



UNIVERSIDAD
DE GRANADA

PRÁCTICA 2: CLONAR LA INFORMACIÓN DE UN SITIO WEB

ANA BUENDÍA RUIZ-AZUAGA

Práctica 2: Clonar la información de un sitio web

Correo electrónico

anabuenrúa@correo.ugr.es

E.T.S. INGENIERÍAS INFORMÁTICA Y DE TELECOMUNICACIÓN

Granada, a 28 de marzo de 2022

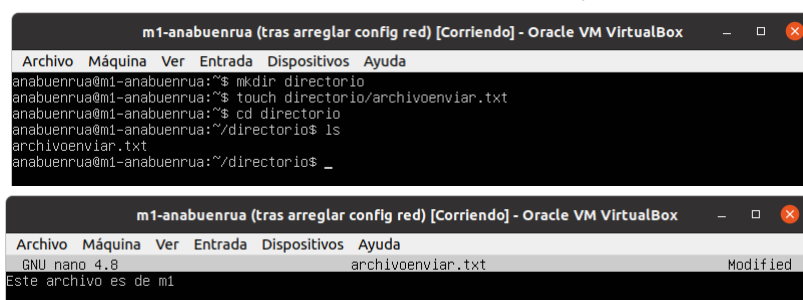
ÍNDICE GENERAL

1.	COPIA DE ARCHIVOS	3
1.1.	Envío mediante tar y scp	4
1.2.	Comandos avanzados	4
2.	UTILIZANDO RSYNC	6
2.1.	Opciones avanzadas	6
3.	ACCESO MEDIANTE SSH SIN CONTRASEÑA	10
3.1.	Opciones avanzadas	10
3.2.	Copia de clave manual	12
4.	USANDO CRONTAB	14
4.1.	Opciones avanzadas	15
5.	BIBLIOGRAFÍA	16

COPIA DE ARCHIVOS

Vamos a comenzar enviando el directorio con tar, de forma simple. Comenzamos creando un directorio con un archivo como se ve en (1).

Figura 1: Creación de directorio con un archivo en m1 y contenido de este archivo.

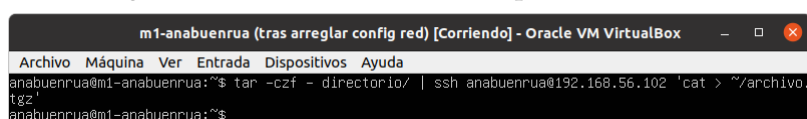


```
m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenrúa@m1-anabuenrúa:~$ mkdir directorio
anabuenrúa@m1-anabuenrúa:~$ touch directorio/archivoenviar.txt
anabuenrúa@m1-anabuenrúa:~$ cd directorio
anabuenrúa@m1-anabuenrúa:~/directorio$ ls
archivoenviar.txt
anabuenrúa@m1-anabuenrúa:~/directorio$ _

m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 4.8 archivoenviar.txt Modified
Este archivo es de m1
```

Y mandamos el directorio comprimido con tar (2). Como ya configuramos el acceso por ssh sin contraseña no nos la pide.

Figura 2: Envío del archivo comprimido con tar



```
m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenrúa@m1-anabuenrúa:~$ tar -czf - directorio/ | ssh anabuenrúa@192.168.56.102 'cat > ~/archivo.tgz'
anabuenrúa@m1-anabuenrúa:~$
```

Finalmente descomprimos y comprobamos que se ha mandado correctamente (3).

Figura 3: Descompresión y comprobación del envío correcto del archivo.



```
m2-anabuenrúa (tras arreglar red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenrúa@m2-anabuenrúa:~$ ls
archivo.tgz cookies.txt fichero.html swap.html
anabuenrúa@m2-anabuenrúa:~$
```

1.1 ENVÍO MEDIANTE TAR Y SCP

Ahora vamos a enviarlo mediante tar y scp. Para ello creamos el tar y lo mandamos mediante scp como se ve en (4)

Figura 4: Compresión con tar y envío mediante scp

```
m2-anabuenruea (tras arreglar red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenruea@m2-anabuenruea:~$ tar -czf archivo.tar.gz cookies.txt fichero.html swap.html
anabuenruea@m2-anabuenruea:~$ scp archivo.tar.gz anabuenruea@m2-anabuenruea:~$
```

Comprobamos en (5) que en la máquina 2 se encuentra directorio2.

Figura 5: Comprobación de la llegada de directorio2 a m2

```
m2-anabuenruea (tras arreglar red) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenruea@m2-anabuenruea:~$ ls
anabuenruea@m2-anabuenruea:~$ scp directorio2.tar.gz anabuenruea@m2-anabuenruea:~$
```

1.2 COMANDOS AVANZADOS

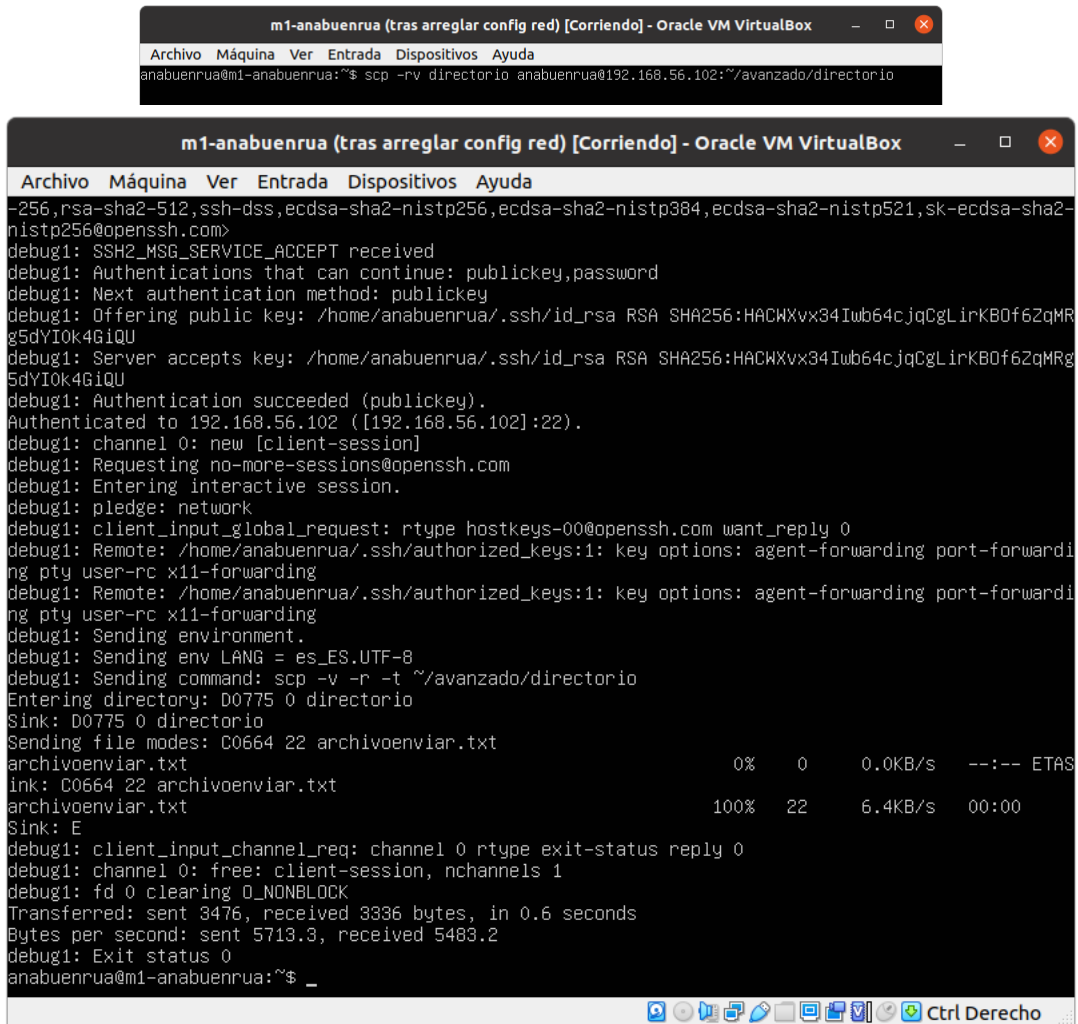
Vamos a enviar el directorio esta vez usando scp y algunas de sus opciones.

Comenzamos son -r, que copia recursivamente directorios enteros, y -v nos da información de la copia y de ssh. Como vemos en (6) muestra mucha información.

También podemos usar la opción -q que desactiva que se muestren mensajes por si se mandan muchos archivos (7)

Finalmente, con -P podemos indicar el puerto. Por ejemplo de m2 a m1 como se muestra en (8)

Figura 6: Uso de scp con comandos avanzados



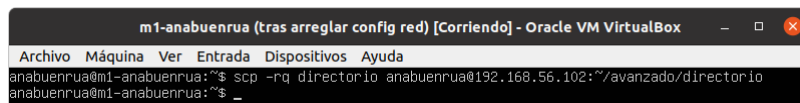
```

m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ scp -rv directorio anabuenrúa@192.168.56.102:~/avanzado/directorio

m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
-256,rsa-sha2-512,ssh-dss,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,sk-ecdsa-sha2-nistp256@openssh.com>
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug1: Authentications that can continue: publickey,password
debug1: Next authentication method: publickey
debug1: Offering public key: /home/anabuenrúa/.ssh/id_rsa RSA SHA256:HACWxvx34Iwb64cjQcGLirKB0f62qMRg5dYI0k4GiQU
debug1: Server accepts key: /home/anabuenrúa/.ssh/id_rsa RSA SHA256:HACWxvx34Iwb64cjQcGLirKB0f62qMRg5dYI0k4GiQU
debug1: Authentication succeeded (publickey).
Authenticated to 192.168.56.102 ([192.168.56.102]:22).
debug1: channel 0: new [client-session]
debug1: Requesting no-more-sessions@openssh.com
debug1: Entering interactive session.
debug1: pledge: network
debug1: client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
debug1: Remote: /home/anabuenrúa/.ssh/authorized_keys:1: key options: agent-forwarding port-forwarding pty user-rc x11-forwarding
debug1: Remote: /home/anabuenrúa/.ssh/authorized_keys:1: key options: agent-forwarding port-forwarding pty user-rc x11-forwarding
debug1: Sending environment.
debug1: Sending env LANG = es_ES.UTF-8
debug1: Sending command: scp -v -r -t ~/avanzado/directorio
Entering directory: D0775 0 directorio
Sink: D0775 0 directorio
Sending file modes: C0664 22 archivoenviar.txt
archivoenviar.txt          0%    0    0.0KB/s  --:-- ETAS
link: C0664 22 archivoenviar.txt
archivoenviar.txt          100%  22    6.4KB/s  00:00
Sink: E
debug1: client_input_channel_req: channel 0 rtype exit-status reply 0
debug1: channel 0: free: client-session, nchannels 1
debug1: fd 0 clearing O_NONBLOCK
Transferred: sent 3476, received 3336 bytes, in 0.6 seconds
Bytes per second: sent 5713.3, received 5483.2
debug1: Exit status 0
anabuenrúa@m1-anabuenrúa:~$ _

```

Figura 7: Uso de scp con comandos avanzados

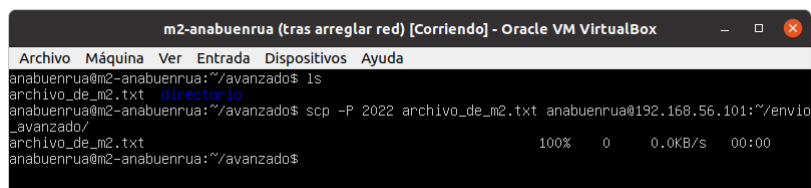


```

m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ scp -rq directorio anabuenrúa@192.168.56.102:~/avanzado/directorio
anabuenrúa@m1-anabuenrúa:~$ _

```

Figura 8: Uso de scp con comandos avanzados



```

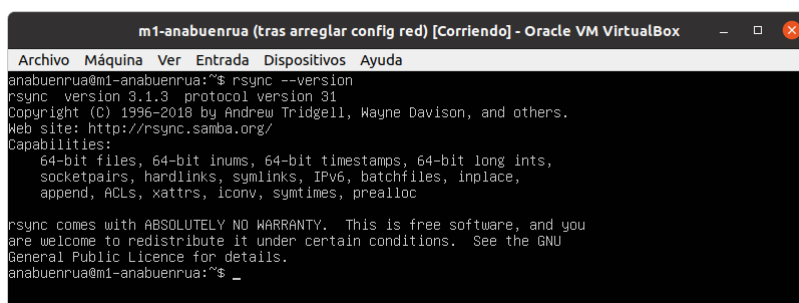
m2-anabuenrúa (tras arreglar red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m2-anabuenrúa:~/avanzado$ ls
archivo_de_m2.txt directorio
anabuenrúa@m2-anabuenrúa:~/avanzado$ scp -P 2022 archivo_de_m2.txt anabuenrúa@192.168.56.101:~/envio/avanzado/
archivo_de_m2.txt          100%    0    0.0KB/s  00:00
anabuenrúa@m2-anabuenrúa:~/avanzado$

```

UTILIZANDO RSYNC

Rsync ya está instalado en ambas máquinas, comprobamos su versión en (9)

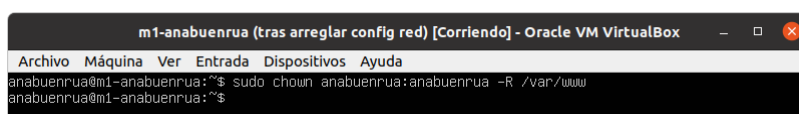
Figura 9: Comprobación de la versión de Rsync.



```
m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ rsync --version
rsync version 3.1.3, protocol version 31
Copyright (C) 1996-2018 by Andrew Tridgell, Wayne Davison, and others.
Web site: http://rsync.samba.org/
Capabilities:
  64-bit files, 64-bit inums, 64-bit timestamps, 64-bit long ints,
  socketpairs, hardlinks, symlinks, IPv6, batchfiles, inplace,
  append, ACLs, xattrs, iconv, symtimes, prealloc
rsync comes with ABSOLUTELY NO WARRANTY. This is free software, and you
are welcome to redistribute it under certain conditions. See the GNU
General Public Licence for details.
anabuenrúa@m1-anabuenrúa:~$ _
```

Con chown cambiamos el propietario de la carpeta /var/www/ ejecutando el comando (10)

Figura 10: Cambiamos el propietario de la carpeta /var/www/



```
m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ sudo chown anabuenrúa:anabuenrúa -R /var/www
anabuenrúa@m1-anabuenrúa:~$
```

Y ejecutamos rsync en m1, para copiar los archivos a m2, como se muestra en (11):

Las opciones usadas son -a, que indica recursividad (archive), -e especifica el shell remoto que se va a utilizar, -v es verbose, para dar más información y -z para comprimir los archivos durante la transferencia.

2.1 OPCIONES AVANZADAS

Como opciones avanzadas vamos a usar --stats, que nos muestra estadísticas, --exclude, para excluir carpetas o directorios, --delete, para borrar en la máquina destino los ficheros borrados de la máquina origen y --dry-run, que permite a rsync hacer un

Figura 11: Sincronización de la carpeta /var/www/ de m1 a m2

```
m1-anabuenrúa (tras arreglar config red) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ ls
directorío directorio2.tgz envío_avanzado
anabuenrúa@m1-anabuenrúa:~$ ls -la /var/www/
total 16
drwxr-xr-x  4 anabuenrúa anabuenrúa 4096 mar  9 12:42 .
drwxr-xr-x 14 root      root      4096 mar  8 19:03 ..
drwxr-xr-x  2 anabuenrúa anabuenrúa 4096 mar  9 12:43 html
drwxr-xr-x  3 anabuenrúa anabuenrúa 4096 mar  9 17:31 prueba
anabuenrúa@m1-anabuenrúa:~$ rsync /var/www/ -avze ssh 192.168.56.102:/var/www
sending incremental file list
html/swap.html
prueba/
prueba/public_html/
prueba/public_html/index.html

sent 479 bytes  received 63 bytes  361.33 bytes/sec
total size is 11,088  speedup is 20.46
anabuenrúa@m1-anabuenrúa:~$
```

clonado de prueba, de forma que podemos ver lo que se va a clonar pero sin llegar a efectuarse la copia.

Comenzamos creando un directorio de prueba a clonar desde m1 a m2 (12).

Figura 12: Creamos directorio de prueba para clonar usando rsync.

```
m1-anabuenrúa (final sesion 1 p2 (rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ ls
fichero1.txt fichero2.txt nomandar
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ _
```

Comenzamos realizando una prueba de lo que sería la copia con --dry-run, como mostramos en (13).

Figura 13: Ejecución de rsync don --dry-run

```
m1-anabuenrúa (final sesion 1 p2 (rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~/p2$ rsync ./prueba_rsync/ -avz --stats --delete --exclude=**/nomandar --dry-run -e ssh 192.168.56.102:/p2/envío/
sending incremental file list
./
fichero1.txt
fichero2.txt

Number of files: 3 (reg: 2, dir: 1)
Number of created files: 2 (reg: 2)
Number of deleted files: 0
Number of regular files transferred: 2
Total file size: 0 bytes
Total transferred file size: 0 bytes
Literal data: 0 bytes
Matched data: 0 bytes
File list size: 0
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 148
Total bytes received: 32

sent 148 bytes  received 32 bytes  120.00 bytes/sec
total size is 0  speedup is 0.00 (DRY RUN)
anabuenrúa@m1-anabuenrúa:~/p2$ _
```

Así, comprobamos que en efecto se va a mandar lo que queremos, pero todavía no hemos clonado nada, como podemos comprobar en la máquina m2, se puede ver en (14)

Figura 14: Estado de m2 tras la ejecución de rsync con `--dry-run`

```
m2-anabuenruea (final sesion 1 p2 (tras rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenruea@m2-anabuenruea:~/p2$ cd envio
anabuenruea@m2-anabuenruea:~/p2/envio$ ls
anabuenruea@m2-anabuenruea:~/p2/envio$ _
```

Ahora sí, procedemos a realizar el envío quitando la opción `--dry-run`, en (15), y comprobamos que se ha copiado con éxito en (16)

Figura 15: Ejecución de rsync con opciones avanzadas.

```
m1-anabuenruea (final sesion 1 p2 (rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Number of regular files transferred: 2
Total file size: 0 bytes
Total transferred file size: 0 bytes
Literal data: 0 bytes
Matched data: 0 bytes
File list size: 0
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 148
Total bytes received: 32

sent 148 bytes  received 32 bytes  120.00 bytes/sec
total size is 0  speedup is 0.00 (DRY RUN)
anabuenruea@m1-anabuenruea:~/p2$ rsync ./prueba_rsync/ -avz --stats --delete --exclude=**/nomandar -e
ssh 192.168.56.102:~/p2/envio/
sending incremental file list
./
fichero1.txt
fichero2.txt

Number of files: 3 (reg: 2, dir: 1)
Number of created files: 2 (reg: 2)
Number of deleted files: 0
Number of regular files transferred: 2
Total file size: 0 bytes
Total transferred file size: 0 bytes
Literal data: 0 bytes
Matched data: 0 bytes
File list size: 0
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 214
Total bytes received: 64

sent 214 bytes  received 64 bytes  556.00 bytes/sec
total size is 0  speedup is 0.00
anabuenruea@m1-anabuenruea:~/p2$
```

Figura 16: Comprobando la copia correcta de los ficheros en m2.

```
m2-anabuenruea (final sesion 1 p2 (tras rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenruea@m2-anabuenruea:~/p2/envio$ ls
fichero1.txt  fichero2.txt
anabuenruea@m2-anabuenruea:~/p2/envio$ _
```

Es claro que el argumento `--exclude` ha evitado que se copie el directorio `nomandar`. Finalmente, probamos a eliminar el fichero `fichero1.txt` y repetir el clonado, comprobando así que la opción `--delete` lo elimina en m2 también, como se ve en (17)

Figura 17: Ejecución de rsync tras borrar fichero1 en m1 y el resultado en m2

```

m1-anabuenrúa (final sesion 1 p2 (rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenrúa@m1-anabuenrúa:~/p2$ cd prueba_rsync/
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ ls
fichero1.txt fichero2.txt nomandar
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ rm fichero1.txt
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ ls
fichero2.txt nomandar
anabuenrúa@m1-anabuenrúa:~/p2/prueba_rsync$ cd ..
anabuenrúa@m1-anabuenrúa:~/p2$ rsync ./prueba_rsync/ -avz --stats --delete --exclude=**/nomandar -e
ssh 192.168.56.102:~/p2/envio/
sending incremental file list
deleting fichero1.txt
./

Number of files: 2 (reg: 1, dir: 1)
Number of created files: 0
Number of deleted files: 1 (reg: 1)
Number of regular files transferred: 0
Total file size: 0 bytes
Total transferred file size: 0 bytes
Literal data: 0 bytes
Matched data: 0 bytes
File list size: 0
File list generation time: 0.001 seconds
File list transfer time: 0.000 seconds
Total bytes sent: 122
Total bytes received: 42

sent 122 bytes received 42 bytes 109.33 bytes/sec
total size is 0 speedup is 0.00
anabuenrúa@m1-anabuenrúa:~/p2$

m2-anabuenrúa (final sesion 1 p2 (tras rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
anabuenrúa@m2-anabuenrúa:~/p2/envio$ ls
fichero2.txt
anabuenrúa@m2-anabuenrúa:~/p2/envio$

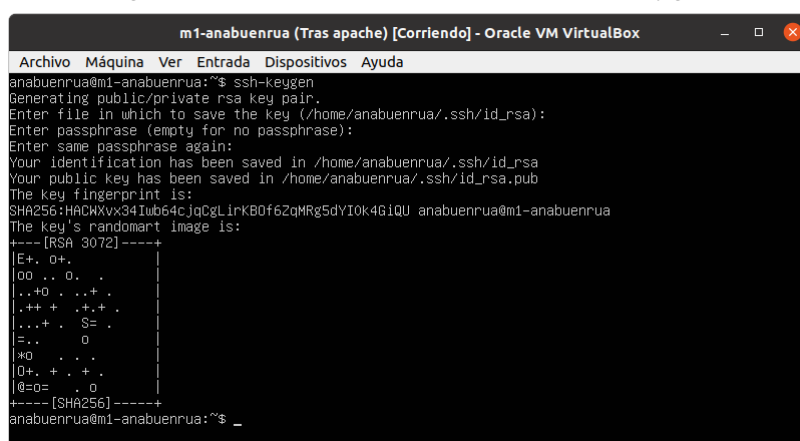
```

ACCESO MEDIANTE SSH SIN CONTRASEÑA

El acceso por ssh sin introducir la contraseña manualmente ya se configuró en la práctica anterior.

Para ello, generamos en cada máquina una clave pública y privada, con `ssh-keygen`, como se muestra en (18).

Figura 18: Generación de claves con `ssh-keygen`



```
m1-anabuenrúa (Tras apache) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/anabuenrúa/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/anabuenrúa/.ssh/id_rsa
Your public key has been saved in /home/anabuenrúa/.ssh/id_rsa.pub
The key's fingerprint is:
SHA256:HACWVx34Iub64cJgCgLiK80f6ZqMRg5dYI0k4GiQU anabuenrúa@m1-anabuenrúa
The key's randomart image is:
----[RSA 3072]-----
E+. 0+.
00 .. 0. .
..+0 .. .+.
..++ + ..++
...+ . S= .
...+ . 0
.. 0
.. 0
0+. +. +.
@=0= . 0
----[SHA256]-----
anabuenrúa@m1-anabuenrúa:~$ _
```

Después compartimos las claves públicas de una máquina a otra con `ssh-copy-id -p 2022 anabuenrúa@192.168.56.102` (de m2 a m1) y `ssh-copy-id anabuenrúa@192.168.56.102` (de m1 a m2). El caso de m2 a m1 puede verse en (19)

3.1 OPCIONES AVANZADAS

Cuando se realizó, se dejaron todas las opciones por defecto, pero se pueden usar algunos argumentos para modificar el comportamiento:

- `-t`: Especifica el tipo de clave que se va a generar, por ejemplo `rsa`.
- `-b`: Indica el número de bits en la clave, por defecto es 2048.
- `-f`: Especifica el archivo de la clave.
- `-l`: No se usa al generar las claves, si no que se usa para ver el fingerprint de una clave pública.

Figura 19: Envío de claves públicas mediante ssh-copy-id

```

m2-anabuenrúa (Tras red e instalacion) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m2-anabuenrúa:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/anabuenrúa/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/anabuenrúa/.ssh/id_rsa
Your public key has been saved in /home/anabuenrúa/.ssh/id_rsa.pub
The key's fingerprint is:
SHA256:+kznJovYd/LlMDyRQCgZt7MpDHemYUw7RA8tNo3a0s8 anabuenrúa@m2-anabuenrúa
The key's randomart image is:
+----[RSA 3072]-----+
|  .*. . . |
| +B=+O    |
| .+B+* .  |
| .+. * + . |
| .+ O S O  |
| O . . . . |
| + . . * . |
| Eo =+O*   |
| . O.+Bo . |
+----[SHA256]-----+
anabuenrúa@m2-anabuenrúa:~$ ssh-copy-id -p 2022 anabuenrúa@192.168.56.101
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/anabuenrúa/.ssh/id_rsa.pub"
/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 prompted now it is to install all the new keys
anabuenrúa@192.168.56.101's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh -p '2022' 'anabuenrúa@192.168.56.101'"
and check to make sure that only the key(s) you wanted were added.

anabuenrúa@m2-anabuenrúa:~$

```

- -v: Verbose.

Ejemplos de uso de estos argumentos son `ssh-keygen -t rsa -b 4096` o (20).

Figura 20: Opciones avanzadas de ssh-keygen

```

m1-anabuenrúa (final sesion 1 p2 (rsync)) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ ssh-keygen -t rsa -b 4096
Enter file in which the key is (/home/anabuenrúa/.ssh/id_rsa):
3072 SHA256:HACXvx34Iub64cJqCgLiRKB0f6ZqMRg5dY10k4GIQU anabuenrúa@m1-anabuenrúa (RSA)
anabuenrúa@m1-anabuenrúa:~$ ssh-keygen -t rsa -b 4096 -v
Enter file in which the key is (/home/anabuenrúa/.ssh/id_rsa):
3072 SHA256:HACXvx34Iub64cJqCgLiRKB0f6ZqMRg5dY10k4GIQU anabuenrúa@m1-anabuenrúa (RSA)
+----[RSA 3072]-----+
| E+. O+. |
| 00 .. O. |
| ..+O . .+ |
| .++ + .+. |
| ...+ . S= |
| =..       |
| *O . . . |
| O+. +. +. |
| @=O= . O  |
+----[SHA256]-----+
anabuenrúa@m1-anabuenrúa:~$

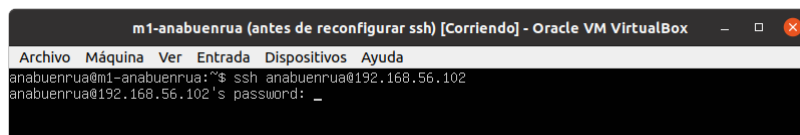
```

Si al generar la clave no usamos la ruta por defecto, para mandarla con `ssh-copy-id` debemos especificar la ruta de la clave pública con `-i`, al igual que al acceder se especifica la de la clave privada con `-i` en `ssh`.

3.2 COPIA DE CLAVE MANUAL

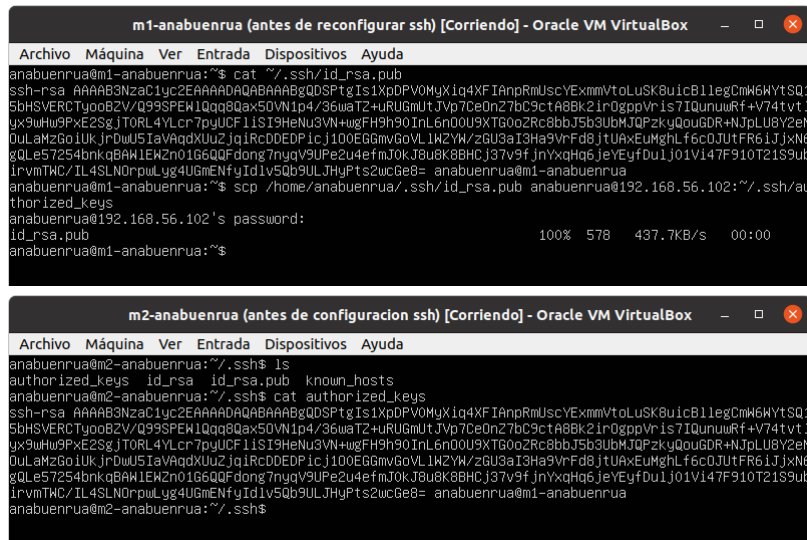
Comenzamos en `m2`, accediendo a `~/.ssh/authorized_keys`, donde está escrita la clave pública de `m1`. Editamos este fichero con `nano` borrando su contenido y comprobamos que ahora para acceder a `m2` desde `m1` nos pide contraseña, (21)

Figura 21: Comprobación de que nos requiere contraseña para conectar mediante `ssh` tras borrar la clave previamente guardada.



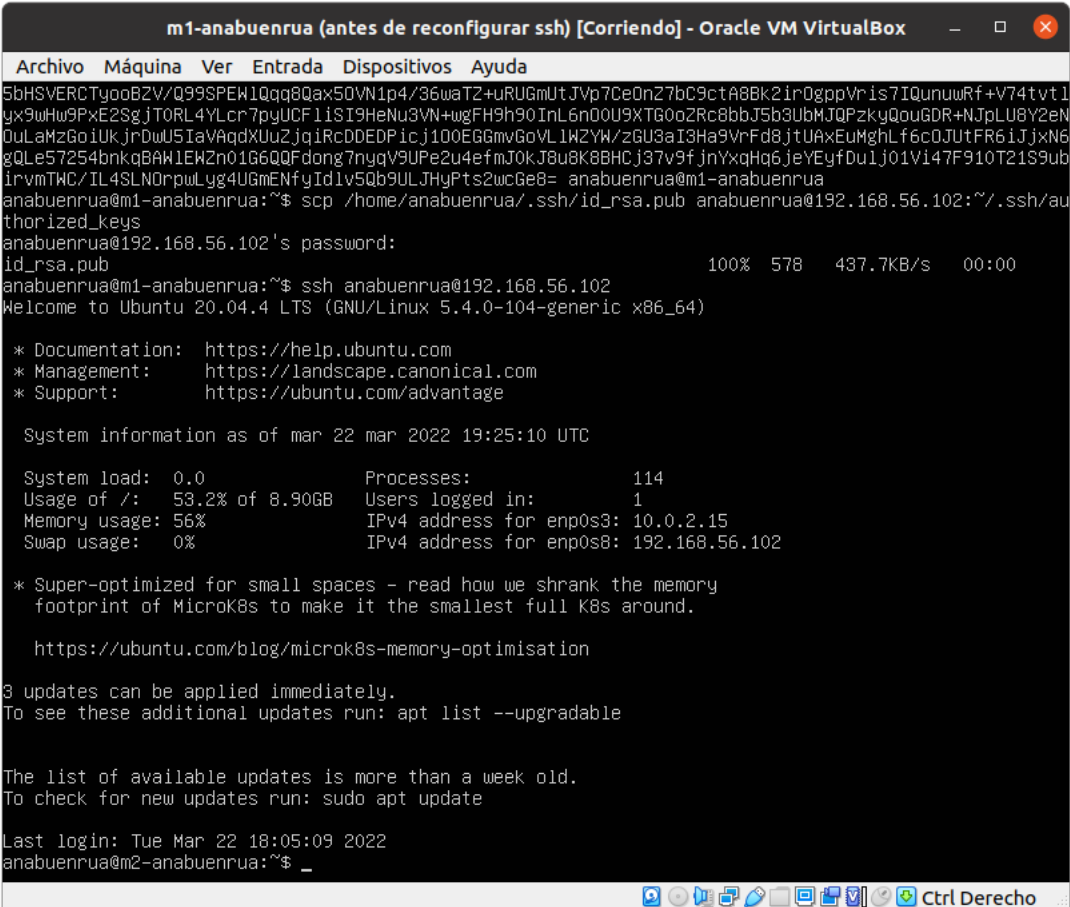
Para volver a tener acceso sin contraseña, vamos a mandar nuestra clave pública a `m2`. Para ello copiamos la clave pública que se encuentra en `~/.ssh/id_rsa.pub` de `m1` mediante `scp` en el archivo `~/.ssh/authorized_keys` de `m2`, como vemos en (22).

Figura 22: Visualización y envío de la clave pública mediante `scp` y comprobación en la máquina `m2` de que se ha realizado correctamente.



Finalmente comprobamos que nos podemos conectar de `m1` a `m2` sin contraseña de nuevo conectándonos por `ssh` como en (23).

Figura 23: Conexión por ssh sin requerir introducir la contraseña.



```
m1-anabuenruea (antes de reconfigurar ssh) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
SbH5VERCTyooB2V/Q99SPEW1Qqq8Qax50VN1p4/36waT2+uRUGmUtJVp7Ce0nZ7bC9ctA8Bk2ir0gppVris7IQunuwRf+v74tvt1
yx9uHw9PxE2Sgjt0RL4YLCr7pyUCF1iSI9HeNu3VN+wgFH9h90InL6n00U9XTG0oZRC8bbJ5b3UbMJQPzkyQouGDR+NjPLU8Y2eN
OulaMzGoiUkjrDwU5IaVAqdXUuZjqIRcDDEDPicj100EGGmvGovL1WZYW/zGU3aI3Ha9VrFd8jtUAxEuMghLf6cDJUtFR6iJjxN6
gQLe57254bnkqBAW1EWZn01G6QQFdong7nyqV9UPe2u4efmJ0kJ8u8K8BHCj37v9fjnYxqHq6jeYEyDu1j01V147F910T21S9ub
irvmTWC/IL4SLN0rpuLYg4UGmENfyIdlv5Qb9ULJHyPts2wcGe8= anabuenruea@m1-anabuenruea
anabuenruea@m1-anabuenruea:~$ scp /home/anabuenruea/.ssh/id_rsa.pub anabuenruea@192.168.56.102:~/.ssh/au
thorized_keys
anabuenruea@192.168.56.102's password:
id_rsa.pub 100% 578 437.7KB/s 00:00
anabuenruea@m1-anabuenruea:~$ ssh anabuenruea@192.168.56.102
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-104-generic x86_64)

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

System information as of mar 22 mar 2022 19:25:10 UTC

System load: 0.0          Processes: 114
Usage of /:  53.2% of 8.90GB Users logged in: 1
Memory usage: 56%         IPv4 address for enp0s3: 10.0.2.15
Swap usage:  0%           IPv4 address for enp0s8: 192.168.56.102

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

   https://ubuntu.com/blog/microk8s-memory-optimisation

3 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

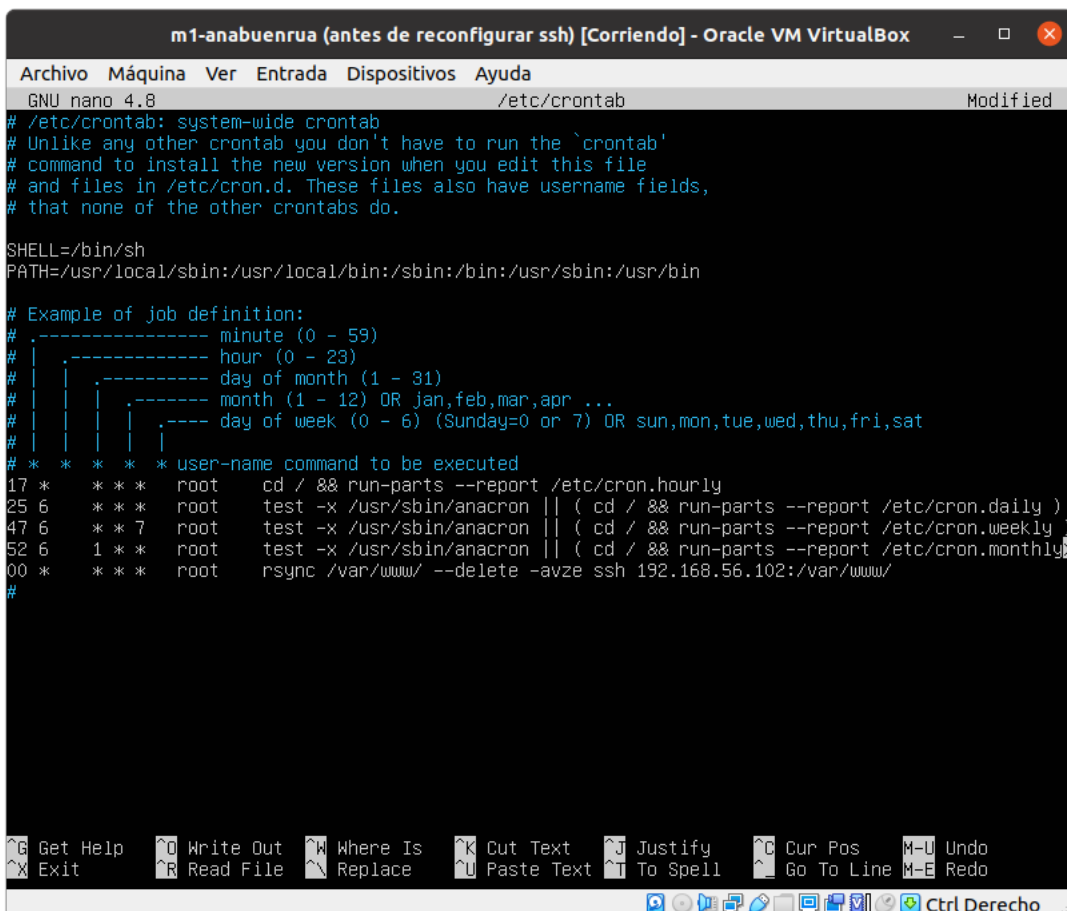
Last login: Tue Mar 22 18:05:09 2022
anabuenruea@m2-anabuenruea:~$ _
```

USANDO CRONTAB

Comenzamos añadiendo una tarea que sincronice completamente las carpetas /var/www/ de m1 y de m2 cada hora.

Para conseguirlo, usamos crontab para programar la ejecución del comando de rsync cada hora, editando el fichero /etc/crontab como sigue en (24).

Figura 24: Configuración del fichero /etc/crontab para programar la sincronización con rsync cada hora.



```
m1-anabuenrúa (antes de reconfigurar ssh) [Corriendo] - Oracle VM VirtualBox
GNU nano 4.8 /etc/crontab Modified
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
00 * * * * root rsync /var/www/ --delete -avze ssh 192.168.56.102:/var/www/
#

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^_ Replace ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
Ctrl Derecho
```

4.1 OPCIONES AVANZADAS

Como opciones avanzadas, tenemos que mientras `*` es para cualquier valor y se pueden especificar varios valores concretos separados por `,`, hay formas más fáciles de especificar cuándo ejecutar ciertas tareas.

Por ejemplo, `-` indica un rango, y `/` el paso o salto. Así, si añadimos la siguiente tarea para escribir "hola" en un fichero `cronprueba.log` cada 2 horas los días 1,2 y 3 (de 1 a 3) sería editando el fichero `/etc/crontab` como se ve en (25)

Figura 25: Configuración del fichero `/etc/crontab` para escribir hola en un fichero los días 1,2 y 3 cada 2 horas.

```

m1-anabuenrúa (antes de reconfigurar ssh) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 4.8 /etc/crontab Modified
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
00 * * * * root rsync /var/www/ --delete -avze ssh 192.168.56.102:/var/www/
00 0-23/2 1-3 * * root echo "hola" >> cronprueba.log
#
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify    ^C Cur Pos   M-U Undo
^X Exit      ^R Read File  ^_ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line M-E Redo
Ctrl Derecho

```

```
00 0-23/2 1-3 * * root echo "hola" >> cronprueba.log
```

Vemos que hemos especificado los minutos a `00` y en las de horas de `0` a `23` cada `2`, en los días `1` a `3`.

BIBLIOGRAFÍA

- Diapositivas y gui3n de la pr3ctica.
- <https://www.thegeekstuff.com/2008/11/3-steps-to-perform-ssh-login-without-password-using-ssh-keygen-ssh-copy-id/>
- <https://linux.die.net/man/1/rsync>
- <https://linux.die.net/man/1/scp>
- <https://crontab.guru/>