

INGENIERÍA DE SERVIDORES (2016-2017)
DOBLE GRADO EN INGENIERÍA INFORMÁTICA Y MATEMÁTICAS
UNIVERSIDAD DE GRANADA

Actividades Complementarias

Simón López Vico

11 de junio de 2017

Índice

1. Cuestión opcional 1: Muestre (con capturas de pantalla) cómo ha comprobado que el RAID1 funciona.	3
2. Cuestión opcional 2:	6
2.1. ¿Qué relación hay entre los atajos de teclado de emacs y los de la consola bash?	6
2.2. ¿Y entre los de vi y las páginas del manual?	6

Índice de figuras

1.1. Creación del archivo “prueba.txt” en el disco 1.	3
1.2. Desconexión del disco 1 de Ubuntu Server.	3
1.3. Captura de la máquina buscando el disco configurado en RAID.	4
1.4. Comandos realizados en la terminal.	4
1.5. Eliminación del RAID creado en el disco “md0”.	5
1.6. Comprobación de la existencia del archivo “prueba.txt” en el disco 2. . . .	5

Práctica 1.

1. Cuestión opcional 1: Muestre (con capturas de pantalla) cómo ha comprobado que el RAID1 funciona.

Para comenzar, arrancaremos Ubuntu Server desde el disco 1 y crearemos un archivo en el directorio home de nuestro usuario para más tarde comprobar si se ha creado en el disco 2. (1.1)

```
simlopvic@ubuntu-server:~$ pwd
/home/simlopvic
simlopvic@ubuntu-server:~$ ls
raid.txt
simlopvic@ubuntu-server:~$ cat raid.txt
*****
*****
Archivo de prueba.
*****
*****
simlopvic@ubuntu-server:~$
```

Figura 1.1: Creación del archivo “prueba.txt” en el disco 1.

A continuación, desde las opciones de nuestra máquina virtual en Virtual Box, desconectaremos el disco 1. (1.2)

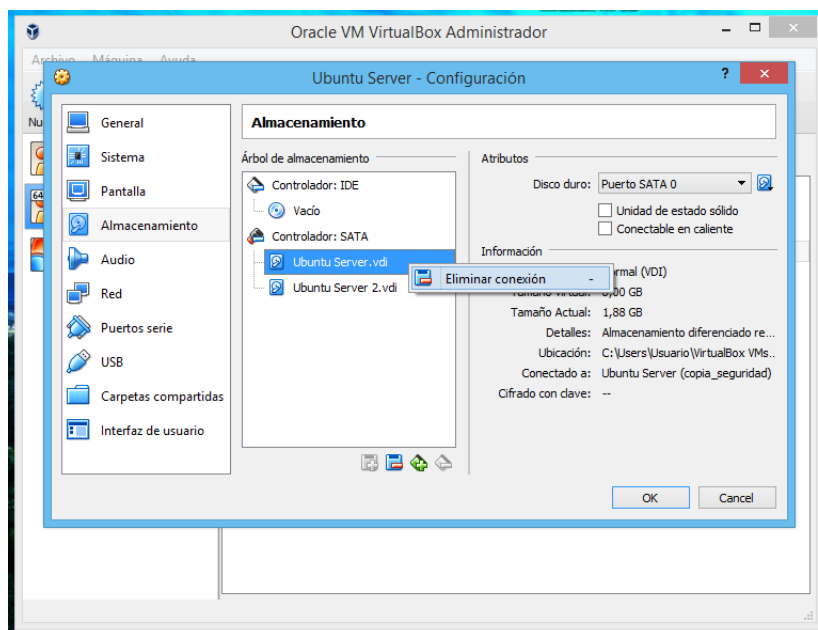


Figura 1.2: Desconexión del disco 1 de Ubuntu Server.

Al arrancar la máquina virtual, el sistema operativo detectará que hay un RAID1 configurado para dos discos duros, pero uno de ellos no estará conectado a la máquina, por lo que continuará buscando el disco duro hasta que pase un tiempo. (1.3)

Tras este tiempo, la máquina iniciará un proceso llamado “king pool” y abrirá una terminal.

```
Begin: Loading essential drivers ... [ 5.230112] md: linear personality registered for level -1
[ 5.232196] md: multipath personality registered for level -4
[ 5.234278] md: raid0 personality registered for level 0
[ 5.236910] md: raid1 personality registered for level 1
[ 5.308853] raid6: sse2x1 gen() 7646 MB/s
[ 5.376803] raid6: sse2x1 xor() 6077 MB/s
[ 5.445249] raid6: sse2x2 gen() 9984 MB/s
[ 5.513181] raid6: sse2x2 xor() 5339 MB/s
[ 5.581309] raid6: sse2x4 gen() 8021 MB/s
[ 5.649090] raid6: sse2x4 xor() 7722 MB/s
[ 5.649137] raid6: using algorithm sse2x2 gen() 9984 MB/s
[ 5.649178] raid6: .... xor() 5339 MB/s, rmw enabled
[ 5.649218] raid6: using ssse3x2 recovery algorithm
[ 5.650739] xor: automatically using best checksumming function:
[ 5.689168]   avx      : 18718.000 MB/sec
[ 5.690623]   async_tx: api initialized (async)
[ 5.700056] md: raid6 personality registered for level 6
[ 5.700118] md: raid5 personality registered for level 5
[ 5.700163] md: raid4 personality registered for level 4
[ 5.706112] md: raid10 personality registered for level 10
done.
Begin: Running /scripts/init-premount ... done.
Begin: Mounting root file system ... Begin: Running /scripts/local-top ... [ 5.720965] random: lvm urandom read with 35 bits of entropy available
   Reading all physical volumes.  This may take a while...
   No volume groups found
   No volume groups found
Begin: Waiting for encrypted source device... ...
```

Figura 1.3: Captura de la máquina buscando el disco configurado en RAID.

Iniciada la terminal, comprobaremos donde nos encontramos y qué archivos se encuentran en nuestro directorio. Si introducimos el comando “ls dev/sd*”, nos mostrará los discos conectados a la máquina, comprobando que solo disponemos de uno de ellos. Consultaremos el nombre de este disco imprimiendo el contenido del fichero mdstat incluido en el directorio proc, como vemos en la figura 1.4.

```
(initramfs) pwd
/
(initramfs) ls
dev      var      lib      lib64    conf     etc      sys      tmp
root     scripts  init     run      bin      sbin     proc
(initramfs) ls dev/sd*
dev/sda1 dev/sda
(initramfs) cat proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [raid10]
md0 : inactive sda1[1](S)
      8382464 blocks super 1.2
unused devices: <none>
(initramfs) _
```

Figura 1.4: Comandos realizados en la terminal.

Ahora que conocemos el nombre del disco duro conectado, eliminaremos el RAID que existe con el comando “mdadm” [1] para poder forzar su inicio. Con la opción -R, eliminaremos el RAID creado por HW para los dos discos para siempre, por lo que aunque reconectemos el disco 1, el RAID no funcionará como debe, haciendo que no se copien los archivos creados de un disco en el otro. (1.5)

```
(initramfs) ls dev/md0
dev/md0
(initramfs) mdadm -R dev/md0
mdadm: CREATE user root not found
mdadm: CREATE group disk not found
[ 1262.436064] md/raid1:md0: active with 1 out of 2 mirrors
[ 1262.436158] md0: detected capacity change from 0 to 8583577600
mdadm: started dev/md0
(initramfs) exit
Unlocking the disk /dev/disk/by-uuid/046dc3ea-c2a9-4bab-bb26-7b135775ee01 (HDS-raiz_crypt)
Enter passphrase: cryptsetup: HDS-raiz_crypt set up successfully
Unlocking the disk /dev/disk/by-uuid/d49eadca-b0b8-40f1-8fc4-c3a34da005ee (HDS-swap_crypt)
Enter passphrase:
```

Figura 1.5: Eliminación del RAID creado en el disco “md0”.

Para terminar, saldremos de la terminal, iniciándose el sistema operativo y pidiéndonos las contraseñas de cifrado de los volúmenes del disco, como podemos ver en la figura 1.5, e iniciando nuestra sesión.

Ya solo nos falta comprobar que el archivo creado en el disco 1 se ha copiado en el disco 2. En la figura 1.6 se muestra el directorio en el que nos encontramos (/home/simlopvic), así como el hecho de que solo se encuentra conectado un disco.

Finalmente, veremos que existe el archivo “prueba.txt” en el directorio “/home/simlopvic” del disco 2, así como lo habíamos creado en el disco 1 en la figura 1.1.

```
simlopvic@ubuntuuserver:(dom mar 19)(20:19:01)~$ pwd
/home/simlopvic
simlopvic@ubuntuuserver:(dom mar 19)(20:19:09)~$ ls ../../dev/sd*
../../dev/sda ../../dev/sda1
simlopvic@ubuntuuserver:(dom mar 19)(20:19:18)~$ ls
raid.txt
simlopvic@ubuntuuserver:(dom mar 19)(20:19:25)~$ cat raid.txt
*****
*****
Archivo de prueba.
*****
*****
simlopvic@ubuntuuserver:(dom mar 19)(20:19:28)~$ _
```

Figura 1.6: Comprobación de la existencia del archivo “prueba.txt” en el disco 2.

2. Cuestión opcional 2:

2.1. ¿Qué relación hay entre los atajos de teclado de emacs y los de la consola bash?

[2] Los atajos de teclado usados en una consola bash son posibles gracias a que por defecto está vinculado emacs como editor para escribir los comandos en la terminal. A continuación listaremos algunos de los comandos compatibles entre los dos softwares:

- **Ctrl+A** -> Desplaza el cursor al inicio de la línea.
- **Ctrl+E** -> Desplaza el cursor al final de la línea.
- **Ctrl+K** -> Borra desde el cursor hasta el final de la línea
- **Ctrl+T** -> Se intercambia el carácter que haya delante del cursor por el que haya detrás.
- **Esc+U** -> Convierte el texto en mayúscula desde el cursor hasta el final de la palabra.

2.2. ¿Y entre los de vi y las páginas del manual?

[3] Al pertenecer ambos al Proyecto GNU, tendrán una serie de atajos de teclado similares, como por ejemplo los descritos a continuación ¹ :

- **:q** -> Finalizará el editor o el manual.
- **ZZ** -> Igual que con :q.
- **Flecha arriba/abajo** -> Mueve el cursor hacia arriba/abajo.
- **h** -> Imprime en pantalla la ayuda del editor vi / del manual de linux.

¹Para este apartado he buscado los atajos de teclado que ofrece vi y he comprobado y comparado su funcionamiento en el manual de linux.

Referencias

- [1] MDADM: A NEW TOOL FOR LINUX SOFTWARE RAID MANAGEMENT, <http://www.linuxdevcenter.com/pub/a/linux/2002/12/05/RAID.html>, consultado el 19 de marzo de 2017.
- [2] TRUCOS DE LA CONSOLA BASH, <https://gnulinuxdocs.wordpress.com/2015/08/01/trucos-de-la-consola-bash/>, consultado el 20 de marzo de 2017.
- [3] GUÍA DE USO DE VI OR VIM, <https://www.garron.me/es/articulos/guia-de-vi-vim.html>, consultado el 20 de marzo de 2017.