

## Práctica 2

Primero instalamos rsync con *sudo apt-get install rsync* pero vemos que ya está instalada por defecto.

Escribimos la orden *sudo chown santiago:santiago -R html* dentro de */var/www* para poder cambiar el usuario de root a santiago, y esto lo hacemos en ambas máquinas

Después hacemos la orden:

```
rsync -avz -e ssh santiago@192.168.1.100:/var/www/html/ /var/www/html/
```

Nos queda :



```
santiago@UbuntuServer:/$ rsync -avz -e ssh santiago@192.168.1.100:/var/www/html
/var/www/html/
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be established.
ECDSA key fingerprint is bf:53:3e:c7:30:63:d3:af:1b:64:59:42:b7:8a:84:59.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.100' (ECDSA) to the list of known hosts.
santiago@192.168.1.100's password:
receiving incremental file list
html/
html/hola.html
html/index.html

sent 66 bytes  received 3,488 bytes  546.77 bytes/sec
total size is 11,558  speedup is 3.25
santiago@UbuntuServer:/$
```

Haciendo *ls-la /var/www/html* podemos ver que el directorio se ha copiado correctamente en la máquina 2.

Ahora vamos a configurar ssh para poder conectarnos automaticamente sin contraseña, para ello escribimos: *ssh-keygen -t dsa* y nos queda algo como lo siguiente:

```
santiago@192.168.1.100's password:
receiving incremental file list
html/
html/hola.html
html/index.html

sent 66 bytes received 3,488 bytes 546.77 bytes/sec
total size is 11,558 speedup is 3.25
santiago@UbuntuServer:/$ ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/santiago/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/santiago/.ssh/id_dsa.
Your public key has been saved in /home/santiago/.ssh/id_dsa.pub.
The key fingerprint is:
53:15:27:df:8a:25:56:ce:68:c7:97:3c:61:50:3f:03 santiago@UbuntuServer
The key's randomart image is:
+--[ DSA 1024]-----+
|                +E=+  |
|                . X+.+ |
|                . = B*+ |
|                . o = o+ |
|                S   .  |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
|                .      |
+-----+
santiago@UbuntuServer:/$
```

El campo enter passphrase lo dejamos en blanco en este caso. La consola nos devuelve donde se ha guardado la identificación y la clave pública.

Nos hemos desplazado a nuestra carpeta y hemos ejecutado el comando:

```
ssh-copy-id -i .ssh/id_dsa.pub santiago@192.168.1.100
```

Ya hemos configurado ssh para poder conectarnos a la otra máquina sin tener que autenticarnos cada vez.

Lo comprobamos haciendo `ssh santiago@192.168.1.100` y metiendo la contraseña del otro equipo y ya nos encontramos dentro de máquina 1.

Ahora vamos a utilizar el archivo crontab para programar que se active el rsync cada hora para mantener actualizado el directorio. Para ello ejecutamos:

```
sudo nano /etc/crontab.
```

Una vez dentro tenemos que añadir la línea:

```
0 * * * * santiago rsync -avz -e ssh santiago@192.168.1.100:/var/www/html/ /var/www/html/
```

De esta forma hacemos que se ejecute una vez al principio de cada hora. El archivo crontab queda de la siguiente manera:

```
# /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 --repo$
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
0 * * * * santiago rsync -avz -e ssh santiago@192.168.1.100:/var/www/html$
#
```

```
^G Ver ayuda  ^O Guardar   ^R Leer Fich ^Y RePág.    ^K Cortar Tex ^C Pos actual
^X Salir      ^J Justificar ^W Buscar    ^U Pág. Sig. ^U PegarTxt  ^T Ortografía
```