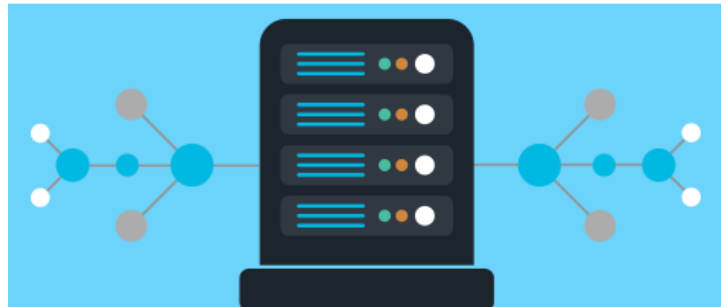


ADMINISTRACIÓN DE SISTEMAS OPERATIVOS Y REDES DE COMPUTADORES

Servicios avanzados sobre servidores.



ÍNDICE:

1.	Explicación de la práctica:	3
2.	Servicios:	4
2.1.	CentOS:	4
	• Servidor Web:	4
	• Servidor BD:	4
	• Backup:	6
	• Proxy:	6
	• Nagios:	7
	• Mensajería Instantánea:	8
	• Servidor de trabajo en grupo:	8
	• RAID:	8
2.2.	FreeBSD:	9
	• Servidor Web:	9
	• Servidor BD:	9
	• Backup:	10
	• Proxy:	10
	• Nagios:	11
	• Mensajería Instantánea:	12
2.3.	Windows Server 2016:	13
	• Servidor Web:	13
	• Servidor BD:	13
	• Backup:	13
	• Proxy:	14
	• CloudRadar:	15
	• Mensajería Instantánea:	15
	• RAID:	15

1. Explicación de la práctica:

En esta tercera práctica se pretende instalar servicios avanzados sobre servidores (Windows Server, Centos/Debian, FreeBSD/Solaris).

Los sistemas operativos que voy a utilizar son:

- Windows Server 2016
- CentOS
- FreeBSD

Y los servicios que voy a instalar en los diferentes sistemas operativos son los siguientes:

- Mensajería instantánea (Jabber)
- RAID
- Servidor Web (dominios Virtuales)
- Backup
- Servidor de trabajo en grupo
- Servidor de BD
- Proxy Cache (Squid)
- Monitorización de servicios (Nagios)

Debemos instalar todos los servicios nombrados, crear las particiones necesarias para algún servicio, además de añadir los repositorios necesarios para instalar algunos servicios.

Además, para mayor comodidad se desactivará el firewall en los diferentes Sistemas Operativos y SELinux en los sistemas Linux.

2. Servicios:

Instalé xfce4 para mayor comodidad a la hora de instalar los servicios en CentOS y FreeBSD.

A continuación, explico como he instalado los diferentes servicios pedidos en el Hito 3:

2.1. CentOS:

- **Servidor Web:**

Para instalar el servidor web apache, hacemos:

```
yum install httpd
```

Una vez instalado, podemos iniciar el servicio de apache con:

```
service httpd start
```

También instalaremos php para que los sitios web puedan acceder a una base de datos y dar una funcionalidad real a la web. Para ello usamos:

```
Yum install php php-mysql php-pgsql
```

Además, vamos a iniciar apache con el sistema usando:

```
chkconfig httpd on
```

Ahora, vamos a crear los directorios que alojarán las webs. Para ello, nos movemos hasta /var/www y creamos un directorio para cada sitio.

Creamos los diferentes dominios virtuales gracias a este video:

<https://www.youtube.com/watch?v=SSeYQpK9hOQ>

Nos descargamos los archivos de Joomla y de Wordpress y cada uno de ellos lo ponemos en /var/www/.

Después de haber puesto los diferentes archivos en esa ruta nos conectamos a los dominios configurados y empezamos a instalar, primero Wordpress y luego Joomla. (Un dominio para Wordpress y otro para Joomla)

Cuando terminemos de instalar Wordpress y Joomla, tras poner el dominio y luego index.php nos aparecerá una página de ejemplo.

- **Servidor BD:**

Instalaremos PostgreSQL para utilizar como nuestro motor de base de datos con el comando `"yum install -y postgresql postgresql-server postgresql-devel postgresql-libs"`.

Después de completar la instalación nos aseguraremos que PostgreSQL inicie automáticamente después de los reinicios con el comando `"systemctl enable postgresql.service"`.

Luego iniciamos la estructura de directorios y base de datos con `"postgresqlsetup initdb"` e iniciamos el servicio con `"systemctl start postgresql.service"`.

Una vez instalado PostgreSQL, necesitamos crear la base de datos, usuario y contraseña. Para eso tenemos que cambiar de usuario con `"su postgres"`.

Creamos la base de datos con nombre `"asorc"`. Probablemente aparecerán algunas advertencias similares a `"could not change directory to /root"` los cuales ignoraremos:

```
createdb asorc
```

Una vez creada la base de datos, creamos el usuario `"userasorc"`:

```
createuser -P userasorc
```

Ahora es necesario setear el usuario administrador de PostgreSQL:

```
psql -U postgres -d postgres -c "ALTER USER postgres WITH PASSWORD  
'CONTRASEÑA-INGRESADA-EN-EL-PASO-ANTERIOR' ;"
```

Las contraseñas están seteadas, ahora configuraremos la base de datos de modo tal que para cada conexión se requiera el ingreso de la contraseña

```
nano /var/lib/pgsql/data/pg_hba.conf
```

Ir al final del archivo y modificar los datos y modificar `"ident"` y `"peer"` por `"md5"`, luego guardar el archivo.

```
# TYPE  DATABASE  USER      CIDR-ADDRESS  METHOD
# "local" is for Unix domain socket connections only
local   all             all                                md5
# IPv4 local connections:
host    all             all       127.0.0.1/32   md5
# IPv6 local connections:
host    all             all       ::1/128        md5
```

Salir de la línea de comando de PostgreSQL con `"exit"` y reiniciamos la base de datos con `"systemctl restart postgresql.service"`.

Ahora la base de datos ya está creada y podremos acceder a ella mediante HTTP.

Finalmente ejecutamos el comando `"setsebool httpd_can_network_connect_db on"` para permitir a httpd conectar con las bases de datos.

Para instalar PHP y poder utilizar los paquetes necesarios para utilizar PostgreSQL instalaremos el paquete necesario con `"yum -y install phpPgAdmin php-pgsql"`.

- Backup:

Antes de empezar, creamos la siguiente estructura para hacer la prueba:

```
Mkdir backups
Ls Documentos
Nano Documentos/datos
Mkdir Documentos/recibos
Nano Documentos/recibos/enero
```

En el directorio backups es donde almacenaremos el contenido de los backups que hagamos del directorio Documentos.

Ahora utilizamos el comando rsync de la siguiente forma para hacer el backup absoluto:

```
rsync -av Documentos/ backups
```

Donde Documentos/ es el directorio del cual queremos hacer el backup absoluto, y backups la ruta de destino.

Como podemos observar, se ha hecho el backup completo de Documentos en la carpeta destino. Tras un par de modificaciones en el archivo datos, hacemos un backup incremental con rsync así:

```
rsync -avvb -backup-dir=$PWD/backup_$(date +%d%m%y% H%M) Documentos/
backups
```

Y comprobamos la salida, como se puede observar, se ha hecho una copia del archivo datos que actualmente está en la carpeta backups en la carpeta backup_1112141913, y posteriormente se ha copiado el nuevo archivo datos a la carpeta backups.

- Proxy:

Para proxy se deberá descargar el paquete de Squid, para instalarlo y configurarlo, simplemente ponemos:

```
Yum install squid
```

Una vez hecho esto, vamos al archivo de configuración que se encuentra en /etc/squid/squid.conf y añadimos nuestras ips

Después, creamos un archivo con las expresiones regulares de los sitios a los cuales queremos denegar el acceso en /etc/squid/listas/expreg-denegadas.

```
Mkdir /etc/squid/listas
Nano /etc/squid/listas/expreg-denegadas
```

Y definimos las palabras que queremos filtrar.

ahora, volvemos al archivo /etc/squid/squid.conf y modificamos la directiva http_acceso allow de la siguiente forma para que las webs de la acl expreg-denegadas no se sirvan:

```
http_access allow localnet localhost !expreg-denegadas
```

Una vez finalizado, reiniciamos el servicio con:

```
Service squid reload
```

Y lo añadimos al arranque del sistema con:

```
Chkconfig squid on
```

Una vez hecho esto configuramos nuestro cliente (en este caso Firefox) para que use el proxy:

```
10.0.2.18
```

Y una vez hecho esto, pasamos a comprobar que nos filtra correctamente los resultados que hemos puesto en el archivo.

- **Nagios:**

Nagios no viene en los repositorios oficiales de CentOS7, por lo que si todavía no los habíamos instalado, hay que hacerlo mediante los siguientes pasos:

```
rpm -Uvh http://download.fedoraproject.org/pub/epel/7/x86\_64/e/epel-release-7-2.noarch.rpm
```

Y una vez instalado, hacemos yum update para actualizar los headers del repositorio. Hecho esto, ya podemos instalar nagios escribiendo:

```
yum install nagios*
```

Una vez instalado, vamos a configurarlo. Lo primero que vamos a hacer es especificar una cuenta de correo para que nos lleguen correos de alerta. Para ello editamos el archivo /etc/nagios/objects/contacts.cfg

Además, como podemos ver estoy define un usuario de nombre nagiosadmin. Para poder acceder a nagios debemos darle una contraseña a este usuario mediante htpasswd /etc/nagios/passwd nagiosadmin.

Una vez hecho esto, procedemos a iniciar nagios, reiniciar apache para que nos muestre la interfaz de nagios, y a iniciar nagios con el sistema.

```
Start nagios
```

```
Restart httpd
```

```
Chkconfig nagios on
```

Y una vez hecho esto, ya podremos comprobar en el navegador el funcionamiento del mismo en la dirección localhost/nagios/.

- **Mensajería Instantánea:**

Instalamos e iniciamos openfire.

Realizamos la configuración básica a través de la web.

Establecemos la cuenta de administrador y creamos usuarios.

Conectamos al servidor mediante el cliente Spark o Tkabber y vemos la conversación entre dos clientes.

- **Servidor de trabajo en grupo:**

Para el servidor de trabajo en grupo en CentOS he instalado Zimbra.

Para descargarlo, instalarlo y configurarlo he seguido los siguientes pasos:

https://www.server-world.info/en/note?os=CentOS_7&p=zimbra&f=1

Para ver como iniciar Zimbra en el navegador he seguido estos pasos:

https://www.server-world.info/en/note?os=CentOS_7&p=zimbra&f=2

- **RAID:**

Para instalar RAID, ya sea de tipo 1 o 5, que son los solicitados en esta práctica utilizaremos, en CentOS, el programa “**mdadm**”, el cual nos permite crear las particiones de manera automática indicando únicamente el nivel con el que se desea realizar la configuración.

Es por esto que para la configuración en CentOS utilizamos la orden “**mdadm --create /dev/nombreRaid --level=X --raid-devices=Y**”, donde “X” es el nivel del raid a crear e “Y” el número de discos añadidos al principio al raid.

Más adelante se pueden añadir nuevos discos con “**mdadm -a /dev/nombreRaid /dev/newDisk**” o quitarlos cambiando la opción “-a” por “-r”. De igual modo, se pueden marcar como faltantes los discos mediante la opción “-f”.

Esta opción marca el disco utilizado como faltante, es decir, lo extrae del raid sin quitarlo de su configuración, de modo que podamos insertar uno nuevo que no lleve ficha marca, mientras que “-r” elimina el disco de la configuración del raid, lo que hace que, en el caso de que se usarán 3 discos, ahora sólo se usen 2 y no se necesite de un tercero.

2.2. FreeBSD:

- **Servidor Web:**

Para instalar el servidor web apache ponemos:

```
Pkg install apache42
```

Vamos a etc/rc.conf y ponemos:

```
apache42_enable="YES"
```

Iniciamos apache con **apachectl start**.

Ahora vamos al directorio /usr/local/etc/apache/extra y editamos el archivo httpdvhosts.conf, que es donde se añaden los hosts virtuales.

Eliminamos los dos ejemplos de VirtualHost que están en el archivo.

En primer lugar, añadiremos localhost, el que vemos en primer lugar, y en segundo lugar añadimos el virtualhost que elijamos, como vemos tenemos que poner en DocumentRoot la ruta donde estarán los archivos del servidor, el ServerName que es el nombre del servidor, el ServerAlias que es lo mismo con www y la parte Directory donde indicamos la ruta donde están los ficheros del servidor y dentro indicamos Allow from all para que se pueda acceder al dominio.

Vamos al directorio /usr/local/etc/apache24 y editamos el archivo de configuración httpd.conf.

Descomentamos la línea de Virtual hosts para que se utilicen los virtual hosts que hemos definido.

Después, vamos a editar el archivo hosts que está situado en el directorio etc.

Añadimos las líneas necesarias para los dominios virtuales.

Creamos las carpetas de Wordpress y Joomla en el directorio que hemos indicado en el archivo de antes y le damos permisos a las dos carpetas que hemos creado.

Añadimos los archivos descargados en las carpetas nuevas creadas y nos conectamos a los dominios configurados y empezamos a instalar, primero Wordpress y luego Joomla. (Un dominio para Wordpress y otro para Joomla)

Cuando terminemos de instalar Wordpress y Joomla, tras poner el dominio y luego index.php nos aparecerá una página de ejemplo.

- **Servidor BD:**

Para instalar el servidor mysql escribimos el siguiente comando:

```
pkg install mysql56-client mysql56-server.
```

Antes de trabajar con mysql deberemos ejecutar el mysql_upgrade. Ahora para activar el servicio y el autoarranque de mysql vamos a /etc/rc.conf y añadimos la siguiente línea:

```
Mysql_enable="YES"
```

Después vamos arrancamos el servidor haciendo `/usr/local/etc/rc.d/mysql-server start` y ejecutamos el `mysql_upgrade` que se nos había especificado al acabar la instalación.

Una vez acabado ya podremos acceder a nuestro servidor mediante el cliente `mysql`.

Ahora vamos a crear las tablas y los registros de ejemplo dentro de `mysql`.

Creamos el script `php` para mostrar los datos vía web y es necesario instalar `mysqli`.

Tras la instalación reiniciamos `apache` y podremos comprobar en el navegador ya nos aparecen los datos de la bd.

- **Backup:**

Para empezar, instalamos `rsync`:

```
Pkg install rsync
```

Después creamos los directorios y archivos de prueba:

```
mkdir backups
```

```
mkdir Documentos
```

```
nano Documentos/datos
```

```
mkdir Documentos/recibos
```

```
nano Documentos/recibos/enero
```

Ahora, para crear el primer backup absoluto hacemos:

```
rsync -av Documentos/ backups
```

Como podemos comprobar `rsync` ha copiado todo, ya que hemos hecho un backup absoluto. Ahora vamos con el backup incremental. Para ello, modificamos el archivo `datos` y ejecutamos lo siguiente:

```
rsync -avvb -backup-dir=$PWD/backup_`date +%d%m%h%H%M` Documentos/  
backup
```

Como podemos ver sólo se ha copiado `datos` ya que `recibos/enero` estaba actualizado.

- **Proxy:**

Para instalar `squid` hacemos:

```
pkg install squid
```

Como podemos ver, antes de iniciar `squid` debemos hacer `squid -z` para crear los directorios de caché, pero antes de eso vamos a modificar el archivo `squid.conf` que se encuentra en `/usr/local/etc/squid/squid.conf` y le vamos a añadir la línea `visible_hostname` y a descomentar `cache_dir`:

```
Visible_hostname freebsd-cjg10.bsd
```

Una vez hecho esto, ya sí que ejecutamos `squid -z` y veremos cómo crea todos los directorios.

Una vez hecho esto, creamos el siguiente archivo un archivo en `/usr/local/etc/squid/deny.txt` que es donde especificaremos las palabras prohibidas, las cuales denegará el proxy.

Y ahora vamos a `/usr/local/etc/squid/squid.conf` para especificar en las acl que queremos denegar el acceso usando ese archivo:

```
Acl expreg-denegadas url_regex "/usr/local/etc/squid/deny.txt"
```

Y ahora denegamos el acceso en la red con:

```
http_access allow localnet !expreg-denegadas
```

Una vez hecho esto, ya solamente nos queda modificar el archivo `/etc/rc.conf` y añadirle la siguiente línea:

```
Squid_enable="YES"
```

Ahora reiniciamos el servicio con `service squid restart`, configuramos el navegador como en los otros S.O. y podremos ver que el proxy nos deniega el acceso a las paginas prohibidas.

- **Nagios:**

Instalamos Nagios con:

```
pkg install nagios
```

Vamos a empezar por el principio. Vamos a `/etc/rc.conf` y añadimos la línea:

```
nagios_enable
```

Después, vamos a `/usr/local/etc/nagios` y copiamos todos los archivos sample a archivos `cfg`

Y ahora vamos a modificar `httpd.conf` con la configuración especificada tras la instalación.

Ahora reiniciamos apache para que cargue la configuración con `apachectl restart` y probamos a entrar a nagios con el navegador desde `localhost/nagios/`

Como podemos observar, nos deniega el acceso. Para solucionar esto, vamos a realizar lo siguiente.

Y después volvemos a configurar apache:

```
<Directory /usr/local/www/nagios>
```

```
Options None
```

```
AllowOverride None
```

```
Allow from all
```

```
AuthName "Acceso de Nagios"
```

```
AuthType Basic
```

```
AuthUserFile /usr/local/etc/nagios/httpasswd.users
```

```
Require valid-user
```

```
</Directory>
```

Una vez guardada la nueva configuración, reiniciamos apache y volvemos a probar. Tras la autenticación, comprobamos que podemos acceder al panel de Nagios correctamente.

- **Mensajería Instantánea:**

Instalamos e iniciamos openfire.

Realizamos la configuración básica a través de la web.

Establecemos la cuenta de administrador y creamos usuarios.

Conectamos al servidor mediante el cliente Spark o Tkabber y vemos la conversación entre dos clientes.

2.3. Windows Server 2016

- **Servidor Web:**

Descargamos, instalamos xampp e iniciamos apache en el panel de control.

Vamos a la ruta de hosts y añadimos los dominios virtuales.

Creamos las carpetas para los dominios.

Ahora editamos el archivo httpdvhhosts.conf, que es donde se añaden los hosts virtuales.

Eliminamos los dos ejemplos de VirtualHost que están en el archivo.

En primer lugar añadiremos localhost, el que vemos en primer lugar, y en segundo lugar añadimos el virtualhost que elijamos, como vemos tenemos que poner en DocumentRoot la ruta donde estarán los archivos del servidor, el ServerName que es el nombre del servidor, el ServerAlias que es lo mismo con www y la parte Directory donde indicamos la ruta donde están los ficheros del servidor y dentro indicamos Allow from all para que se pueda acceder al dominio.

Ahora vamos a httpd.conf y descomentamos la línea de Virtual hosts para que se utilicen los virtual hosts que hemos definido.

Añadimos los archivos descargados (Wordpress y Joomla) en las carpetas nuevas creadas y nos conectamos a los dominios configurados y empezamos a instalar, primero Wordpress y luego Joomla. (Un dominio para Wordpress y otro para Joomla)

Cuando terminemos de instalar Wordpress y Joomla, tras poner el dominio y luego index.php nos aparecerá una página de ejemplo.

- **Servidor BD:**

Xampp viene con mariadb, lo único que hay que hacer es meterse a la página web de administrador de Apache, generar una base de datos pequeña con un usuario y permisos y acceder a ella mediante la web y ya estaría.

- **Backup:**

Lo primero que deberemos hacer es descargar CobianBackup desde su página web (descargaremos la versión más reciente, la 11 en este caso) y procederemos con la instalación:

Antes de nada, seleccionaremos "ESPAÑOL" en la lista de la izquierda, y pulsaremos "Próximo".

Ahora, marchamos el checkbox "Yo acepto las condiciones" y pulsamos de nuevo en "Próximo".

Ahora, nos permitirá cambiar la dirección de instalación y ciertos parámetros de ésta. Como en nuestro caso no queremos cambiar nada, simplemente volveremos a pulsar en "Próximo".

Ahora, seleccionaremos en "Tipo de instalación" la opción de "Aplicación (sin auto-inicio)" y pulsaremos en "Próximo".

Pulsamos en "Instalar" y, una vez acabe el proceso, ya habremos acabado con la instalación.

Ahora, iniciaremos Cobian Backup 11 y procederemos a ir a "Tarea / Nueva tarea"

Se nos abrirá una nueva ventana, donde escribiremos el nombre que le queremos dar a la Tarea (en este caso "ASORC Completa") y dejamos el resto de las opciones por defecto.

Ahora, en el menú izquierdo, pulsamos en "Ficheros", y, en "Fuente" seleccionaremos la o las carpetas (o directamente todo el disco duro) del que queremos hacer la copia, mientras que en "Destino" seleccionaremos la carpeta donde se guardará dicho backup.

Pincharemos en "Horario" (en el menú izquierdo), lo que nos mostrará un menú para configurar que la copia de seguridad se lance automáticamente cada X tiempo. En nuestro caso seleccionaremos "Diaria" en "Tipo de horario" y la configuraremos para que se lance a las 21:00. Además, no está de más saber que, si pulsamos sobre "Archivo" en el menú izquierdo, podríamos configurar que se comprima la copia de seguridad, que se divida en partes de cierto tamaño (para, por ejemplo, guardarla en DVDs) o que se cifre.

Bien, ya tenemos configurada la copia, por lo que vamos a iniciarla de manera manual. La guardaremos pulsando "OK" y, en el menú principal de Cobian, tras seleccionarla, daremos click en el icono que simboliza "Ejecutar las tareas seleccionadas ahora".

Nos mostrará un popup preguntando si deseamos ejecutar las tareas seleccionadas y permitiéndonos apagar el ordenador al finalizar, en nuestro caso simplemente pulsaremos en "OK".

Una vez acabado el proceso, comprobamos que se la copia se ha realizado correctamente.

Por último, vamos a crear una copia diferencial, por lo que crearemos una tarea exactamente de la misma manera que la anterior, a excepción de las opciones del menú principal de "Nueva Tarea", donde la llamaremos "ASORC Incremental" y en "Tipo de respaldo" seleccionaremos "Incremental".

Una vez guardada la tarea procedemos a iniciar la copia y, cuando haya acabado, creamos un archivo nuevo en el origen y volvemos a lanzar de nuevo la tarea. Hecho y finalizado esto, podremos comprobar cómo se ha creado una copia incremental con los nuevos archivos.

- **Proxy:**

Descargamos squid desde su web oficial <http://www.squid-cache.org>. Usaremos la versión 2.7, y una vez descargado descomprimos el archivo en C:

Ahora vamos a C:\squid\etc, hacemos una copia de los archivos default (para tenerlos como respaldo o tomarlos de referencia en futuras configuraciones) y a las copias le quitamos la extensión default para que queden como *.conf.

Ahora, con la consola vamos a la ruta C:\squid\sbin. Usamos squid -i para que se inicie como servicio de Windows y después squid -z para que cree los archivos necesarios.

Ahora vamos a la ventana de servicios y activamos squid.

Ahora vamos a usar un archivo para restringir el acceso de las páginas que queremos bloquear. Para ello creamos el archivo deny.txt en la ruta C:\squid\etc\.

En el deberemos poner las palabras que queremos filtrar, de este modo, cuando se intente acceder a un dominio que contenga una de estas palabras, el proxy denegará la conexión. Guardamos el archivo y volvemos a modificar squid.conf de la siguiente manera:

```
Acl expreg-denegadas url_regex "C:\squid\etc\deny.txt"
```

Buscamos las acl's y añadimos la línea que hay marcada en la imagen de arriba, donde especificamos que se deniegue el acceso a las url que contengan las palabras escritas en el archivo especificado. Ahora buscamos esta línea en squid.conf:

```
http_access allow localnet  
http_access allow localnet !expreg-denegadas
```

Después vamos a abrir el navegador para especificar el proxy.

Ahora reiniciamos el servicio y probamos a entrar en alguna web que contenga una palabra de las especificadas en el archivo.

Y podremos ver que el proxy nos está impidiendo entrar en la web.

- **CloudRadar:**

Para la monitorización de servicios en Windows solo hace falta registrarse en CloudRadar, configurar las redes y el servidor e iniciar sesión.

- **Mensajería Instantánea:**

Instalamos e iniciamos openfire.

Realizamos la configuración básica a través de la web.

Establecemos la cuenta de administrador y creamos usuarios.

Conectamos al servidor mediante el cliente Spark o Tkabber y vemos la conversación entre dos clientes.

- **RAID:**

Agregamos 2 discos para el RAID 1 y 3 discos para el RAID 5 mediante la configuración de VirtualBox.

Vamos al administrador de disco e inicializamos los discos.

Elegimos los discos a utilizar (2 para RAID 1 y 3 para RAID 5)

Seguimos los pasos para la creación de los RAID y finalizamos.

Para ver si se han creado correctamente vamos a Equipo, deberán aparecer como nuevo volumen cada RAID generada.