
Servidores Web de Altas Prestaciones.

Práctica 2

Clonar la información de un Sitio Web.

Ricardo Ruiz Fernández de Alba

22/05/2023



Índice

Introducción	2
Tarea 1. Copiar archivos mediante SSH	2
Usando tubería PIPE	2
Usando SCP	3
Tarea 2. Clonar contenido entre máquinas	4
Rsync Avanzado	6
Tarea 3. Configurar SSH para acceder sin contraseña	10
Tarea 4. Establecer una tarea con cron	11

Introducción

En esta práctica, se abordarán varios aspectos relacionados con el uso de SSH y la automatización de tareas. Los objetivos principales son aprender a copiar archivos mediante SSH, clonar contenido entre máquinas, configurar SSH para acceder a máquinas remotas sin contraseña y establecer tareas periódicas utilizando cron.

Realizaremos las siguientes tareas básicas:

1. Probar la funcionalidad de copia de archivos a través de SSH.
2. Realizar el clonado de una carpeta entre las dos máquinas.
3. Configurar SSH para permitir el acceso sin solicitar contraseña.
4. Establecer una tarea programada en cron que se ejecute cada hora para mantener actualizado el contenido del directorio /var/www entre las dos máquinas.

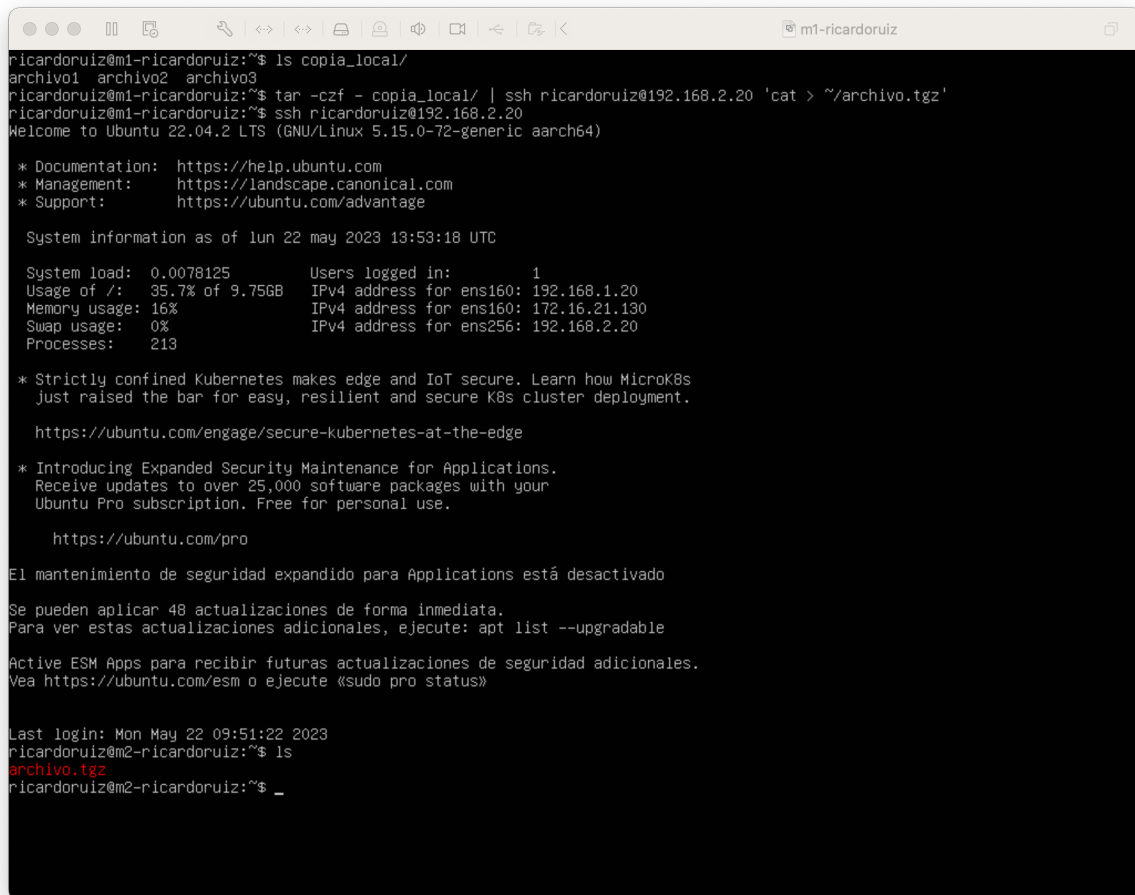
Tarea 1. Copiar archivos mediante SSH

Usando tubería PIPE

Supongamos que no tenemos suficiente espacio en disco local para crear un archivo tar.gz. Podemos crearlo directamente en el equipo remoto mediante SSH.:

```
1 ricardoruiz@m1-ricardoruiz $ tar -czf - copia_local/ | ssh  
ricardoruiz@192.167.2.20 'cat > ~/archivo.tgz'
```

De esta manera, se creará el archivo tar.tgz en el equipo remoto.



```
ricardoruiz@m1-ricardoruiz:~$ ls copia_local/
archivo1 archivo2 archivo3
ricardoruiz@m1-ricardoruiz:~$ tar -czf - copia_local/ | ssh ricardoruiz@192.168.2.20 'cat > ~/archivo.tgz'
ricardoruiz@m1-ricardoruiz:~$ ssh ricardoruiz@192.168.2.20
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-72-generic aarch64)

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

System information as of lun 22 may 2023 13:53:18 UTC

System load:  0.0078125      Users logged in:      1
Usage of /:   35.7% of 9.75GB IPv4 address for ens160: 192.168.1.20
Memory usage: 16%           IPv4 address for ens160: 172.16.21.130
Swap usage:   0%            IPv4 address for ens256: 192.168.2.20
Processes:    213

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

   https://ubuntu.com/pro

El mantenimiento de seguridad expandido para Applications está desactivado
Se pueden aplicar 48 actualizaciones de forma inmediata.
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: Mon May 22 09:51:22 2023
ricardoruiz@m2-ricardoruiz:~$ ls
archivo.tgz
ricardoruiz@m2-ricardoruiz:~$ _
```

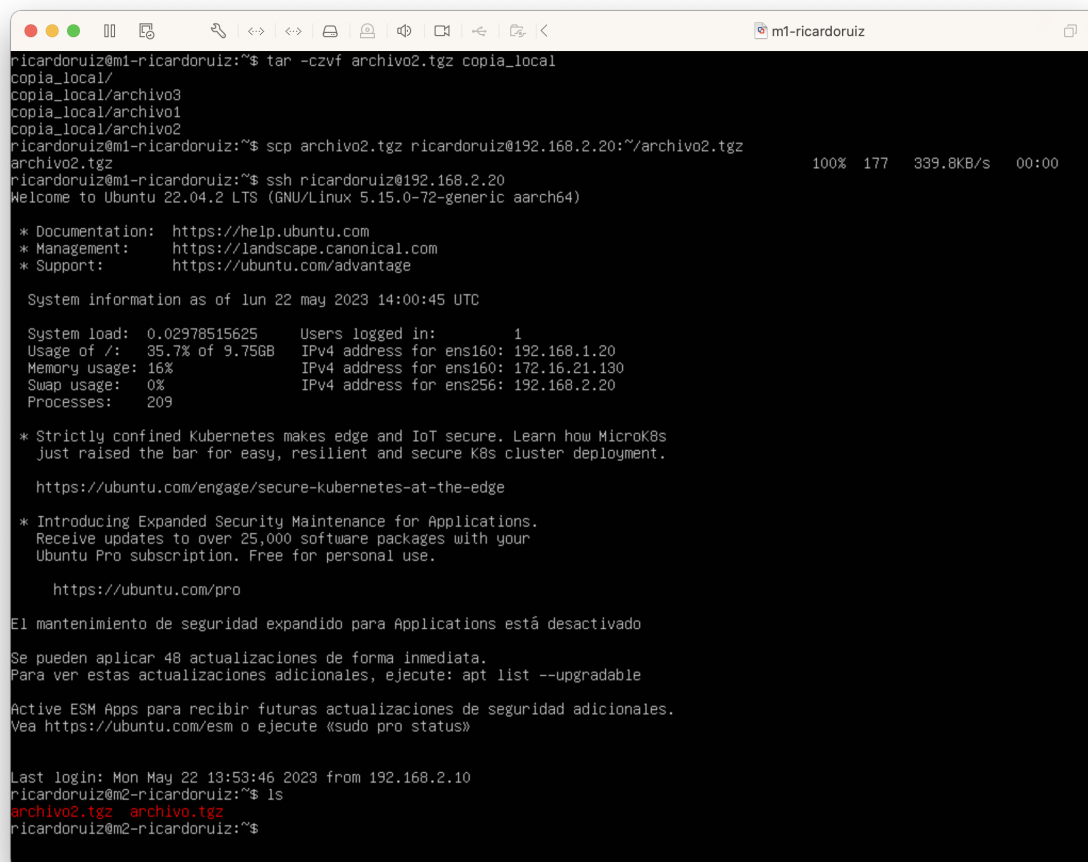
Figura 1: Copia mediante SSH

También es posible utilizar SCP, que utiliza SSH para realizar copias seguras y encriptadas de archivos o directorios. Podemos crear un archivo tar.gz localmente y luego copiarlo al equipo remoto utilizando SCP. Los comandos serían los siguientes:

Usando SCP

También podemos realizar esta tarea usando SCP que utiliza SSH para hacer copias seguras y encriptadas de archivos o directorios.

```
1 ricardoruiz@m1-ricardoruiz $ tar -czvf archivo2.tgz copia_local
2 ricardoruiz@m1-ricardoruiz $ scp archivo.tgz ricardoruiz@192
  .168.2.20:~/archivo2.tgz
```



```
ricardoruiz@m1-ricardoruiz:~$ tar -czvf archivo2.tgz copia_local
copia_local/
copia_local/archivo3
copia_local/archivo1
copia_local/archivo2
ricardoruiz@m1-ricardoruiz:~$ scp archivo2.tgz ricardoruiz@192.168.2.20:~/archivo2.tgz
100% 177 339.8KB/s 00:00
ricardoruiz@m1-ricardoruiz:~$ ssh ricardoruiz@192.168.2.20
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-72-generic aarch64)

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

System information as of lun 22 may 2023 14:00:45 UTC

System load: 0.02978515625   Users logged in:      1
Usage of /:   35.7% of 9.75GB   IPv4 address for ens160: 192.168.1.20
Memory usage: 16%           IPv4 address for ens160: 172.16.21.130
Swap usage:   0%             IPv4 address for ens256: 192.168.2.20
Processes:   209

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

   https://ubuntu.com/pro

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 48 actualizaciones de forma inmediata.
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: Mon May 22 13:53:46 2023 from 192.168.2.10
ricardoruiz@m2-ricardoruiz:~$ ls
archivo2.tgz  archivo.tgz
ricardoruiz@m2-ricardoruiz:~$
```

Figura 2: Copia mediante SCP

O enviar el propio directorio sin comprimir:

```
1 $ scp -r copia_local ricardoruiz@m2-ricardoruiz:~/copia_local
```

Copia del directorio mediante SCP

Figura 3: Copia del directorio mediante SCP

Tarea 2. Clonar contenido entre máquinas

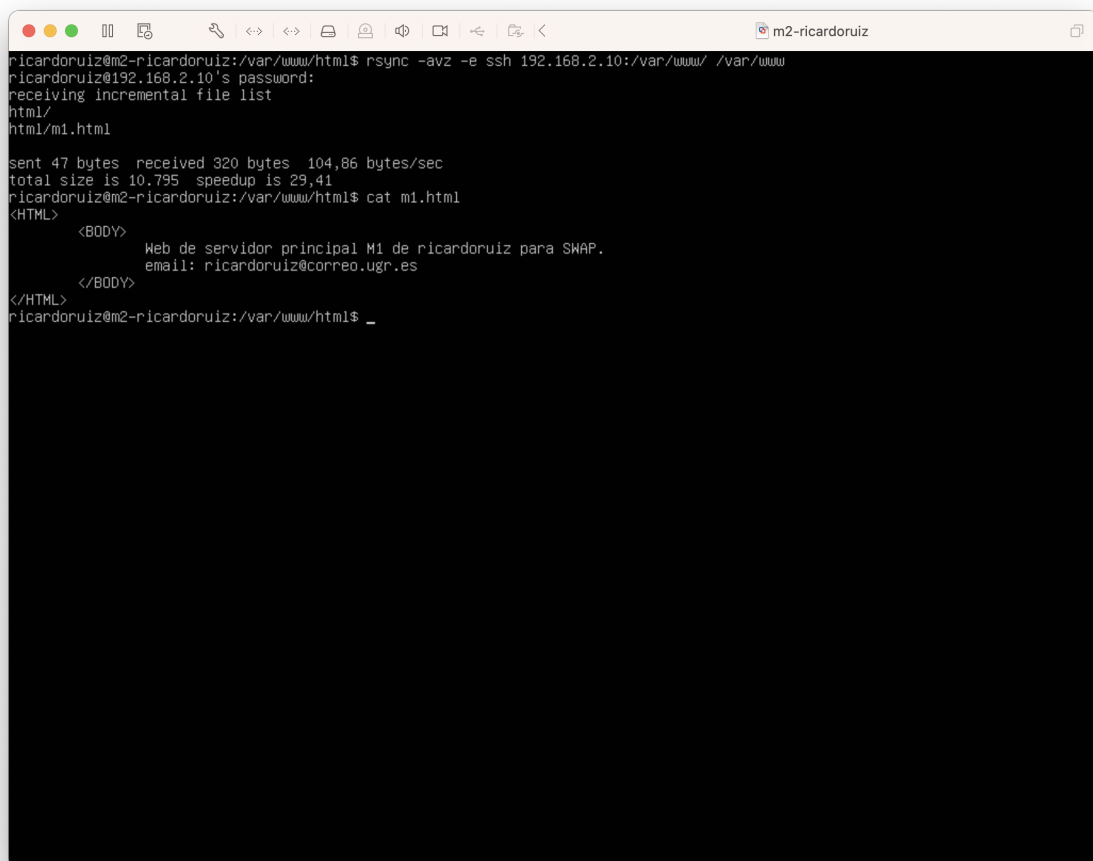
Para directorios de mayor tamaño, es mejor utiliza rsync. La herramienta rsync es una opción útil para realizar copias y sincronización de archivos. Lo instalamos:

```
1 ricardoruiz@m2-ricardoruiz $ sudo apt-get install rsync
```

Queremos clonar la carpeta que contiene el servidor web principal desde la máquina secundario. Creamos un archivo `m1.html` en `/var/www/html` para distinguir el servidor principal. Es necesario que el usuario sea propietario de la carpeta que queremos sincronizar.

```
1 ricardoruiz@m2-ricardoruiz $ sudo chown ricardoruiz:ricardoruiz -R /var/www
2 ricardoruiz@m2-ricardoruiz $ rsync -avz -e ssh 192.168.2.10:/var/www/ /var/www
```

Rsync nos pedirá la clave de usuario en M1 y comprobamos que se actualiza el contenido del directorio `/var/www/html` con el contenido del servidor principal (`m1.html`)



```
ricardoruiz@m2-ricardoruiz:/var/www/html$ rsync -avz -e ssh 192.168.2.10:/var/www/ /var/www
ricardoruiz@192.168.2.10's password:
receiving incremental file list
html/
html/m1.html

sent 47 bytes  received 320 bytes  104,86 bytes/sec
total size is 10.795  speedup is 29,41
ricardoruiz@m2-ricardoruiz:/var/www/html$ cat m1.html
<HTML>
  <BODY>
    Web de servidor principal M1 de ricardoruiz para SWAP.
    email: ricardoruiz@correo.ugr.es
  </BODY>
</HTML>
ricardoruiz@m2-ricardoruiz:/var/www/html$ _
```

Figura 4: Rsync

Rsync Avanzado

Podemos especificar qué directorios ignorar durante el proceso de copia. Por ejemplo, si queremos realiza rcopia de `/var/www` pero excluir los directorio `/var/www/error`, `/var/www/stats` y `/var/www/files/pictures` usamos `--exclude`.

En primer lugar, creamos la estructura de directorios en el servidor secundario, con los archivos `error.html`, `stats.html` y añadimos `img.jpg` `/var/www/files/pictures`.

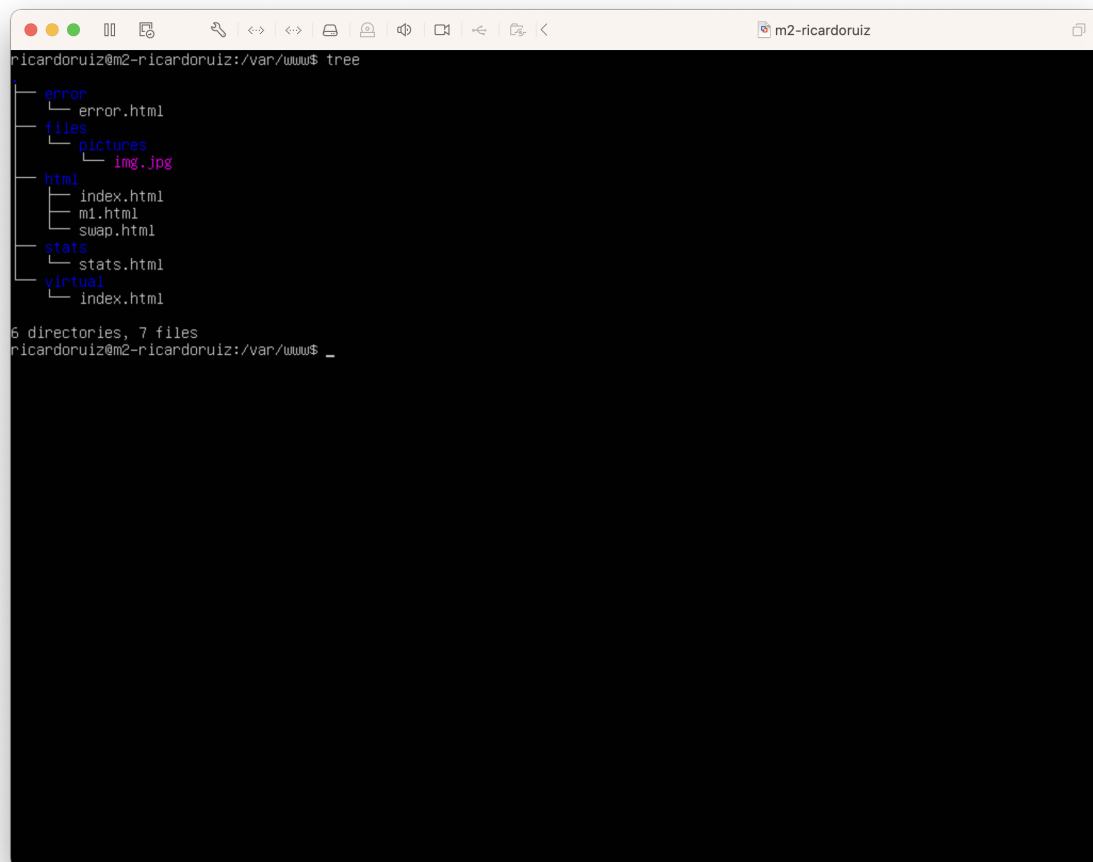
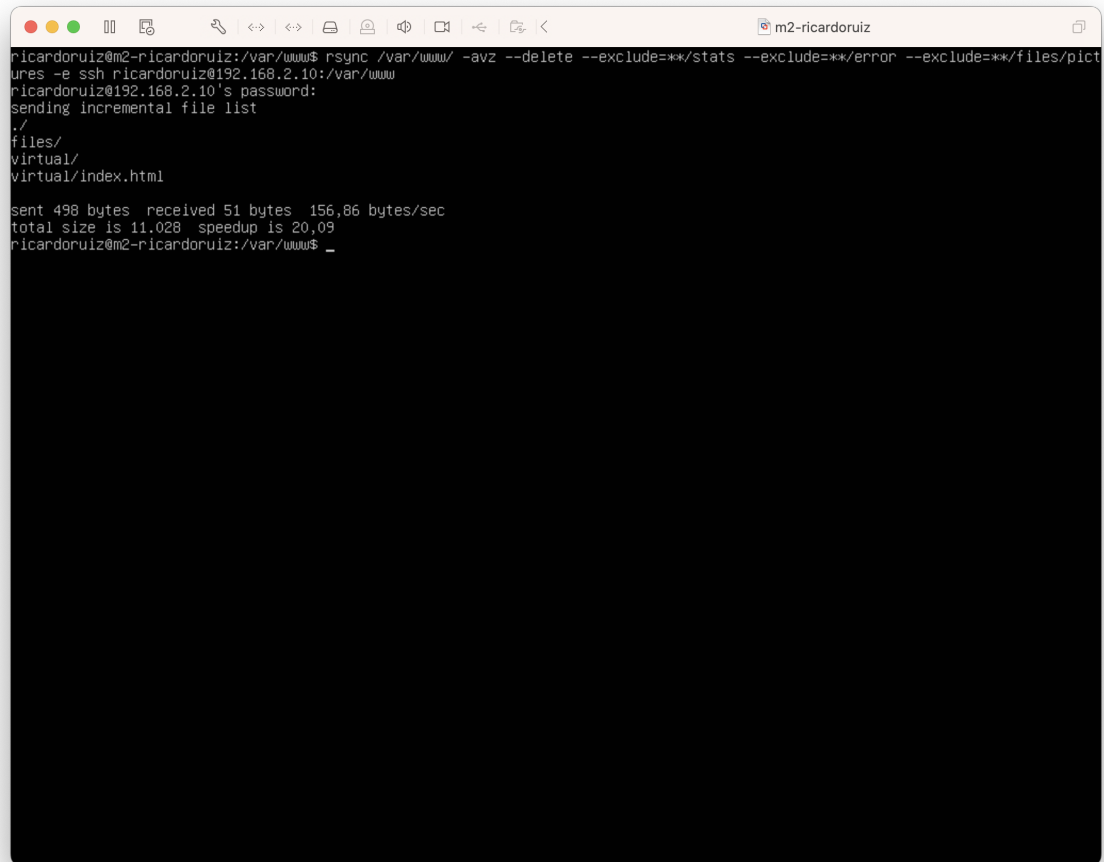
A terminal window titled 'm2-ricardoruiz' showing the output of the 'tree' command in the directory '/var/www'. The output displays a hierarchical structure: 'error' (containing 'error.html'), 'files' (containing 'pictures' which has 'img.jpg'), 'html' (containing 'index.html', 'm1.html', and 'swap.html'), 'stats' (containing 'stats.html'), and 'virtual' (containing 'index.html'). Below the tree, it states '6 directories, 7 files'. The prompt is 'ricardoruiz@m2-ricardoruiz:/var/www\$'.

Figura 5: Estructura de directorios

```
1 ricardoruiz@m2-ricardoruiz $ rsync /var/www/ -avz --delete --exclude  
=**/stats --exclude=**/error --exclude=**/files/pictures -e ssh  
192.168.2.10:/var/www
```

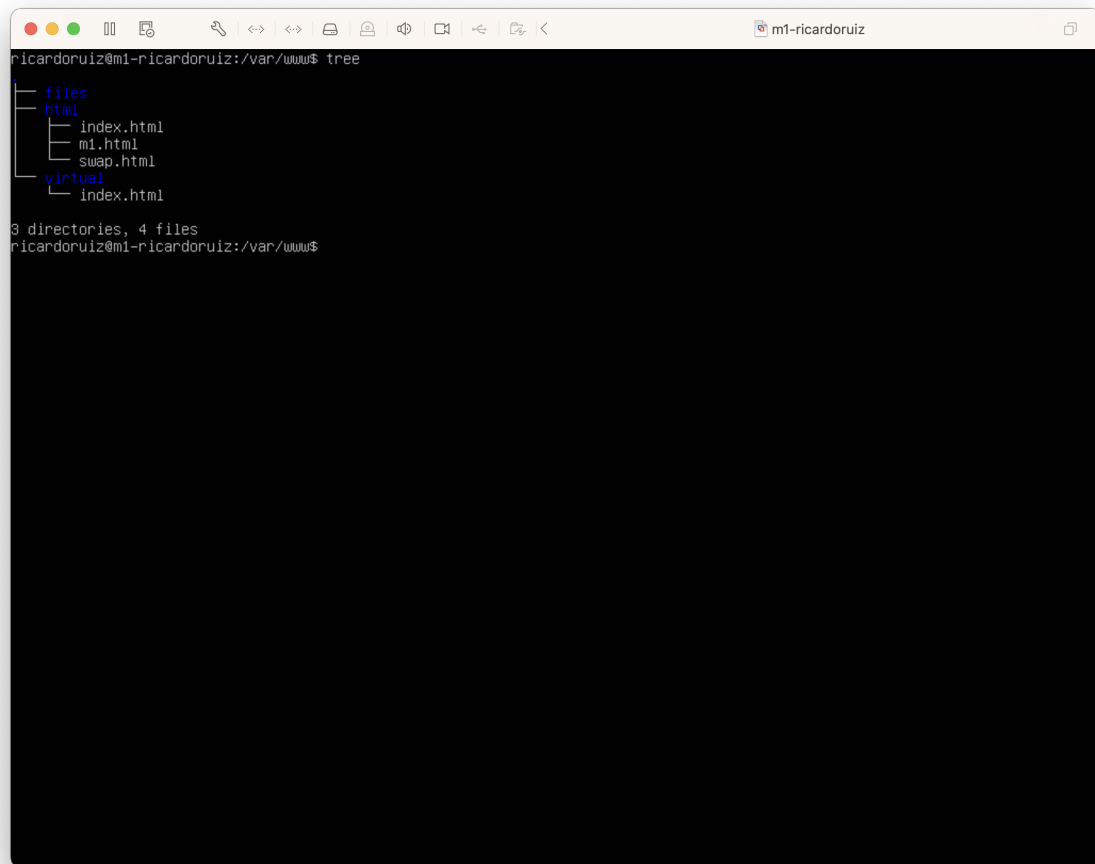
Así evitamos que los errores y estadísticas de la máquina 2 se sobrescriban con los de la principal.

A terminal window titled 'm2-ricardoruiz' showing a command execution. The command is 'rsync /var/www/ -avz --delete --exclude=**/stats --exclude=**/error --exclude=**/files/pictures -e ssh ricardoruiz@192.168.2.10:/var/www'. The output shows the connection to the remote host, the password prompt, the sending of an incremental file list, and the transfer of files. The files listed are 'virtual/' and 'virtual/index.html'. The transfer statistics show 498 bytes sent, 51 bytes received, a speed of 156.86 bytes/sec, a total size of 11.028, and a speedup of 20.09.

```
ricardoruiz@m2-ricardoruiz:/var/www$ rsync /var/www/ -avz --delete --exclude=**/stats --exclude=**/error --exclude=**/files/pictures -e ssh ricardoruiz@192.168.2.10:/var/www
ricardoruiz@192.168.2.10's password:
sending incremental file list
./
files/
virtual/
virtual/index.html

sent 498 bytes  received 51 bytes  156.86 bytes/sec
total size is 11.028  speedup is 20.09
ricardoruiz@m2-ricardoruiz:/var/www$ _
```

Figura 6: Opción exclude



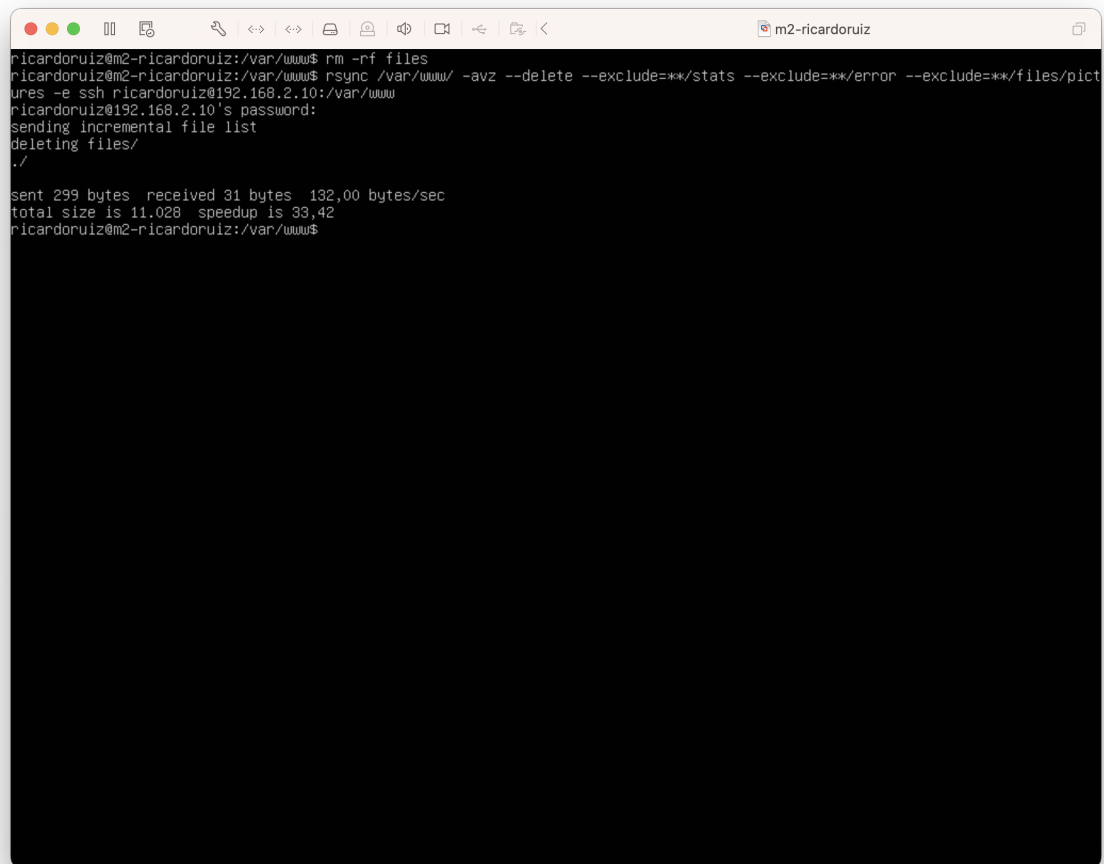
```
ricardoruiz@m1-ricardoruiz:/var/www$ tree
.
├── files
├── html
│   ├── index.html
│   ├── m1.html
│   └── swap.html
└── virtual
    └── index.html

3 directories, 4 files
ricardoruiz@m1-ricardoruiz:/var/www$
```

Figura 7: Opción exclude

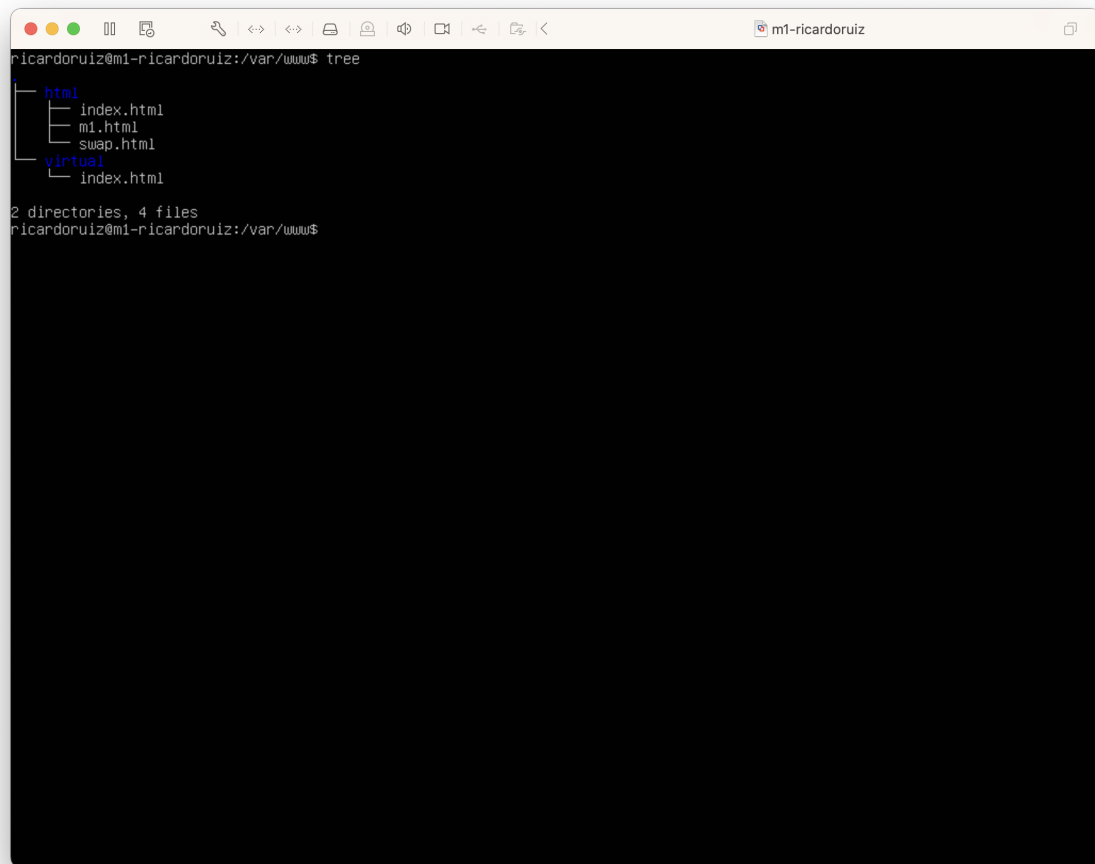
La opción `--delete` indica que los archivos que se hayan eliminado en la máquina de origen también se eliminarán en la máquina de destino para asegurar una clonación idéntica.

Comprobémoslo eliminando el directorio `files` y volviendo a ejecutar `rsync`:

A terminal window titled 'm2-ricardoruiz' showing a series of commands and their outputs. The user is in the directory /var/www. They run 'rm -rf files' to delete a directory. Then they run 'rsync /var/www/ -avz --delete --exclude=**/stats --exclude=**/error --exclude=**/files/pictures -e ssh ricardoruiz@192.168.2.10:/var/www' to synchronize the directory with a remote server. The output shows the password prompt, file list sending, and deletion of files. Finally, it shows the transfer statistics: sent 299 bytes, received 31 bytes, 132.00 bytes/sec, total size is 11.028, speedup is 33.42.

```
ricardoruiz@m2-ricardoruiz:/var/www$ rm -rf files
ricardoruiz@m2-ricardoruiz:/var/www$ rsync /var/www/ -avz --delete --exclude=**/stats --exclude=**/error --exclude=**/files/pictures -e ssh ricardoruiz@192.168.2.10:/var/www
ricardoruiz@192.168.2.10's password:
sending incremental file list
deleting files/
./
sent 299 bytes  received 31 bytes  132,00 bytes/sec
total size is 11.028  speedup is 33,42
ricardoruiz@m2-ricardoruiz:/var/www$
```

Figura 8: Opción delete



```
ricardoruiz@m1-ricardoruiz:/var/www$ tree
.
├── html
│   ├── index.html
│   ├── m1.html
│   └── swap.html
└── virtual
    └── index.html

2 directories, 4 files
ricardoruiz@m1-ricardoruiz:/var/www$
```

Figura 9: Opción delete

Tarea 3. Configurar SSH para acceder sin contraseña

Para lograr una actualización automática sin intervención del administrador, es necesario utilizar autenticación con claves pública-privada.

De forma parecida como se hicimos en la configuración avanzada de la primer práctica, ejecutamos esta vez en M2:

```
1 ricardoruiz@m2-ricardoruiz $ ssh-keygen -b 4096 -t rsa
```

Esto generará, por defecto, el fichero `~/.ssh/id_rsa` para la clave privada y el fichero `~/.ssh/id_rsa.pub` para la clave pública. Este formato es válido para el protocolo 2 de SSH. Debemos copiar la clave pública al equipo remoto (máquina principal) en `~/.ssh/authorized_key` con permisos

600.

Podemos realizarlo de forma sencilla utilizando el comando `ssh-copy-id`.

```
1 ricardoruiz@m2-ricardoruiz $ ssh-copy-id 192.168.1.20
```

Finalmente, podemos destacar la manera de ejecutar comandos en el equipo remoto, esta vez sin solicitud de contraseña.

```
1 ricardoruiz@m2-ricardoruiz $ ssh 192.168.1.20 uname -a
```

Acceder sin contraseña

Figura 10: Acceder sin contraseña

Tarea 4. Establecer una tarea con cron