

PRÁCTICA 1

CentOS pertenece a red-hat. Estable, bien documentado e ideal para usar en servidores.

La documentación de CentOS no está disponible a partir de la v.6, v.7, hay que consultar redhat

RAID: Redundant arrays of independent disk. Permiten que diversos discos duros se comporten como uno solo.

Hay distintos tipos de RAID. Por **software:** Divide los datos y escribe en cada disco duro. Esto supone una carga en la CPU:

- ➔ **RAID0:** Necesita al menos dos discos duros. Se divide los datos en dos partes y se escribe en cada disco duro, si se pierde la información en uno se pierde toda la info. Se gana eficiencia
- ➔ **RAID1:** Se duplica la información en el segundo disco para evitar la pérdida de información si uno de los disco falla. (mirror, necesitamos como mínimo 2 discos)
- ➔ **RAID5:** Necesitamos al menos tres discos duros, dividimos los datos en dos partes, guardamos cada mitad en cada uno de los dos primeros discos duros y con el tercero usamos bits de paridad para mantener la seguridad que nos proporciona RAID1.

Por **hardware:** Retira la carga de la CPU y se encarga directamente de esto el hardware, suponiendo un coste económico mayor. Se presenta al sistema como un único disco por conjunto de RAID.

LVM: (Logical Volume Manager) Administrador de volúmenes lógicos que permite crear particiones lógicas en Linux.

Tipos de virtualización: Por software: Un sistema operativo anfitrión (Host) da lugar al uso de diversos sistemas operativos mediante el uso de una máquina virtual.

Por **hardware:** Es la virtualización de ordenadores como plataformas hardware, abstrae el funcionamiento de una máquina para emularla.

Dockers, contenedores, es una tecnología reciente que permite la compartición de recursos entre máquinas virtuales y anfitrión.

MD: Es una utilidad (dispositivo) de Linux para gestionar y monitorizar los RAID.

FDE: Full Disk Encryption, hace ilegibles los datos del disco duro para aquellos que no tengan permisos. En concreto indica que todo en el disco salvo el gestor de arranque está cifrado.

Linux device-mapper: módulo que utiliza LVM para gestionar los vol. lógicos.

MBR: "Master Boot Record"

Bloques reservados: son los bloques que se reserva el kernel para que en el momento que haya un 100% de uso de disco, el sistema pueda seguir funcionando (por defecto 5%)

Para la toma de decisiones:

- Raid **hardware:** la controladora
- Raid **software:** el módulo software

Diferencias:

El hardware es más eficiente, es más seguro ya que es menos probable que haya un problema a nivel de bit, el hardware es más caro pero sin embargo en el hardware el fabricante nos puede dejar tirados, cosa que en el software no ocurre

SESIÓN 1 DE PRÁCTICAS:

INSTALACIÓN DE UBUNTU SERVER

Para instalar Ubuntu Server queremos un RAID1, con LVM y cifrado.

Tendremos tres particiones, para home, boot y swap.

LVM: Permite crear abstracciones y redimensionar fácilmente las particiones del disco duro. Permite unir distintos volúmenes físicos en grupos de volúmenes que permite un alto nivel de abstracción para gestionar el espacio en disco.

Consta de 4 capas, Almacenamiento real, volúmenes físicos, grupo de volúmenes y volúmenes lógicos.

Preparamos la máquina virtual:

- ➔ Creamos la máquina con todo por defecto, Linux 64 bits. RAM de 1024 MB.
- ➔ Creamos un disco duro virtual usando VDI como tipo.
- ➔ Tamaño dinámico
- ➔ Le otorgamos 10GB
- ➔ Añadimos la iso en “almacenamiento”
- ➔ Añadimos otro disco ya que al ser Raid1 necesita dos discos como mínimo

Iniciamos la máquina virtual y su instalación:

- ➔ La contraseña siempre será **practicass,ISE**
- ➔ Usaremos un método de particionado manual
- ➔ Aquí tendremos los dos discos duros (sda y sdb), Creamos tablas de particionado para cada disco
- ➔ Seleccionamos “configuración por software”, aceptamos los cambios en las tablas de particionado y creamos un dispositivo “MD” usando RAID1
- ➔ Indicaremos que usaremos 2 discos duros activos (2 y 0 usaremos), y seleccionamos (con espacio) los dos discos duros. Confirmamos los cambios.

Aquí se deberían apreciar los diversos discos creados, un solo RAID con el tamaño total.

No todos los sistemas de archivos permiten el resizing, pero un ext4 si que nos lo permite pudiendo aumentar o disminuir el tamaño de algún vol.lógico

No cifraremos la carpeta persona, lo haremos de forma más elaborada, FDE

- ➔ Procedemos a configurar los volúmenes lógicos (en la configuración LVM). Nombre del grupo de volúmenes: "Servidor", y ahí metemos nuestro RAID (/dev/md0)
- ➔ Creamos cuatro volúmenes lógicos. Dentro del grupo de volúmenes "Servidor" indicamos el nombre del volumen "Arranque", que recibirá 200MB. Posteriormente crearemos el volumen "Hogar", de 800MB. Tercer volumen "swap" que se le otorga el doble de la RAM disponible, 2048MB y el volumen "raíz" que se le otorga el resto.

Ahora bien, ciframos los volúmenes:

CIFRADO

Creamos una nueva encriptación, pero sólo cifraremos los dispositivos hogar, swap y raíz. (Swap se encripta por seguridad, pues se podría hacer un volcado de datos en la swap y se obtendrían los datos sin cifrar, donde puede haber información sensible expuesta en esta memoria).

No hace falta cifrar /boot ya que si lo ciframos, las versiones que traen por defecto no tienen la capacidad de activar un volumen lógico cifrado, luego si lo ciframos no arrancamos.

Utilizaremos la configuración por defecto. Le daremos a terminar y ponemos nuestra pass.

En este momento ya tenemos las particiones cifradas y procedemos a configurar los puntos de montaje.

PUNTOS DE MONTAJE

Seleccionamos cada partición para usarla con la extensión correspondiente, ext4 para todas salvo para swap, /boot se montará en esta primera partición de arranque.

Posteriormente configuraremos el cifrado para /home, con ext4.

Después el RAID cifrado, utilizamos ext4 y punto de montaje la raíz /

Por último swap, lo usaremos como área de intercambio.

Aquí finalizamos el proceso de particionado y escribimos los cambios en el disco duro. No necesitamos proxy, ni actualizaciones automáticas, ni programas adicionales.

Si nos pide instalar GRUB le decimos que si.

Con esto concluimos la instalación de Ubuntu Server. Con lsblk comprobamos que debajo de cada imagen pone crypt porque hemos encriptado cada volumen. Para apagar y que se guarde el historial usar power off o guardar el estado de la máquina

INSTALACIÓN DE CENTOS

En CentOS tendremos un único disco duro, con una partición física y otra mapeada.

Indicamos opciones por defecto para la instalación de la máquina, añadiendo la iso:

- ➔ Instalamos CentOS
- ➔ Después de elegir el idioma, en la siguiente pantalla seleccionamos un único disco.
- ➔ Creamos el Root, estableciendo la contraseña de siempre.

Podemos observar los disco con `lsblk` y obtener información con `mount | grep sda`

`df -h` (entendible para humanos) muestran los vol. lógicos, su punto de montaje y el espacio que ocupan

➔ `lvmdiskscan` muestra más información sobre los volúmenes lógico, así como `lvdisplay` muestra info. al respecto.

Ahora apagamos la máquina y y añadimos un nuevo disco por defecto.

Después de la instalación de **Ubuntu Server** lo configuramos para añadir una red:

Archivo ➔ Preferencias ➔ Red ➔ Redes solo anfitrión ➔ añadir

Vamos a Ubuntu ➔ adaptador 2 ➔ adaptador solo anfitrión ➔ iniciar Ubuntu

Nos dirigimos hacia `vi /etc/network/interfaces`, lo abrimos usando `vi`:

`i` ➔ cambiamos lo que queramos ➔ `esc`

Añadimos:

```
auto enp0s8
iface enp0s8 inet static
    address 192.168.56.105
```

➔ `ifup enp0s8` (súbeme la interfaz `enp0s8`)

-Ver `ifconfig` y comprobar que está, Hacemos un ping entre el anfitrión y la máquina para ver si lo hemos hecho bien:

➔ `ping -c 3 192.168.56.1`

Ahora hacemos una instantánea para guardar el estado de la máquina ➔ instantáneas ➔ tomar

Con la misma máquina apagada, modificamos y hacemos lo mismo

Editamos `vi /etc/sysconfig/network-scripts/ifcfg-enp0s3`

`ONBOOT=NO` ➔ `Yes` (para una interfaz de red)

Configuramos la otra interfaz:

→cd /etc/sysconfig/network-scripts

→cp ifcfg-enp0s3 ifcfg-enp0s8

→vi ifcfg-enp0s8 :

```
TYPE=Ethernet
BOOTPROTO=none
NAME=enp0s8
DEVICE=enp0s8
ONBOOT=yes
IPADDR=192.168.56.110
NETMASK=255.255.255.0
```

Podemos probar a hacer un ping de Ubuntu a centos con Ubuntu encendida

SESIÓN 2 DE PRÁCTICAS: Configurar CENTOS con LVM

Modificaremos la configuración por defecto para cambiar un punto de montaje a otro volumen lógico y redimensionarlo

/var necesita:

- +espacio

- +prestaciones. En /var se almacenan los correos electrónicos, Bd, loss, copias temporales. Si queremos agrandar el espacio ya que se nos queda pequeño debemos asignar a otro volumen lógico (+espacio)

Cada volumen lógico puede tener su propio sistema de archivos (nivel más alto de abstracción)

Centos → lsblk (Tenemos que saber para qué sirve cada columna de este comando).

Podemos verlo usando lsblk -h . Actualmente tenemos 2 particiones (sda1 y sda2)

Automáticamente CentOS configura la raíz y swap como volúmenes lógicos

→mount | grep sda

El sistema de archivos es xfs (por defecto en CentOS) ya que tiene mejor rendimiento para archivos.

→cat /etc/fstab (fstab = file system table)

Monitorización

→df -h (nos da información sobre archivos, punto de montaje..etc, la opción -h es para que sea entendible a los humanos)

→lvmdiskscan (nos da información de los LVM – (cómo formatea los discos))

→lvdisplay (logical volumes)

→pvdisplay (physical volumes)

→power off

Configuración → añadimos disco VDI por defecto en centos → iniciamos de nuevo

→pvcreate /dev/sdb (añadimos volumen físico disp. Sobre el que creamos la abstracción)

→pvdisplay (información sobre los physical volumen, lo comprobamos)

Añadimos al grupo de volúmenes (lo extendemos)

➔ vgdisplay (información)

➔ vgextend cl /dev/sdb (expande al max disponible, como hemos añadido otro disco pasa de 8GB a 14,99GB o algo parecido)

➔ Comprobamos con vgdisplay que ha aumentado

Creamos nuevo volumen lógico:

lvcreate -L 4G -n newvar cl (-L = large, 4G = 4gb, -n=nombre, cl=grupo vol. donde lo ubicamos)

→lvdisplay // lvs

Dejamos opciones por defecto para ext4

Hacemos un sistema de archivos, lvresize nos da fallo con xfs, por eso usamos ext4

mkfs -t ext4 /dev/cl/newvar (mkfs=make filesystem)

Para evitar la fragmentación → tamaño de bloque pequeño

Montando el sistema de archivos:

1) mkdir /media/newvar

2) systemctl isolate runlevel1.target

a) Hay una serie de “Niveles de ejecución (run levels)” 6 en total, Lo pasamos al nivel 1 ya que es el modo mantenimiento (root) (al acabar todo lo pasamos al nivel 5, no lo pasamos al nivel 6 ya que ese es el de reinicio)

b) SE Linux = Security Enhanced Linux

- Establece normas para no cambiar los puertos de los servicios más importantes. Si por ejemplo html trabaja en el puerto 80 y queremos cambiarlo al puerto 82 no nos deja.

-Establece normas para los archivos y sus contextos (:object_r)

`cp -a === cp - -preserve=context`

(ls /var -Z) (contextos de los archivos por defecto de /var)

3) `mount /dev/mapper/cl-newvar /media/newvar`

4) `mount` (para comprobar)

5) `cp -a /var/. /media/newvar` (el /var/. Copia archivos ocultos y visibles)

➔ `systemctl status` (nos muestra el status/nivel en el que nos encontramos)

6) `ls /media/newvar`

7) Añadimos al /etc/fstab la línea:

`/dev/mapper/cl-newvar /var ext4 defaults 0 0`

(el 0 0 establece la prioridad de las copias de seguridad)

8) `mv /var /var_old` (liberar espacio de var, si no lo hago no puedo liberar la memoria de /var porque está inaccesible)

`mkdir /var` (ya que lo hemos movido arriba)

`mount -a` (Nos monta /var en nuestro volumen lógico)

`mount` (comprueba) (`mount | grep var`)

`df -h` (vemos tamaño)

9) `umount /media/newvar`

SESIÓN 3 DE PRÁCTICAS:

¿Por qué queremos /var en un volumen lógico? Por espacio y por seguridad.

HACER EN CENTOS

Añadimos dos disco a Centos justo después de instalarlo (podemos rescatarla desde una instantánea)

➔ Almacenamiento ➔ Creamos 2 nuevos disco aparte de la iso

Iniciamos la máquina Centos y con lsblk comprobamos que se han creado bien. Ahora instalamos el raid con el comando:

➔ yum install -y mdadm (multiple device administrator)

➔ ip addr (no tenemos asignada una dirección IP)

➔ ifup enp0s3 (subimos la tarjeta de red)

➔ ip addr (ya la tenemos asignada)

➔ ping www.google.es

➔ yum install -y mdadm (-y para no tener que darle a yes)

Ahora creamos el raid:

➔ mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sdb /dev/sdc
(es un raid1) (tenemos 2 discos, los indicamos)

➔ lsblk

sdb__md0 raid1

sdc__md0 raid1 (En esta salida sdb y sdc son dispositivos lógicos y md0 es abstracción lógica)

Queda crear volumen físico y grupo de volúmenes:

➔ pvcreate /dev/md0 (physical volumen créate en /dev/md0)

➔ pvs (lo comprobamos)

➔ pvdisplay (lo comprobamos)

Creamos un grupo de volúmenes en el que cuelga /dev/md0 y de este cuelgan sdb y sda, así tenemos la certeza que se mapea al raid1

➔ vgcreate pmraid1 /dev/md0 (grupo de vol. pmraid1 creado)

➔ vgs (lo comprobamos)

Creamos el volumen lógico donde montamos /var)

➔ lvcreate -L 1G -n newvar pmraid1

➔ lvdisplay (lo comprobamos)

Cambiamos el nivel de ejecución al root level 1

→systemctl isolate runlevel.target

→system status (comprobarlo)

Creamos el Sistema de Archivos, por defecto xfs en Centos:

→mkfs -f xfs /dev/mapper/pmraid1-newvar

Creamos un directorio para copiar los archivos

→mkdir /media/newvar

→mount /dev/mapper/pmraid1-newvar (montamos el vol. lógico en el directorio creado)

→mount | grep var (para ver si está OK)

→cp -a /var/. /media/newvar/

→mkdir /var_old/

→mv /var/. /var_old/ ← (Error)!!!

→mv /var /var_old

→mkdir /var

→restorecon /var //restaura el contexto de /var

1º)Subimos la interfaz de red ya que nos hemos cambiado de nivel de ejecución

→ifup enp0s3

→yum install lsof

→lsof /var (Nos muestra los archivos abiertos)

→vi /etc/fstab

Añadimos en /etc/fstab:

/dev/mapper/pmraid1-newvar /var /xfs defaults 0 0

→mount -a

→lsblk:

pmraid1-newvar /var

pmraid1-newvar /var (var está en raid1)

→mount | grep /var → vemos que no lo hemos desmontado

→umount /media/newvar

Hacemos Escenario 3, Ciframos /var

Hay dos formas de cifrado:

-LVM on LUKS: formateamos sobre LUKS, es más sencillo ya que hay una única clave para todo

-LUKS on LVM: Nos aporta más flexibilidad y seguridad ya que podemos tener una clave por volumen lógico, es decir, podemos cifrar los volúmenes lógicos que queramos. Usaremos esta segunda.

→yum -y install cryptsetup

Montamos /var colgado de / (Suponemos que hay suficiente espacio en cl-root)

→systemctl status (debemos estar en el nivel 1, root)

→mkdir /varRAID

→cp -a /var/. /varRAID (por seguridad)

→vi /etc/fstab (borramos la línea que hemos añadido antes)

→mount -a

→lsblk (vemos que /var no está montado en /)

→umount /dev/mapper/pmraid1-newvar

→lsblk (/var ya cuelga de / , ya que no aparece en MOUNTPOINT)

→cp -a /varRAID/. /var

→ls -Z /var

Ciframos el volumen lógico /newvar (LUKS on LVM). Antes del luksFormat metemos podemos meter basura con el comando "shred" para más seguridad.

→cryptsetup luksFormat /dev/mapper/pmraid1-newvar

→lsblk (no hemos apreciado ningún cambio)

→mount /dev/mapper/pmraid1-newvar /media/newvar

(No nos deja montar ya que está cifrado), Debemos activar un volumen lógico antes de usarlo.

→cryptsetup luksOpen /dev/mapper/pmraid1-newvar pmraid1-newvar_crypt

→lsblk (vemos que ____crypt cuelga de newvar)

Ya podemos montar el volumen lógico.

→mount /dev/mapper/pmraid1_newvar_crypt /media/newvar (ERROR), tenemos que crear primero el sistema de archivos SA.

→mkfs -t xfs /dev/mapper/pmraid1_newvar_crypt

Lo montamos con el comando anterior y se comprueba con mount

Cogemos el /var y lo metemos...

→cp -a /var. /media/newvar

→vi /etc/fstab Añadimos:

/dev/mapper/pmraid1_newvar_crypt /var /xfs defaults 0 0

(Los 00 indican que no queremos prioridad para las copias de seguridad)

→mv /var /var_old2

→mkdir /var

→restorecon /var

→mount -a

→lsblk (/var dentro de raid1 y el volumen lógico está cifrado)

→mount /media/newvar (comprobamos, aún está)

→umount /media/newvar/

→mount /media/newvar (comprobamos, ya está desmontado)

En el archivo /etc/crypttab

Nombredestinounavezabierto UUID=..... none

(el none indica que no busca ningún archivo buscando la clave para que en el arranque te la pida a ti)

→blkid

→blkid | grep newvar (Obtener el UUID universal unique ID)

→less /etc/crypttab (vemos que está vacío)

→blkid | grep crypto >> /etc/crypttab

→vi /etc/crypttab

Dejamos: pmraid1_newvar_crypt UUID=.... none

→blkid | grep crypto == cat /etc/crypttab

→cat /etc/fstab 1 | grep newvar

→reboot

HISTIMEFORMAT="%d %m %y %T"

History > historial.txt

Less historial.txt