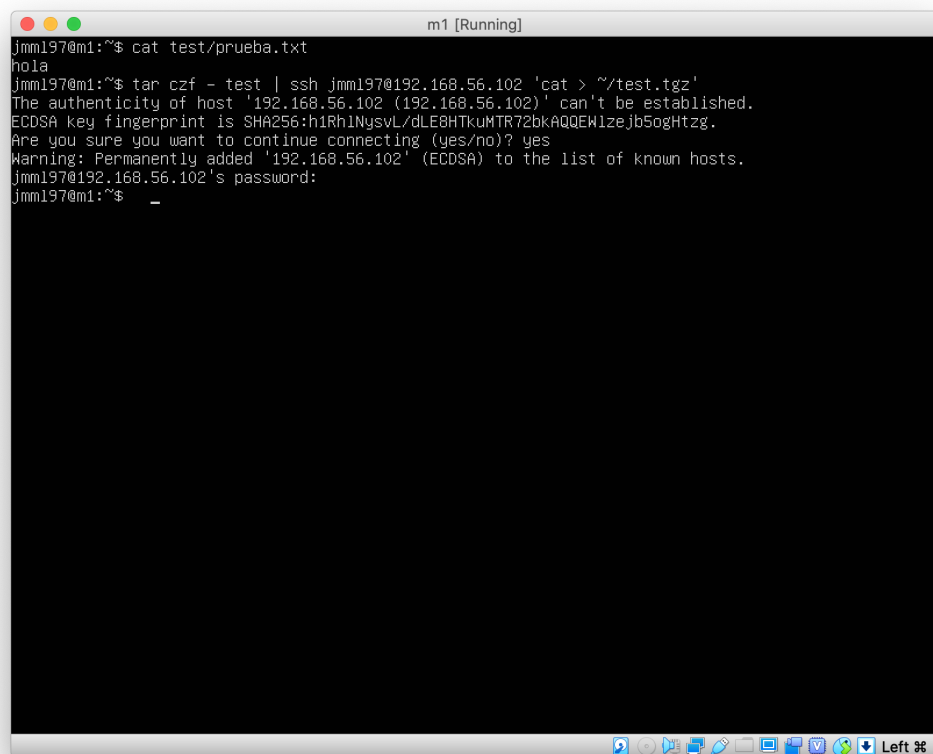


3 de abril de 2020

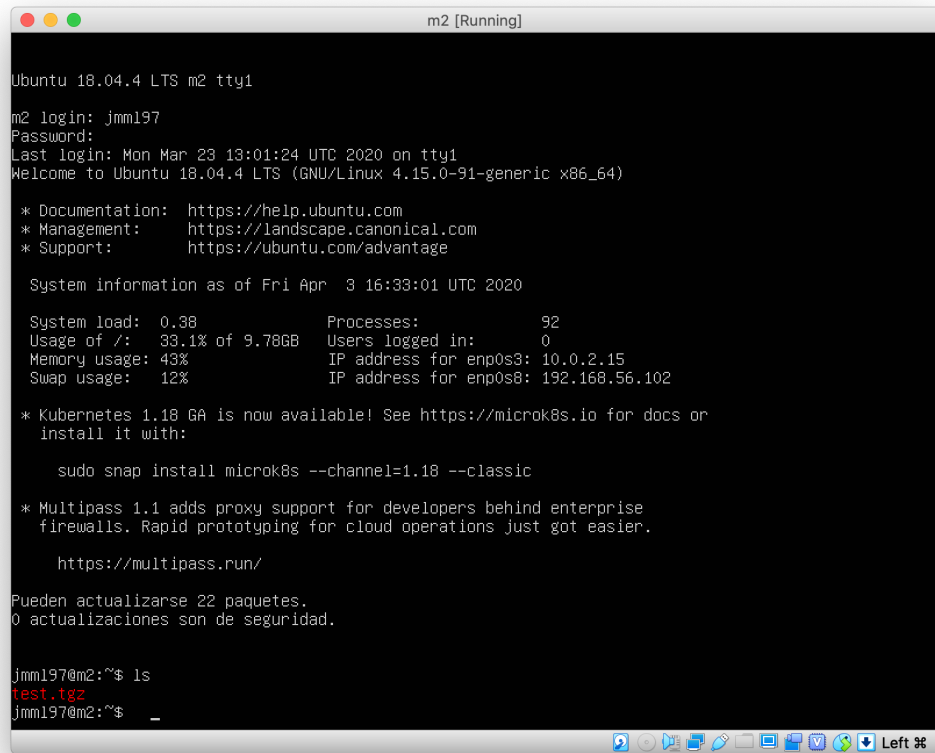
## I. COPIA POR SSH

Continuamos usando las máquinas m1 y m2 de la práctica anterior. Creamos un directorio en la máquina m1 llamado test y creamos un archivo prueba.txt. Vamos a copiarlo mediante ssh a la máquina m2 —comprimiéndolo primero mediante la orden tar—. En las imágenes siguientes comprobamos el proceso y el resultado de dicha copia.



```
m1 [Running]
jmm197@m1:~$ cat test/prueba.txt
hola
jmm197@m1:~$ tar czf - test | ssh jmm197@192.168.56.102 'cat > ~/test.tgz'
The authenticity of host '192.168.56.102 (192.168.56.102)' can't be established.
ECDSA key fingerprint is SHA256:h1Rh1NysvL/dLE8HTkuMTR72bkAQQEW1zejb5ogHtzg.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.102' (ECDSA) to the list of known hosts.
jmm197@192.168.56.102's password:
jmm197@m1:~$
```

FIGURA 1: Proceso de copia

A terminal window titled 'm2 [Running]' showing the login process for user 'jmm197' on an Ubuntu 18.04.4 LTS machine. The terminal displays the login prompt, password input, and a welcome message. It then shows system information including system load, disk usage, memory usage, swap usage, processes, users logged in, and IP addresses. A message about Kubernetes 1.18 GA is also present. The terminal ends with a command prompt where the user has typed 'ls' and 'test.tgz'.

```
Ubuntu 18.04.4 LTS m2 tty1
m2 login: jmm197
Password:
Last login: Mon Mar 23 13:01:24 UTC 2020 on tty1
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-91-generic x86_64)

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

System information as of Fri Apr  3 16:33:01 UTC 2020

System load:  0.38               Processes:           92
Usage of /:   33.1% of 9.78GB    Users logged in:    0
Memory usage: 43%              IP address for enp0s3: 10.0.2.15
Swap usage:   12%              IP address for enp0s8: 192.168.56.102

 * Kubernetes 1.18 GA is now available! See https://microk8s.io for docs or
   install it with:

   sudo snap install microk8s --channel=1.18 --classic

 * Multipass 1.1 adds proxy support for developers behind enterprise
   firewalls. Rapid prototyping for cloud operations just got easier.

   https://multipass.run/

Pueden actualizarse 22 paquetes.
0 actualizaciones son de seguridad.

jmm197@m2:~$ ls
test.tgz
jmm197@m2:~$
```

FIGURA 2: Resultado de la copia

## 2. CLONADO DE UNA CARPETA

Instalamos rsync en ambas máquinas mediante apt-get. Vamos a copiar el contenido de /var/www en m1 a m2. Ejecutamos la orden

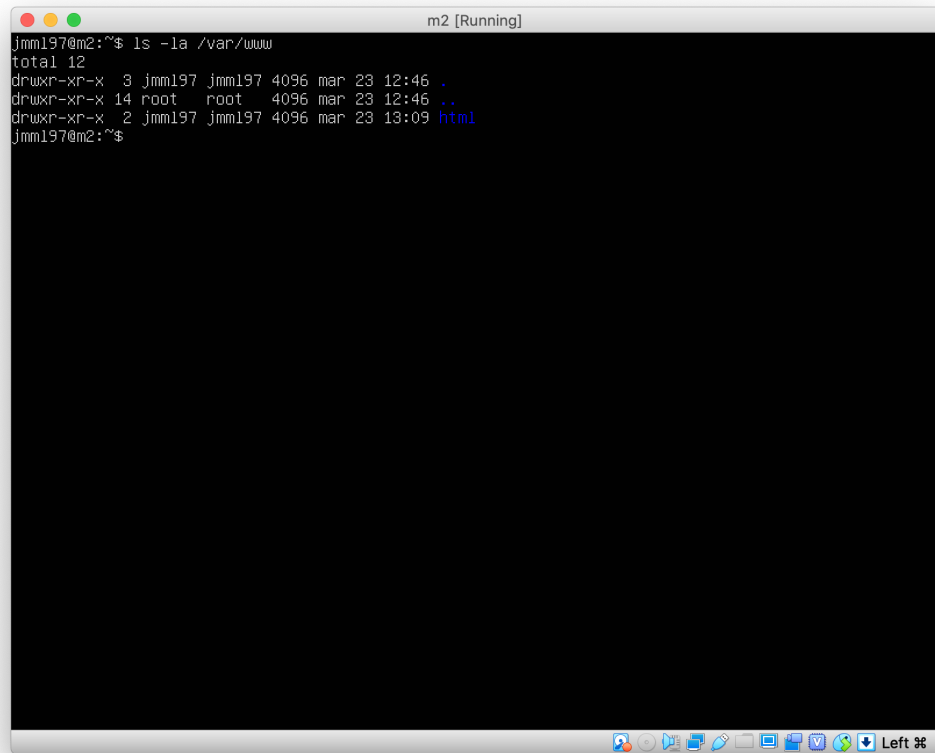
```
rsync -avz -e ssh 192.168.56.102:/var/www/ /var/www/
```

El resultado de esta orden en ambas máquinas puede observarse en las siguientes imágenes.

```
m1 [Running]
jmm197@m1:~$ rsync -avz -e ssh 192.168.56.102:/var/www/ /var/www/
opening connection using: ssh 192.168.56.102 rsync --server --sender -vvlogDtpre.iLsfxC --new-compre
ss . /var/www/ (9 args)
jmm197@192.168.56.102's password:
receiving incremental file list
delta-transmission enabled
./
html/
html/index.html
html/prueba.txt
total: matches=16 hash_hits=16 false_alarms=0 data=40

sent 169 bytes  received 333 bytes  91.27 bytes/sec
total size is 10,958  speedup is 21.83
jmm197@m1:~$ ls -la /var/www
total 12
drwxr-xr-x  3 jmm197 jmm197 4096 mar 23 12:46 .
drwxr-xr-x 14 root   root   4096 mar 23 11:34 ..
drwxr-xr-x  2 jmm197 jmm197 4096 mar 23 13:09 html
jmm197@m1:~$ _
```

FIGURA 3: Sincronización mediante rsync

A terminal window titled 'm2 [Running]' showing the output of the command 'ls -la /var/www'. The output lists three directories: '.', '..', and 'html', each with permissions 'drwxr-xr-x', owned by 'jmm197', size '4096', and modified on 'mar 23'. The 'html' directory was modified at '13:09'. The prompt 'jmm197@m2:~\$' is visible at the bottom of the terminal output.

```
jmm197@m2:~$ ls -la /var/www
total 12
drwxr-xr-x  3 jmm197 jmm197 4096 mar 23 12:46 .
drwxr-xr-x 14 root   root   4096 mar 23 12:46 ..
drwxr-xr-x  2 jmm197 jmm197 4096 mar 23 13:09 html
jmm197@m2:~$
```

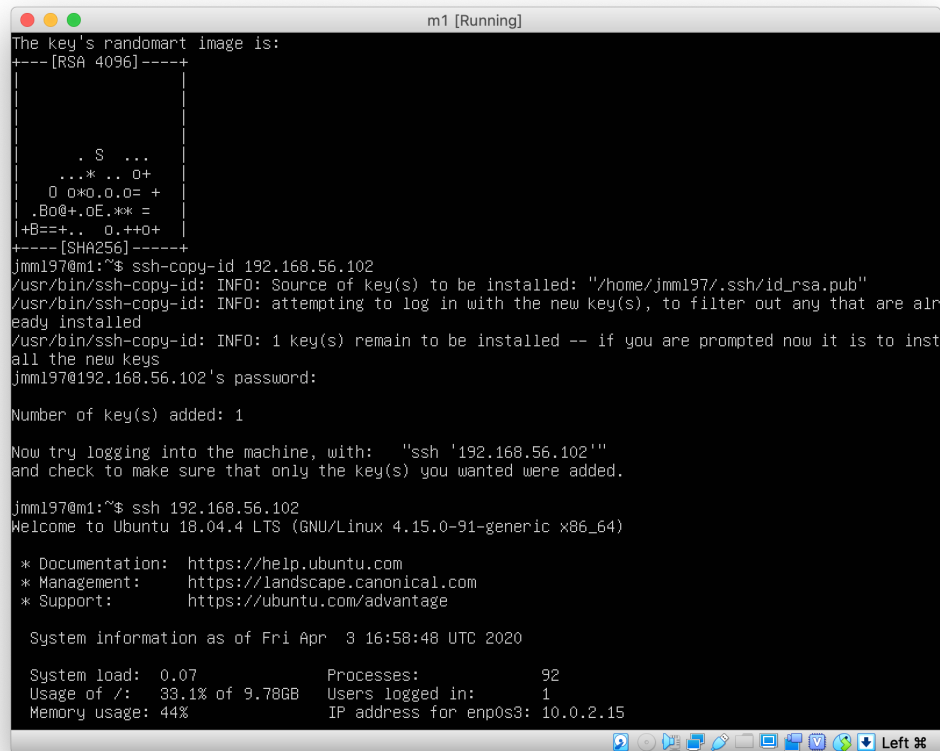
FIGURA 4: Resultado de la copia mediante rsync

### 3. SSH SIN CONTRASEÑA

Generamos las claves en ambas máquinas mediante `ssh-keygen` con el siguiente comando:

```
ssh-keygen -b 4096 -t rsa
```

Dejamos el campo *passphrase* en blanco. Copiamos las claves mediante `ssh-copy-id`. En la siguiente imagen podemos ver todo este proceso en la máquina m1, así como el acceso a la máquina m2 sin contraseña.

A terminal window titled 'm1 [Running]' showing the process of installing an SSH key and logging into a remote machine. The user 'jmm197' runs 'ssh-copy-id 192.168.56.102'. The terminal displays the source of the key, attempts to log in, and confirms that 1 key(s) remain to be installed. The user then enters the password for 'jmm197@192.168.56.102'. The terminal shows the login success, system information for Ubuntu 18.04.4 LTS, and system statistics.

```
m1 [Running]
The key's randomart image is:
+---[RSA 4096]-----+
|
| . S .
| ..*.. 0+
| 0 0*0.0.0= +
| .Bo@+.oE.* =
|+B==+. . 0.++0+
+---[SHA256]-----+
jmm197@m1:~$ ssh-copy-id 192.168.56.102
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jmm197/.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
jmm197@192.168.56.102's password:
Number of key(s) added: 1
Now try logging into the machine, with: "ssh '192.168.56.102'"
and check to make sure that only the key(s) you wanted were added.
jmm197@m1:~$ ssh 192.168.56.102
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-91-generic x86_64)

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

System information as of Fri Apr 3 16:58:48 UTC 2020

System load: 0.07          Processes: 92
Usage of /: 33.1% of 9.78GB Users logged in: 1
Memory usage: 44%         IP address for enp0s3: 10.0.2.15
```

FIGURA 5: Acceso a m2 sin contraseña

#### 4. PROGRAMAR TAREAS CON crontab

Edito el archivo /etc/crontab con la orden `crontab -e` y añado la línea

```
15 * * * * jmm197 rsync -avz -e ssh 192.168.56.102:/var/www/ /var/www/
```

A continuación utilizo `sudo service cron restart` para reiniciar el servicio. Finalmente, compruebo en el archivo /var/log/syslog que la tarea se ejecuta correctamente, como se puede comprobar en la siguiente imagen.

```
m1 [Running]
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
15 * * * * jmm197 rsync -avz -e ssh 192.168.56.102:/var/www/ /var/www/

"crontab.Y1xgyA/crontab" 24L, 960C escritos
crontab: installing new crontab
jmm197@m1:~$ sudo service cron restart
[sudo] password for jmm197:
jmm197@m1:~$ tail /var/log/syslog
Apr  3 17:08:47 m1 systemd[1]: Stopped Regular background program processing daemon.
Apr  3 17:08:47 m1 systemd[1]: Started Regular background program processing daemon.
Apr  3 17:08:47 m1 cron[3818]: (CRON) INFO (pidfile fd = 3)
Apr  3 17:08:47 m1 cron[3818]: (CRON) INFO (Skipping @reboot jobs -- not system startup)
Apr  3 17:09:08 m1 systemd-networkd[608]: enp0s8: DHCP: No routes received from DHCP server: No data available
Apr  3 17:09:08 m1 systemd-timesyncd[436]: Network configuration changed, trying to establish connection.
Apr  3 17:09:08 m1 systemd-timesyncd[436]: Synchronized to time server 91.189.94.4:123 (ntp.ubuntu.com).
Apr  3 17:15:01 m1 CRON[3841]: (jmm197) CMD (jmm197 rsync -avz -e ssh 192.168.56.102:/var/www/ /var/www/)
Apr  3 17:15:01 m1 CRON[3833]: (CRON) info (No MTA installed, discarding output)
Apr  3 17:17:01 m1 CRON[3845]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)
jmm197@m1:~$ _
```

FIGURA 6: Crontab y log de que cron funciona