

Last login: Fri May 17 11:27:06 2024 from 192.168.10.2
instalador@cabuerniga:~$ ls
bocarte
instalador@cabuerniga:~$ cd bocarte/
instalador@cabuerniga:~/bocarte$ ls
muyricos.txt  sardinas.txt
instalador@cabuerniga:~/bocarte$

```

9. Cerramos la sesión remota.



Nombre	Grupo	Servidor	Complemento	Última utilización
 server01		server01	SSH	2024-05-17 - 13:35:02

ACTIVIDAD 3. SERVICIO SSH. ASIMÉTRICO

Desde **cliente01**. Iniciamos sesión con el usuario **tu-nombre-de-pila**.

1. Crea un juego de claves pública /privada. Copia la clave correspondiente al servidor ssh, asociándolo al usuario **tu-nick**. ¿Qué clave has copiado al servidor?

Escribimos el comando

```
$ ssh-keygen -t rsa
```

El parámetro `-t rsa` especifica el algoritmo utilizado por el juego de claves. RSA es el más común.

```

fcuadradoa01@cerredo:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/fcuadradoa01/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/fcuadradoa01/.ssh/id_rsa
Your public key has been saved in /home/fcuadradoa01/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:fz98eubV3QH8WUdIhB3ABu9QUHfSkfrEEzVoJ9IEWUC fcuadradoa01@cerredo
The key's randomart image is:
+---[RSA 3072]-----+
|      .+BOBEo|
|      +==*+.|
|      ..*==.o|
|      . o.o* +|
|      S   .o =|
|      o   .  =|
|      +   .. =|
|      .   .o =|
|      oB.|
+-----[SHA256]-----+
fcuadradoa01@cerredo:~$

```

Dejamos las opciones por defecto: la ubicación de los ficheros y si queremos securizarlo con una clave.

Confirmamos que las claves están:

```

fcuadradoa01@cerredo:~$ ls .ssh/ -l
total 16
-rw----- 1 fcuadradoa01 fcuadradoa01 2610 may 21 13:35 id_rsa
-rw-r--r-- 1 fcuadradoa01 fcuadradoa01  574 may 21 13:35 id_rsa.pub
-rw----- 1 fcuadradoa01 fcuadradoa01  978 may 17 13:21 known_hosts
-rw-r--r-- 1 fcuadradoa01 fcuadradoa01  142 may 17 13:21 known_hosts.old
fcuadradoa01@cerredo:~$

```

La llave pública lleva el .pub

2. Conéctate con ellas desde la terminal a `server01`.

Primero tenemos que copiar nuestra clave publica al servidor donde nos queremos conectar. Decir, que con este juego de llaves que es nuestro, nos podemos conectar a tantos servidores como queramos, sin más que copiarlos la clave publica, y mantener en nuestro poder la clave privada.

```

fcuadradoa01@cerredo:~$ ssh-copy-id instalador@server01
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
instalador@server01's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'instalador@server01'"
and check to make sure that only the key(s) you wanted were added.

fcuadradoa01@cerredo:~$

```

Como no hemos cambiado la ruta por defecto de las claves el comando ssh-copy-id sabe donde ir a buscarlas y su nombre. En caso contrario hubiese que haberlo especificado para decir al servidor que fichero contiene la clave pública que queremos copiarle.

Nos conectamos:

```
fcuadradoa01@cerredo:~$ ssh instalador@server01
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of mar 21 may 2024 11:41:54 UTC

System load:  0.0               Processes:           211
Usage of /:   29.0% of 18.53GB   Users logged in:    0
Memory usage: 18%              IPv4 address for ens33: 192.168.128.128
Swap usage:   0%               IPv4 address for ens34: 192.168.10.1

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 135 actualizaciones de forma inmediata.
85 de estas son actualizaciones de seguridad estándares.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

Last login: Tue May 21 11:30:54 2024 from 192.168.10.2
instalador@cabuerniga:~$
```

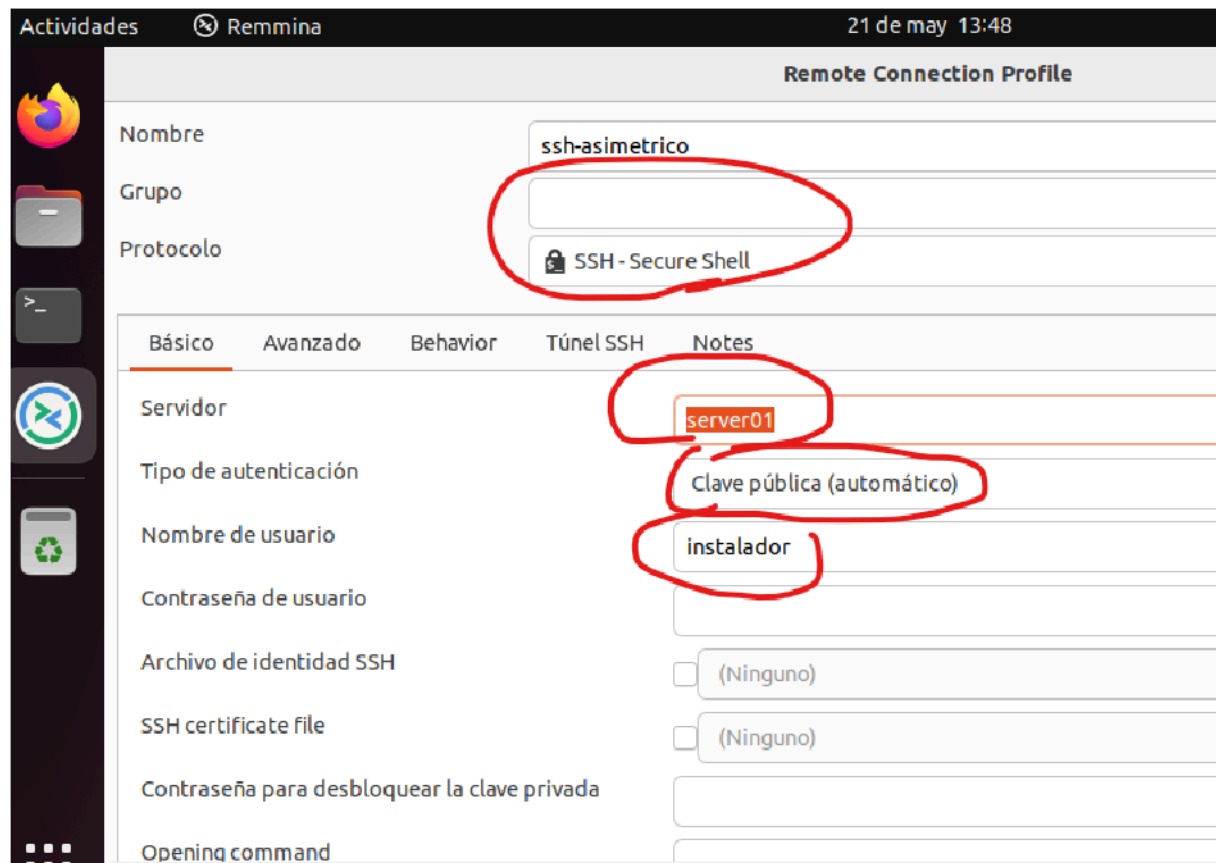
El servidor podemos confirmar que tenemos la clave publica copiada:

```
instalador@cabuerniga:~$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCxTxDTLY9f0A8Kvz30qFw8nkB8THN94JbjeYLKjTdw
IlpvdKl6cLJCUHWACUilG1I2gWCOZ0XkwoT7aKNXj9ZyA6qVDFHd0Em91Gf+15VYYSo6e/Fq8hR8NTzd
o27CZJV83tH3nQWHxvcWqQ026fYq1oab2l1wC+9Xar20vbhJSld3ovgFJBpP1Az1o0wgzqWV7SbzUR3
hh9mcjRtEXG4KmHaH6nNGKKOzmz4X1nzB0GpCBdyStvE4TuihILk7bDlLKPOlpx3sxMTw4ZRfkJo0NYD
p8HXOf2dp5FG09NSBngHhahfUhnR5koDCdsVTTV+yNg8wn6J4lqQ8HA0+T0M3aLMG33n2c+yrz5aGWh
YHNZP0ayqWqlsecLfY509//Lx28pXl98pVs7dMqv/LSpumJpEXPCx2GUB96RIINXhEBjhH/chS9itOp
A7XEFHNGEUNTuviiaJYdRHQGMN4Aqmun8srrW1AQDlSFw90iS7PoqWCLTrhVA06YDvpUHpE= fcuadra
doa01@cerredo
instalador@cabuerniga:~$
```

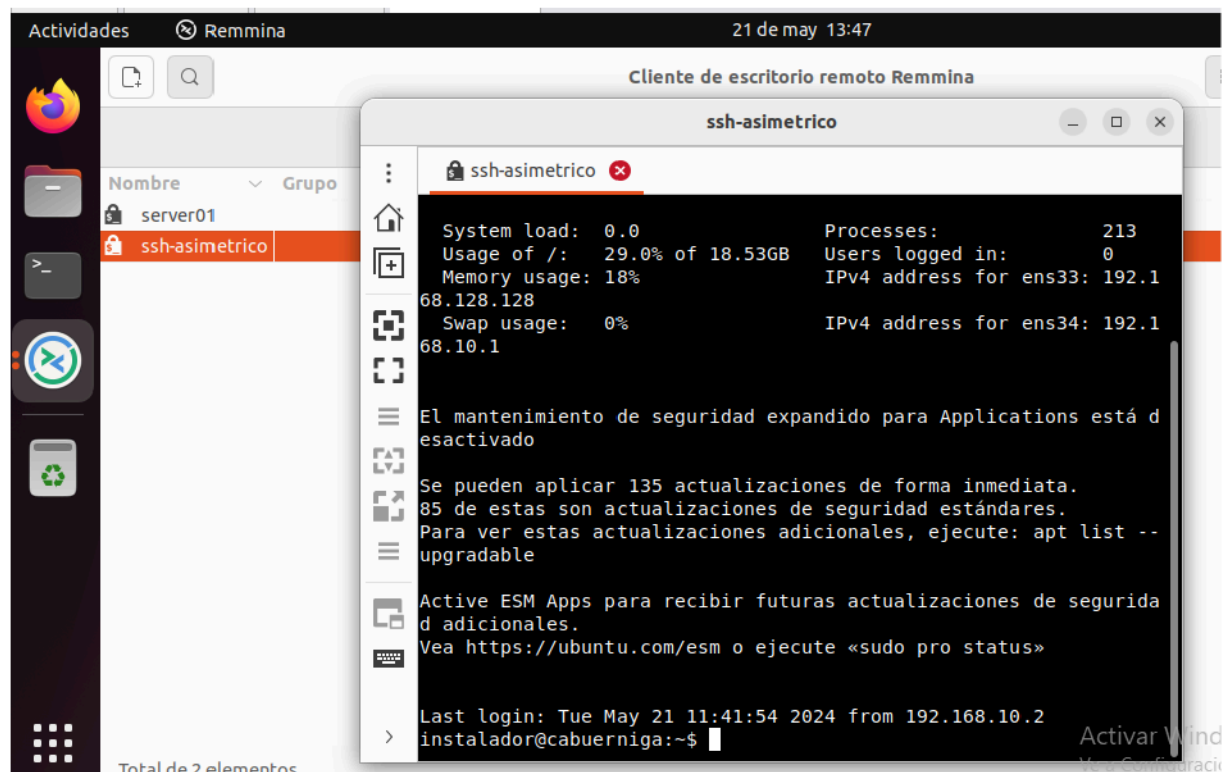
Que está asociada no solo al servidor sino también al usuario instalador.

3. Conéctate con ellas desde el software Remmina a server01.

La configuración quedaría así:



Y si nos conectamos:



ACTIVIDAD 4. SERVIDOR WEB

Vamos a implementar una intranet en nuestra LAN:

1. Instala el servidor web Apache2 en la maquina `server01`. Confirma que el servicio está corriendo.

Para la instalación, comando:

```
$ sudo apt-get install apache2
```

Confirmar que el servicio está corriendo:

```
$ service apache2 status
```

```
instalador@cabuerniga:~$ service apache2 status
• apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-05-22 09:25:02 UTC; 23s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2241 (apache2)
    Tasks: 55 (limit: 2178)
   Memory: 5.0M
      CPU: 50ms
   CGroup: /system.slice/apache2.service
           └─2241 /usr/sbin/apache2 -k start
             └─2242 /usr/sbin/apache2 -k start
               └─2244 /usr/sbin/apache2 -k start

may 22 09:25:02 cabuerniga systemd[1]: Starting The Apache HTTP Server...
may 22 09:25:02 cabuerniga apachectl[2240]: AH00558: apache2: Could not reliably determine the serv
may 22 09:25:02 cabuerniga systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

2. Sobrescribe la página `index.html` y añade código HTML que te identifique, a tu elección.

Antes de modificar la página original, una buen práctica es hacer una copia:

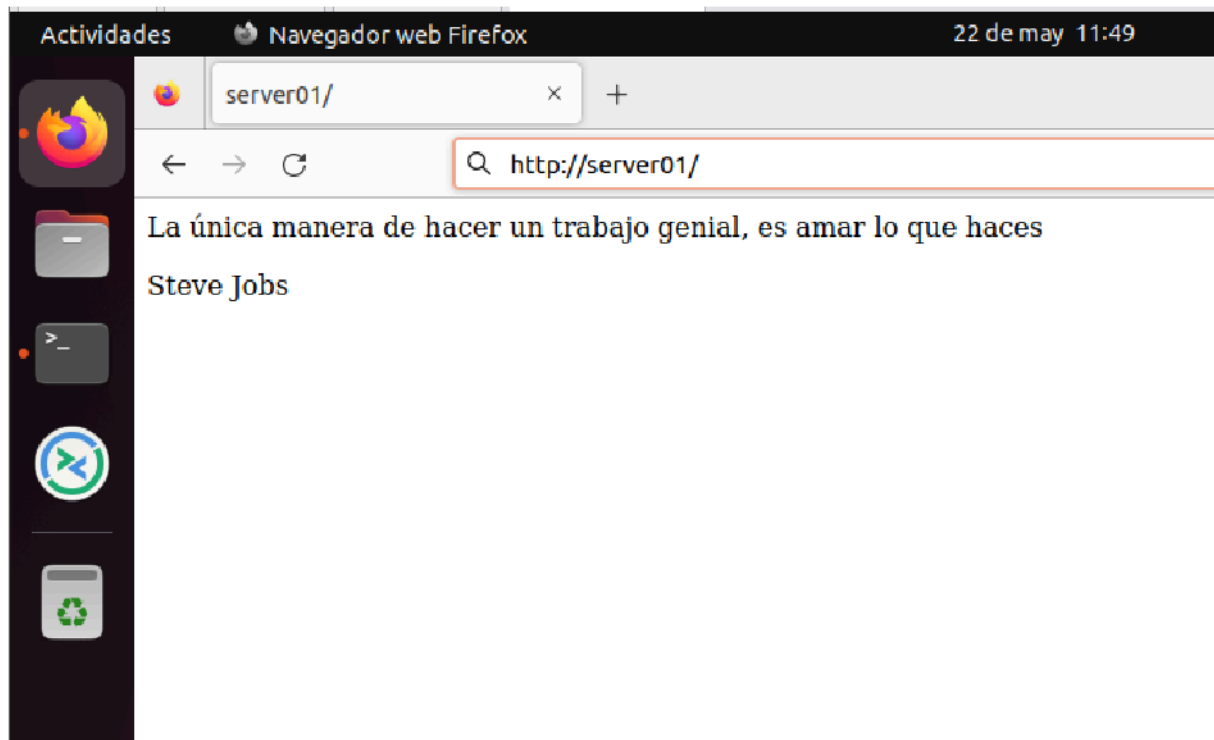
```
instalador@cabuerniga:~$ sudo mv /var/www/html/index.html /var/www/html/index.html.old
instalador@cabuerniga:~$ _
```

Ahora, ya editamos con nano:

```
GNU nano 6.2 /var/www/html/index.html
<html>
<body>
<p><k>La &uacute;nica manera de hacer un trabajo genial, es amar lo que haces</p></k>
<p>Steve Jobs</p>
</body>
</html>
```

Y...

3. Visualiza la página web anterior desde `cliente01`. ¿A través de que puerto escucha el servidor web?



¿Puedes demostrar que puertos tienes abiertos y escuchando **server01**?

Se deduce, que al menos el puerto 80 del servidor está abierto, ya nos ha brindado una página HTML a través de HTTP (puerto por defecto).

Instalamos en el cliente01 la herramienta [nmap](#)

```
fcuadradoa01@cerredo:~$ sudo apt install nmap
[sudo] contraseña para fcuadradoa01:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  libflashrom1 libftdi1-2
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  libblas3 liblinear4 liblua5.3-0 lua-lpeg nmap-common
Paquetes sugeridos:
  liblinear-tools liblinear-dev ncat ndiff zenmap
Se instalarán los siguientes paquetes NUEVOS:
  libblas3 liblinear4 liblua5.3-0 lua-lpeg nmap nmap-common
0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 406 no actualizados.
Se necesita descargar 6.113 kB de archivos.
Se utilizarán 26,8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
0% [Conectando a es.archive.ubuntu.com]
```

Y la ejecutamos contra **server01**:

```
fcuadradoa01@cerredo:~$ nmap -A -v -o resultados.txt 192.168.10.1
Warning: The -o option is deprecated. Please use -oN
Starting Nmap 7.80 ( https://nmap.org ) at 2024-05-22 12:07 CEST
```

Y efectivamente confirmamos que está escuchando por el puerto 22 y el puerto 80:

```
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http      Apache httpd 2.4.52 ((Ubuntu))
|_ http-methods:
|_   Supported Methods: GET POST OPTIONS HEAD
|_ http-server-header: Apache/2.4.52 (Ubuntu)
|_ http-title: Site doesn't have a title (text/html).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Además hemos dejado el resultado en el fichero resultados.txt por si es necesario un uso posterior.

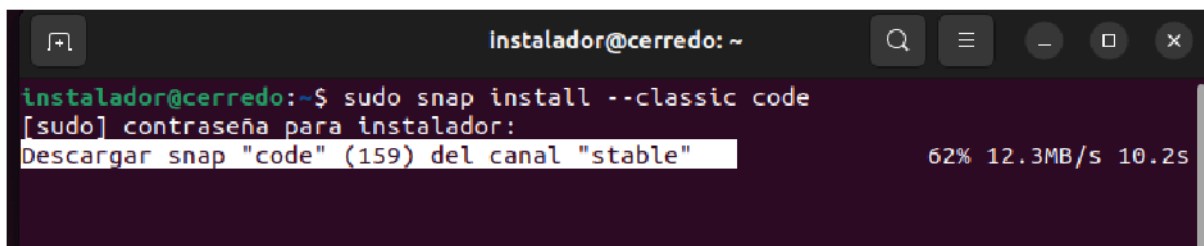
ACTIVIDAD 5. VS CODE + SSH + APACHE2

Ahora, queremos editar la página web `index.html` del punto 3, pero desde un IDE, para que nos resulta más cómodo y amigable. Para ello utilizaremos [MS Visual Studio Code](#). Este IDE nos proporciona un plugin para conectarnos a través de SSH a la máquina que tiene el código, en nuestro caso `server01`, y editarlo directamente desde nuestro equipo local (`cliente01`).

En la máquina `cliente01`:

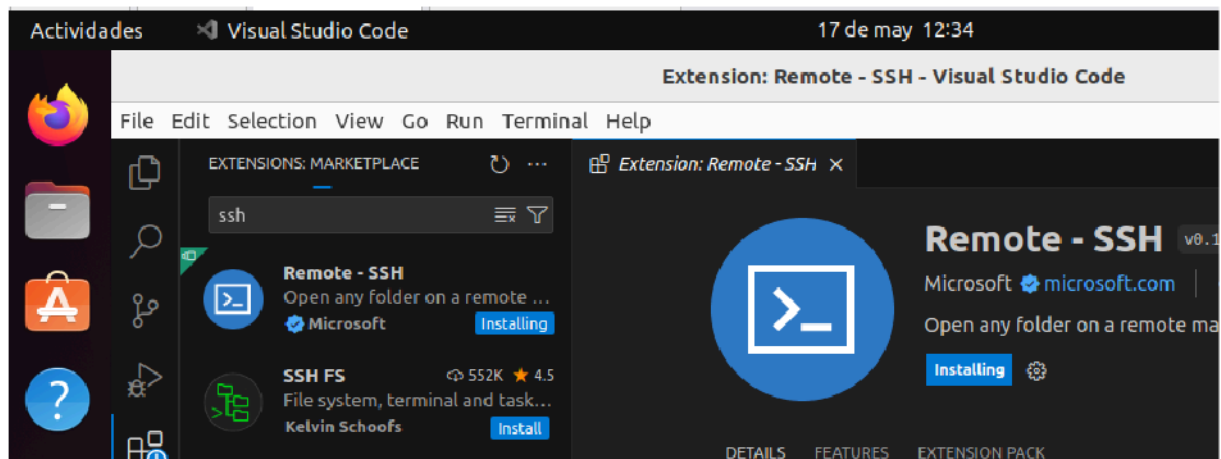
1. Instalamos Microsoft Visual Studio Code.

Para instalar el software en cuestión lo podemos hacer con el gestor de paquetes snap, quizás sea la opción mas sencilla. Además **snap** se va a convertir en el gestor de paquetes por defecto para Ubuntu, por lo que es bueno nos vayamos familiarizando con él.



```
instalador@cerredo: ~
instalador@cerredo:~$ sudo snap install --classic code
[sudo] contraseña para instalador:
Descargar snap "code" (159) del canal "stable" 62% 12.3MB/s 10.2s
```

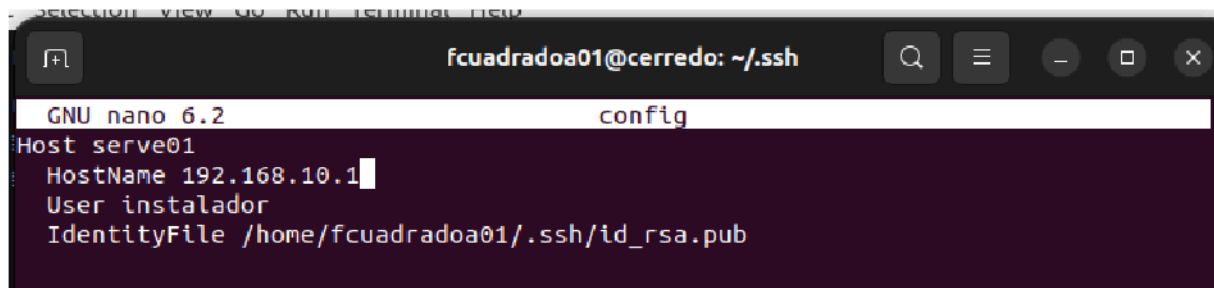
2. Instalamos un plugin, en VS Code, para conectarnos vía ssh.



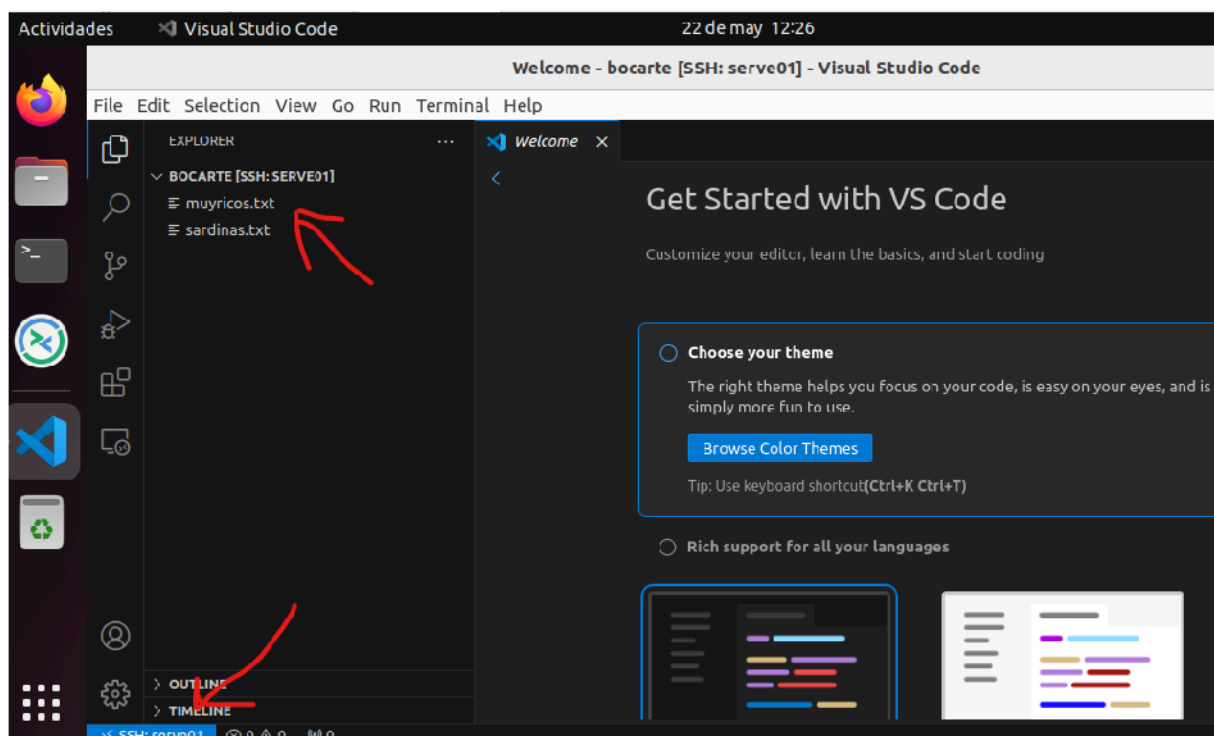
Mejor instalar la oficial de Microsoft para tal fin.

3. Configuramos y guardamos una conexión ssh a **server01**.

Aprovechamos la clave asimétrica que ya tenemos configurada de pasos anteriores y agregamos un nuevo registro en nuestro fichero `./ssh/config`



Ahora escogemos la conexión serve01... (lo siento me he comido la "r" final)

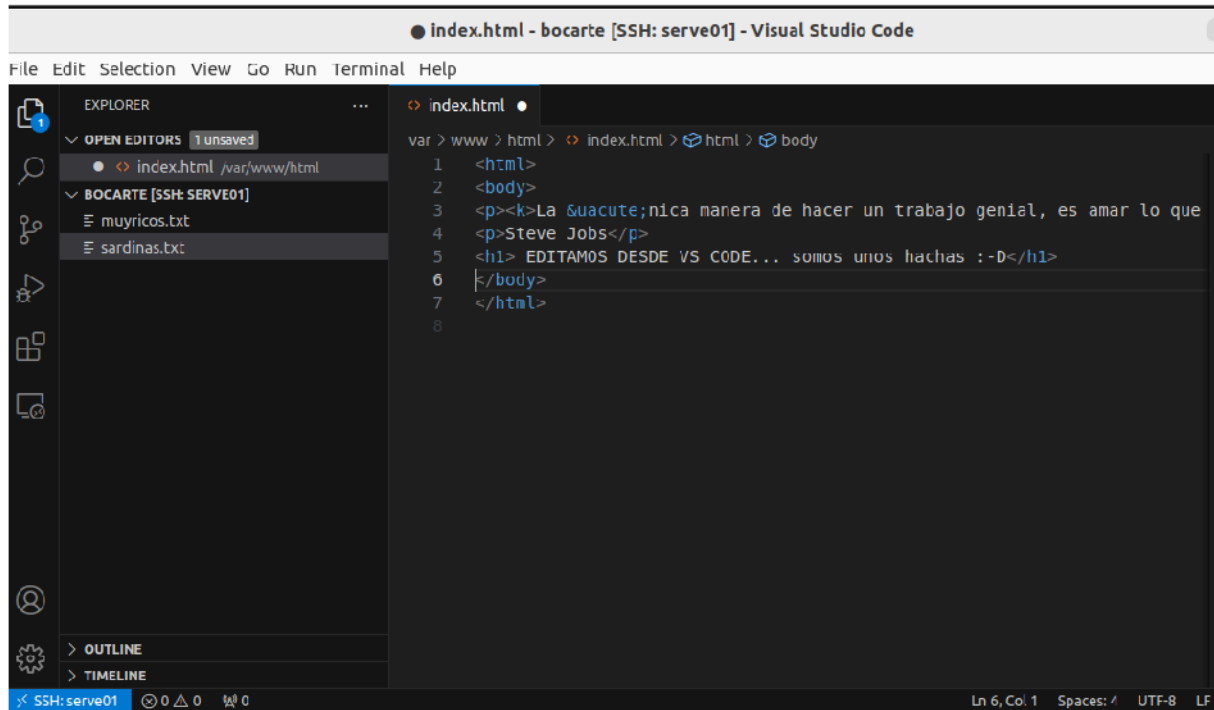


4. Editamos la página index.html, desde la conexión anterior.

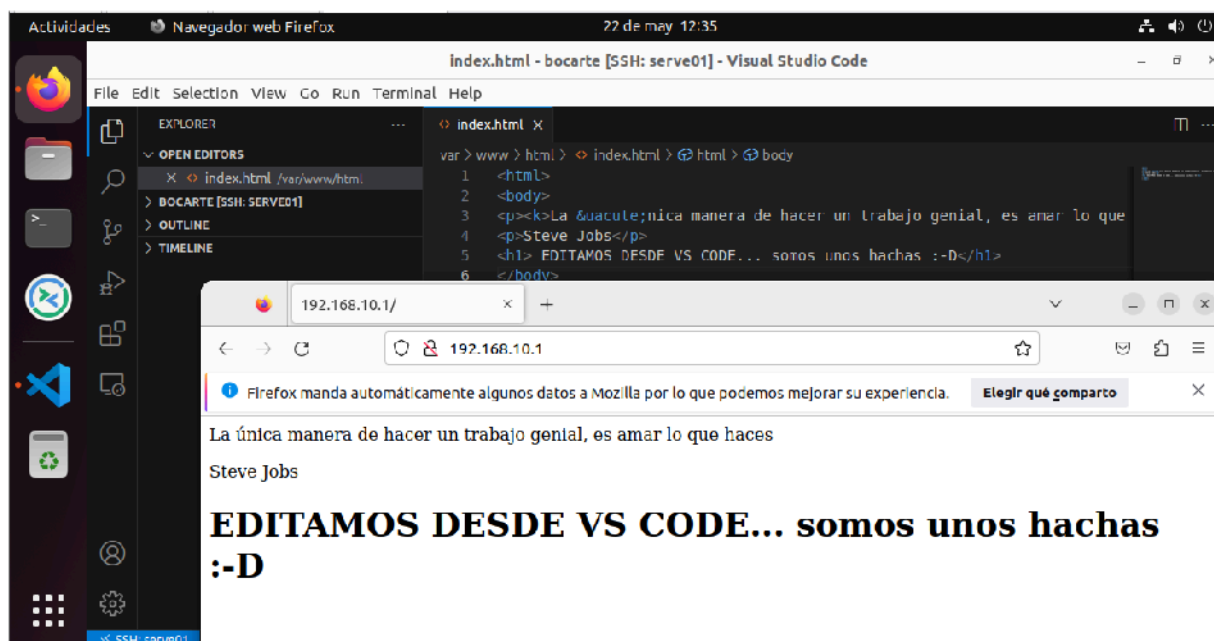
Hay que tocar permisos del archivo index.html

```
instalador@cabuerniga:/var/www/html$ sudo chmod o+w index.html
instalador@cabuerniga:/var/www/html$ _
```

Y luego lo abrimos desde menú File...



5. Visualizamos los cambios desde nuestro navegador web.



RECURSOS

[consultado marzo 2024] Netplan.io

<https://netplan.io/>

[consultado marzo-2024] Netplan documentation

[How to enable DHCP on an interface - Netplan documentation](#)

[consultado feb-2023] SSH clave pública y privada:

[SSH clave publica y privada - El Taller del Bit](#)

[consultado feb-2023] Copiar clave pública SSH a un server Linux | SCP y ssh-copy-id

<https://eltallerdelbit.com/copiar-clave-publica-ssh-scp-ssh-copy-id/>

Apuntes de la plataforma.

CRITERIOS DE CORRECCIÓN

Actividad 1. 2 puntos.

Actividad 2. 2 puntos

Actividad 3: 2 puntos

Actividad 4: 2 puntos.

Actividad 5: 2 puntos

**RECORDAD QUE LAS ENTREGAS QUE NO CUMPLAN CON EL FORMATO
EXIGIDO DE “ENTEGA DE TAREAS” SERAN PENALIZADAS.**

CAPTURA LAS PANTALLAS que JUSTIFIQUEN lo solicitado

COMENTA las CAPTURAS de PANTALLA

IDENTIFICATE EN TODAS LAS CAPTURAS

Fecha fin entrega: **3 de mayo de 2024**