
Clonando la información de un sitio web

Andrés Millán Muñoz (amilmun@correo.ugr.es)

March 23, 2022

Contents

1	Copiando archivos de una máquina a otra	2
1.1	Parámetros de scp	2
1.2	Parámetros de tar	3
2	Uso de Rsync	4
2.1	Parámetros de rsync	5
2.1.1	Excluir archivos o subdirectorios	5
2.1.2	Eliminar archivos que no existen en el destino	5
2.1.3	Otros parámetros	5
3	SSH	6
3.1	Parámetros	7
3.1.1	ssh-keygen	7
3.1.2	ssh-copy-id	8
4	Automatizando la copia con Crontab	9
4.1	Parámetros de crontab	10
5	Referencias	11

En esta práctica vamos a automatizar la copia de archivos de una máquina virtual a otra. Para ello, usaremos las herramientas SSH, rsync y crontab. Veremos cómo configurar los respectivos programas para llevar a cabo esta tarea.

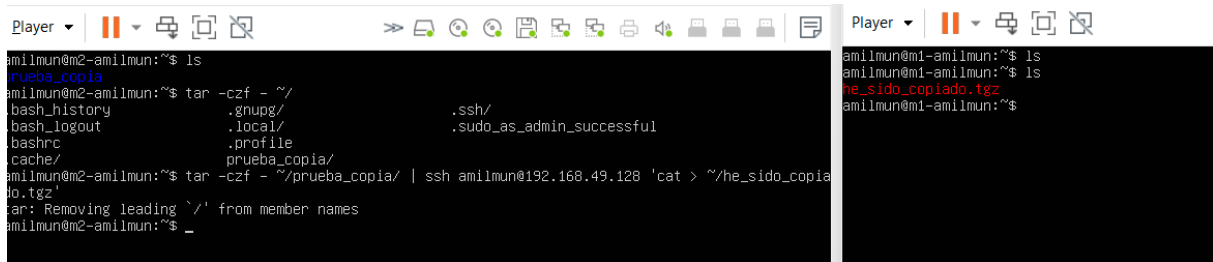
Nota: las IPs de las máquinas son 192.168.49.128 (M1) y 192.168.49.129 (M2).

1 Copiando archivos de una máquina a otra

Hay varias formas de traspasar archivos de una máquina a otra.

Una de ellas es usando pipelines: se comprime un directorio y se redirige la salida a ssh mediante un pipeline, pasando el directorio comprimido de una máquina a otra. La sintaxis sería

```
1 tar -czf - {/path/a/directorio} | ssh {usuario}@{ip} 'cat > {/ruta/que/querramos}/{nombre_archivo}.tgz'
```



Otra opción es hacerlo en dos pasos: primero, comprimimos el directorio, y luego lo mandamos mediante `scp`:

```
1 tar -czvf {archivo_a_enviar}.tgz {/path/a/directorio}
2 scp {archivo_a_enviar}.tgz {usuario}@{ip}:{/ruta}/{archivo_recibido}.tgz
```

Podemos hacerlo del tirón con

```
1 scp -r directorio {usuario}@{ip}:{/ruta/destino/}
```



1.1 Parámetros de scp

Aunque no nos serán necesarios, `scp` tiene varios parámetros útiles. Por ejemplo:

- `-P` especifica el puerto en el host remoto.
- `-v` activa el modo verboso.
- `-o {ssh_options}` permite especificar algunos parámetros de ssh. Alguno de estos son la compresión, el timeout, puerto, usuario, etc...
- `-C` activa la compresión. Se lo pasa a ssh.

1.2 Parámetros de tar

Al igual que con `scp`, no nos importan demasiado los diferentes parámetros de `tar`. Sin embargo, este programa es muy complejo, y permite cambiar su comportamiento de maneras muy diferentes. Entre las opciones más destacables, se encuentran:

- `c`: crea un nuevo archivo comprimido.
- `v`: muestra verbosamente el progreso.
- `f {nombre_de_archivo}`: especifica el nombre del archivo.
- `x`: extrae el comprimido.
- `u` añade archivos, pero solo los que sean más nuevos que los que se encuentran en el comprimido.
- `--exclude`: excluye los archivos que se especifiquen.

Se puede encontrar un uso de la herramienta resumido [en este enlace](#), así como en [tldr pages](#)

2 Uso de Rsync

Rsync viene instalado por defecto, así que no será necesario ponerlo a mano.

```
amilmun@m1-amilmun:~$ rsync --version
rsync version 3.1.2 protocol version 31
Copyright (C) 1996-2015 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

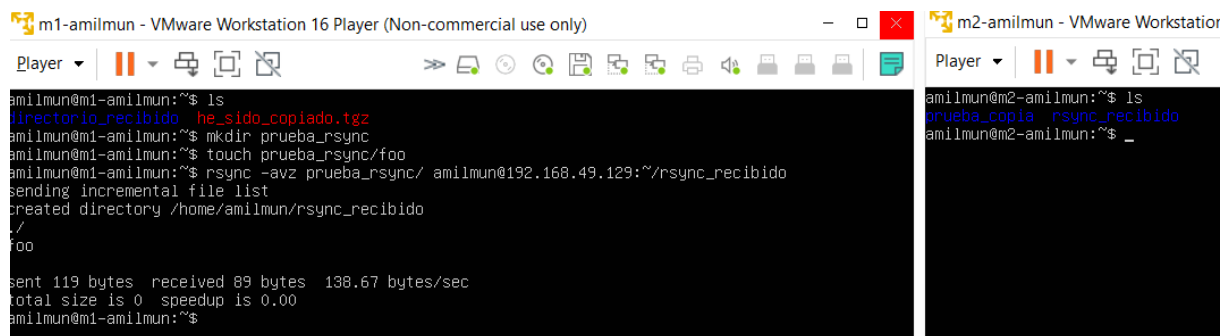
rsync comes with ABSOLUTELY NO WARRANTY. This is free software, and you
are welcome to redistribute it under certain conditions. See the GNU
General Public Licence for details.
amilmun@m1-amilmun:~$
```

Operaremos en nuestro usuario llamado `amilmun`, así que, para facilitarnos el trabajo, vamos a darnos permisos sobre la carpeta `/var/www`:

```
1 sudo chown -R amilmun:amilmun /var/www
```

Como dice el guion, para probar su funcionamiento, vamos a copiar un directorio de prueba de M1 a M2. Para ello, hacemos

```
1 rsync -avz {/path/a/dir} {usuario}@{ip}:{path/a/dir}
```



La opción `-avz` se usa para transferir un [a]rchivo comprimido ([z]ipped) en modo [v]erboso.

2.1 Parámetros de rsync

2.1.1 Excluir archivos o subdirectorios

A veces, nos interesará omitir algunos subdirectorios de la ruta que queramos copiar. Para conseguirlo, podemos hacer

```
1 rsync -avz --exclude {/path/a/subdir} {/path/a/dir/dir/} {usuario}@{ip}
   :{ruta/destino}
```

2.1.2 Eliminar archivos que no existen en el destino

Si queremos una sincronización pura y dura, obviando cualquier archivo que existiera en el destino previamente, usamos `--delete`:

```
1 rsync -avz {/path/a/dir} --delete {usuario}@{ip}:{ruta/destino}
```

2.1.3 Otros parámetros

Sin ningún orden en particular, aquí presento una lista de algunas opciones interesantes:

- `-hP` muestra un [p]rogreso más entendible por un [h]umano.
- `L` resuelve *softlinks*.
- `u` solo transfiere archivos más nuevos que los presentes.
- Para cambiar de puerto de `ssh`, puedes usar `-e 'ssh -p {puerto}'`.
- `--dry-run` simula la transferencia pero no la realiza. De esta manera, podemos comprobar que nuestra configuración es correcta.

3 SSH

Aunque hice la configuración de SSH en la práctica anterior, parece que no funciona correctamente. Creo que es debido a que sobrescribí las claves al mandarlas de una máquina a otra. Así que, para solucionarlo, lo vuelvo a hacer de nuevo.

```
1 ssh-keygen -b 4096 -t rsa
2 # Nos damos permisos de escritura como dice el guion
3 chmod 700 ~/.ssh/authorized_keys
4 ssh-copy-id 192.168.49.129 # M1 -> M2
5 ssh-copy-id 192.168.49.128 # M2 -> M1
```

```
all the new keys
amilmun@192.168.49.129's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh '192.168.49.129'"
and check to make sure that only the key(s) you wanted were added.

amilmun@m1-amilmun:/tmp$ ssh 192.168.49.129
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 4.15.0-171-generic x86_64)

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

System information as of Wed Mar 16 17:28:08 UTC 2022

System load:  0.0               Processes:            171
Usage of /:   44.6% of 8.79GB   Users logged in:     1
Memory usage: 41%              IP address for ens33: 192.168.121.131
Swap usage:   0%               IP address for ens37: 192.168.49.129

 * 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

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

New release '20.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Wed Mar 16 17:15:35 2022
amilmun@m2-amilmun:~$
```

Se puede evitar usar `ssh-copy-id` copiando manualmente la clave en el servidor. La siguiente forma

utiliza `cat` para lograrlo:

```
1 cat ~/.ssh/id_rsa.pub | ssh {usuario}@{host-remoto} 'cat >> ~/.ssh/authorized_keys'
```

Alternativamente, `scp` consigue lo mismo:

```
1 scp ~/.ssh/id_rsa.pub {usuario}@{host-remoto}:~/.ssh/authorized_keys
```

Nosotros no necesitaremos usar ninguna de estas dos formas. A fin de cuentas, `ssh-copy-id` encapsula `scp` para hacerlo más intuitivo.

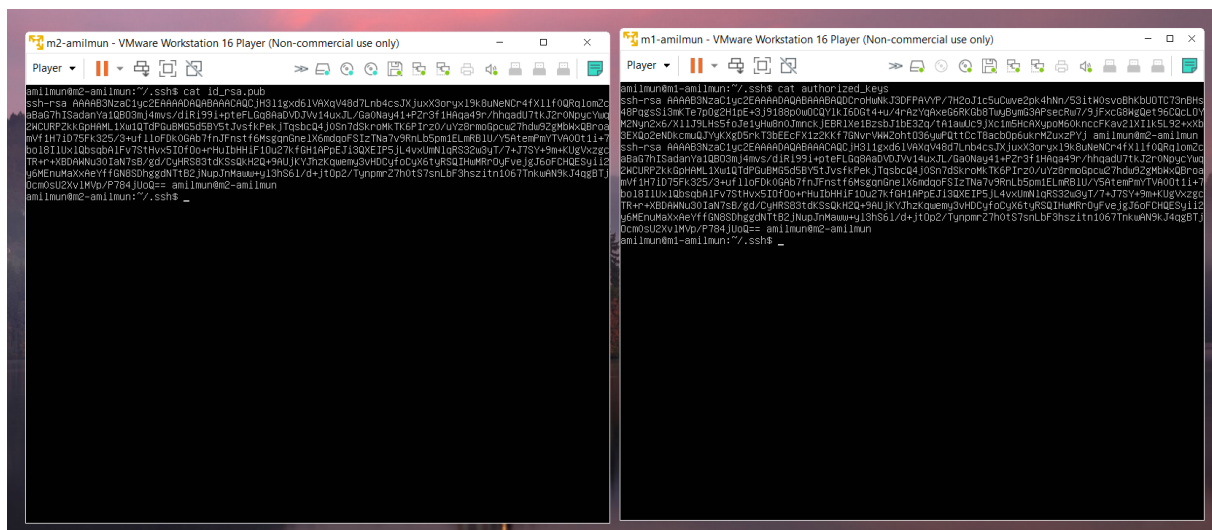


Figure 3.1: Podemos ver que el archivo `authorized_keys` es, en esencia, una copia de cierta clave pública. Usando el pipe de `cat`, añadiríamos al final del archivo nuestra clave. De hecho, en la captura de pantalla se pueden ver los restos de la práctica anterior.

3.1 Parámetros

3.1.1 ssh-keygen

Este programa permite una cantidad de parámetros importante. Se puede modificar considerablemente su comportamiento. No obstante, la mayoría de ellos están enfocados a la seguridad, como es natural. Algunos de estos son:

- `-f {nombre_de_archivo}`: especifica el nombre del archivo.
- `-p -f {nombre_de_archivo}`: cambia la passphrase de una clave.

- `-t {algoritmo} -a {rondas}`: especifica el tipo de clave. Por ejemplo, `-t ed25519 -a 100` genera una clave usando ed25519 con 100 rondas.
- `-b {bits}`: especifica el número de bits. En los comandos de arriba, usamos `-b 4096`, lo cual dice que tiene 4096 bits.
- Se pueden añadir comentarios con `-C {comentario}`.
- `-R {host_remoto}` permite eliminar las claves de un host. Esto puede ser útil si se sabe que las claves han sido actualizadas.

3.1.2 ssh-copy-id

Este programa no tiene tantos parámetros útiles. Los dos que más destacan son:

- `-i {certificado}`: copia el certificado pasado como argumento en vez de tus propias claves.
- `-p`: especifica el puerto.

4 Automatizando la copia con Crontab

Crontab es una herramienta dedicada a la ejecución de programas en una determinada franja de tiempo. Cada minuto, revisará su archivo de configuración, vigilando si ha llegado el momento correcto de ejecutar alguna orden específica. El archivo de configuración se encuentra en `/etc/crontab`.

Nosotros usaremos crontab desde M1 para sincronizar los contenidos de la carpeta `/var/www`, duplicando así la página web en M2. La página crontab.guru resulta muy útil para configurar el programa. Ahí se pueden encontrar las diferentes formas de especificar la fecha.

Añadimos la línea

```
1 0 * * * * rsync -avz /var/www amilmun@192.168.49.129:/var/www
```

```
# /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

# m h dom mon dow user  command
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 )
#

amilmun@m1-amilmun:~$ ls
directorio_recibido  he_sido_copiado.tgz  prueba_rsync
amilmun@m1-amilmun:~$
```

4.1 Parámetros de crontab

De la misma manera que con los anteriores programas, vamos a listar una lista de parámetros interesantes de `crontab` sin ningún orden en particular:

- `-e`: edita el archivo de crontab para el usuario actual.
- `-e -u {usuario}`: igual que el anterior, pero especificando un usuario en particular.
- `crontab {archivo}`: reemplaza el archivo de crontab con el que se ha pasado como argumento.
- `-l` mira la lista de tareas de cron del usuario actual.
- `r`: elimina todas las tareas de cron del usuario actual.

5 Referencias

1. [Crontab.guru](https://crontab.guru)
2. [TLDR pages](https://tldr.sh/)
3. <https://linuxize.com/post/how-to-exclude-files-and-directories-with-rsync/>
4. <https://techdocs.akamai.com/netstorage/docs/use-secure-copy#secure-copy-scp-connection-options>
5. <https://www.tecmint.com/18-tar-command-examples-in-linux/>
6. <https://www.computerhope.com/unix/utar.htm>
7. <https://www.simplified.guide/ssh/copy-public-key>
8. Ana Buendía me ha informado sobre `--dry-run` de `rsync`.