

# Administración de sistemas UNIX

## Práctica *Script maestro para la configuración de un cluster Linux*

Curso 2019/2020

### Introducción

Esta práctica consiste en la elaboración de un conjunto de *scripts* para facilitar a un administrador el proceso de configuración de un conjunto de máquinas que formen parte de un *cluster*. En concreto los *scripts* facilitarán la configuración de dispositivos de almacenamiento y las partes servidora y cliente de servicios típicos en un *cluster*.

### Normas básicas

- **Entrega:** 22 de diciembre de 2019. Se realizará a través de Moodle, en el apartado habilitado a tal efecto.
- **Grupos:** El trabajo se realizará en grupos de tres alumnos.

### 1. Sistema objetivo

El sistema sobre el que se debe realizar la práctica está formado por una serie de instancias de una máquina virtual VirtualBox con sistema operativo Debian. El trabajo debe realizarse **obligatoriamente en esta máquina**. La misma máquina virtual será utilizada después para la evaluación de las prácticas entregadas. Una copia de dicha máquina se encuentra disponible para su descarga en la siguiente URL:

- <https://laurel.datsi.fi.upm.es/~atorre/external/PracticaASI2019.tgz>

La máquina tienen un usuario, creado con el siguiente nombre y contraseña:

- Nombre: practicas
- Contraseña: .practic@s

En la máquina se han instalado las extensiones de cliente para facilitar el intercambio de ficheros entre la máquina virtual y la máquina anfitrión.

Los alumnos no deben hacer ninguna suposición acerca de cuántas máquinas formarán parte del sistema de corrección, ni tampoco cuáles serán sus nombres o IPs. Sin embargo, sí se podrá asumir que las máquinas ya tienen la configuración de red preestablecida. También se podrá asumir que un servidor de `ssh` está presente en todas las máquinas y que el usuario `root` puede conectarse de unas a otras sin tener que insertar contraseña. Los alumnos deberán llevar a cabo cierta preconfiguración en su propio entorno de pruebas. Para ello, deberán tener en cuenta los siguientes aspectos:

- Crear diferentes copias de la máquina virtual usando la opción *Clone* (clonar) de VirtualBox.
- Generar una nueva dirección MAC de las máquinas clonadas, para que sea distinta a la original.
- Crear una red NAT específica para todas las máquinas virtuales y asegurarse de que todas las máquinas virtuales estén conectadas a ella.

## 2. Objetivo de la práctica

El objetivo de esta práctica es programar un *script* `bash` que sea capaz de instalar y configurar de forma totalmente automática un conjunto de servicios en un grupo de máquinas conectadas a una red local para formar un cluster. El *script* se llamará `configurar_cluster.sh` y su funcionamiento será el siguiente:

1. El administrador inicia sesión en cualquiera de las máquinas de la red.
2. El administrador prepara un fichero de configuración del cluster, indicando qué servicios se deben instalar en cada máquina (el contenido de este fichero se explica en detalle a continuación).
3. El administrador ejecuta el *script* `configurar_cluster.sh`, pasándole el fichero de configuración del *cluster* como parámetro.
4. De forma totalmente automática, el *script* `configurar_cluster.sh` se irá conectando por `ssh` a cada una de las máquinas especificadas e instalando y configurando los servicios indicados.

El *script* podrá invocar *scripts* auxiliares proporcionados por los alumnos. Obsérvese que es responsabilidad del *script* principal copiar los *scripts* auxiliares a las máquinas objetivo de ser necesario. Resolver la práctica en uno o varios *scripts* queda completamente a la elección de cada grupo.

**IMPORTANTE:** Todas las operaciones de administración deberán realizarse dentro de los *scripts* (incluyendo la instalación de los paquetes necesarios, por ejemplo). El

objetivo es que la solución funcione correctamente ejecutada como superusuario sobre la máquina virtual **tal cual se proporciona**, obteniéndose la funcionalidad requerida sin ninguna operación manual adicional (con la salvedad de la pre-configuración de red y ssh antes mencionada). Cualquier solución que no cumpla con este requisito será considerada incorrecta.

### 3. Descripción detallada

El *script* principal que debe programarse se denominará `configurar_cluster.sh` y será invocado del siguiente modo:

```
$ ./configurar_cluster.sh fichero_configuración
```

Una vez ejecutado este mandato, el *script* debe ir informando por su salida estándar de las operaciones que realice. El formato de estos mensajes queda a elección de los alumnos, pero éstos deben ser suficientemente explicativos para poder seguir las acciones del *script*. Los mensajes de error deberán imprimirse por la salida de error estándar. Se podrá optar por redirigir al dispositivo nulo la salida de aquellos mandatos cuya salida dificulte la lectura de la traza de progreso.

Para facilitar el desarrollo del script por parte de los alumnos, el apéndice A contiene algunos ejemplos de ficheros de entrada y salidas esperadas, tanto cuando los ficheros de configuración son correctos como cuando contienen algún tipo de error.

#### 3.1. Fichero de configuración del cluster

El fichero de configuración del *cluster* será un fichero de texto plano. El *script* leerá el fichero y ejecutará cada una de las instrucciones de administración indicadas por orden. El formato del fichero de configuración del cluster debe ser el siguiente:

- Las líneas en blanco se ignorarán.
- Las líneas que empiecen por `#` (comentarios) se ignorarán.
- Las líneas de configuración deberán cumplir el siguiente formato:

```
maquina-destino nombre-del-servicio fichero-de-perfil-de-servicio
```

Cualquier línea que no cumpla este formato deberá ser detectada y considerada errónea. Se deberá imprimir un mensaje de error con la línea incorrecta por la salida de error estándar e interrumpir la ejecución.

Ésta es la descripción detallada de los parámetros de las líneas de configuración:

1. **maquina-destino**: El nombre o dirección IP de la máquina donde se desea realizar la configuración.
2. **nombre-del-servicio**: Servicio que se desea instalar y configurar.
3. **fichero-de-perfil-de-servicio**: Fichero de configuración adicional con los parámetros del servicio que se desea configurar.

### 3.1.1. Ejemplo de fichero de configuración

El siguiente ejemplo configura un RAID y un servidor de NFS en la máquina 192.168.0.1 y un cliente de NFS en las demás:

```
# Máquina servidora
192.168.0.1 raid raid.conf
192.168.0.1 mount mount_raid.conf
192.168.0.1 nfs_server nfs_server.conf

# Máquinas cliente
192.168.0.2 nfs_client nfs_client.conf
192.168.0.3 nfs_client nfs_client.conf
192.168.0.4 nfs_client nfs_client.conf
192.168.0.5 nfs_client nfs_client.conf
```

Como se puede ver, esta configuración hace uso además de cuatro ficheros de perfil de servicio (`mount_raid.conf`, `raid.conf`, `nfs_server.conf` y `nfs_client.conf`). El formato de estos ficheros se describe a continuación.

## 3.2. Servicios requeridos

El *script* `configurar_cluster.sh` deberá ser capaz de instalar y configurar una serie de servicios en las máquinas que se soliciten. Cada servicio tendrá unos parámetros asociados, dependiendo de su funcionalidad. Como norma general, el código debe comprobar que los parámetros de cada servicio son correctos, y mostrar qué errores se detectaron sin fallar, colgarse, dejar la máquina en un estado inutilizable, etc. En caso de detectarse algún error, el *script* debe devolver un código de error distinto de 0.

Además debe realizar de forma general las siguientes tareas, cuando sea necesario:

- Instalación de software necesario para cada servicio, si no está presente en la máquina, asegurándose de que no es necesaria ninguna intervención manual en este punto.
- Creación de directorios necesarios, como puntos de montaje, etc.

A continuación se describe cada uno de los servicios y sus parámetros en detalle.

### 3.2.1. Montaje

Montaje de un dispositivo en un punto del árbol de directorios. Esto implica además que el sistema debe ser configurado para que el dispositivo se monte siempre en ese punto al arrancar la máquina (fichero `/etc/fstab`). El nombre del servicio es `mount`. El fichero de perfil de servicio debe ser un fichero de texto plano con 2 líneas. El formato de estas líneas debe ser:

```
nombre-del-dispositivo
punto-de-montaje
```

El *script* debe crear el directorio escogido como punto de montaje si no existe. En caso de existir, deberá comprobar que el directorio está vacío.

Ejemplo de un fichero de perfil de servicio:

```
/dev/sda2
/home
```

### 3.2.2. RAID

Configuración de un RAID software, usando *mdadm*. El nombre del servicio es **raid**. El fichero de perfil de servicio debe ser un fichero de texto plano con 3 líneas. El formato de estas líneas debe ser:

```
nombre-del-nuevo-dispositivo-raid
nivel-de-raid
dispositivo-1 [dispositivo-2 ...]
```

A continuación se muestra un ejemplo de fichero de perfil de servicio. En este ejemplo se configura un RAID 1 usando los dispositivos `/dev/sdb1` y `/dev/sdc1`.

```
/dev/md0
1
/dev/sdb1 /dev/sdc1
```

### 3.2.3. LVM

Configuración de un grupo de volúmenes lógicos. El nombre del servicio es **lvm**. El fichero de perfil de servicio debe ser un fichero de texto plano con 3 o más líneas. El formato de estas líneas debe ser:

```
nombre-del-grupo-de-volúmenes
lista-de-dispositivos-en-el-grupo
nombre-del-primer-volumen tamaño-del-primer-volumen
nombre-del-segundo-volumen tamaño-del-segundo-volumen ...
```

El número de líneas de este fichero es variable, ya que depende de cuántos volúmenes lógicos se deseen crear. El *script* debe comprobar que no se excede el tamaño del grupo al crear los volúmenes.

A continuación se muestra un ejemplo de fichero de perfil de servicio. En este ejemplo se configura un grupo llamado **serverdata** usando los dispositivos `/dev/sdb1`, `/dev/sdc1` y `/dev/sdd1` y se crean dos volúmenes lógicos: **software** (de 100GB) y **users** (de 500GB).

```
serverdata
/dev/sdb1 /dev/sdc1 /dev/sdd1
software 100GB
users 500GB
```

#### 3.2.4. Servidor NIS

Configuración de un servidor de NIS. El nombre del servicio es **nis\_server**. El fichero de perfil de servicio debe ser un fichero de texto plano con 1 línea. El formato de esta línea debe ser:

```
nombre-del-dominio-nis
```

Ejemplo de un fichero de perfil de servicio:

```
clusterASI2014
```

#### 3.2.5. Cliente NIS

Configuración de un cliente de NIS. El nombre del servicio es **nis\_client**. El fichero de perfil de servicio debe ser un fichero de texto plano con 2 líneas. El formato de estas líneas debe ser:

```
nombre-del-dominio-nis  
servidor-nis-al-que-se-desea-conectar
```

Ejemplo de un fichero de perfil de servicio:

```
clusterASI2014  
192.168.0.1
```

#### 3.2.6. Servidor NFS

Configuración de un servidor de NFS con una serie de directorios que exportar. Dado que no se puede asumir ninguna configuración de red, los directorios deberán exportarse por todas las interfaces de red del servidor. El nombre del servicio es **nfs\_server**. El fichero de perfil de servicio debe ser un fichero de texto plano con 1 o más líneas. El formato de estas líneas debe ser:

```
ruta-del-primer-directorio-exportado  
ruta-del-segundo-directorio-exportado ...
```

El número de líneas de este fichero es variable, ya que depende de cuántos directorios se deseen exportar. Ejemplo de un fichero de perfil de servicio:

```
/home  
/usr/local
```

### 3.2.7. Cliente NFS

Configuración de un cliente de NFS. El nombre del servicio es `nfs_client`. El fichero de perfil de servicio debe ser un fichero de texto plano con una línea por sistema de ficheros que debe montarse. El formato de estas líneas debe ser:

```
ip-servidor ruta-de-directorio-remoto punto-de montaje
...
```

El número de líneas de este fichero es variable, ya que depende de cuántos directorios remotos se deseen montar. Esto implica además que el sistema debe ser configurado para que el directorio remoto se monte siempre en ese punto al arrancar la máquina (fichero `/etc/fstab`). Los sistemas de ficheros deberán montarse con permisos de lectura y escritura para todos los usuarios. De no existir, el directorio con el punto de montaje deberá ser creado. Ejemplo de un fichero de perfil de servicio:

```
192.168.0.1 /home /home
192.168.0.1 /usr/local /usr/local
```

### 3.2.8. Servidor de backup

Configuración de un servidor de backup. El nombre del servicio es `backup_server`. El fichero de perfil de servicio debe ser un fichero de texto plano con 1 línea. El formato de esta línea debe ser:

```
directorio-donde-se-realiza-el-backup
```

Ejemplo de un fichero de perfil de servicio:

```
/storage/backup
```

### 3.2.9. Cliente backup

Configuración de un cliente de backup. El nombre del servicio es `backup_client`. El fichero de perfil de servicio debe ser un fichero de texto plano con 4 líneas. El formato de estas líneas debe ser:

```
ruta-del-directorio-del-que-se-desea-hacer-backup
dirección-del-servidor-de-backup
ruta-de-directorio-destino-del-backup
periodicidad-del-backup-en-horas
```

El *script* debe configurar el backup para que se realice de forma periódica. En este ejemplo de fichero de perfil de servicio fijamos que se haga un backup del directorio `/home` cada 24 horas dentro del directorio `/storage/backup/node5` de la máquina `192.168.0.1`:

```
/home
192.168.0.1
/storage/backup/node5
24
```

**Nota sobre los servicios de backup:** Se deja a decisión de los alumnos la elección de la tecnología de backup empleada en la práctica. Se debe justificar esta elección en la memoria. Se recuerda a los alumnos que no todas las tecnologías posibles siguen un enfoque cliente-servidor (algunas emplean sólo un cliente, por ejemplo). El hecho de que los servicios especificados incluyan un cliente y un servidor no implica que la tecnología empleada tenga forzosamente que emplear este modelo. Se escoja la tecnología que se escoja, se debe poder configurar usando los mecanismos que se han indicado. Esto quiere decir, por ejemplo, que si se escoge una solución carente de parte servidora, la configuración de esta parte debe asegurarse, al menos, de que el directorio donde se realizará el backup exista y pueda ser escrito.

### 3.3. Memoria de la práctica

Por último, los alumnos deberán elaborar una breve memoria de la práctica, explicando el funcionamiento de la solución propuesta, así como cualquier dificultad o aspecto de interés. La memoria debe ser un documento formal, correctamente estructurado, redactado y formateado usando alguna herramienta de procesado y edición de textos (Microsoft Word, L<sup>A</sup>T<sub>E</sub>X, etc.) que permita su exportación a formato PDF. Debe contener figuras explicativas, tablas y otros recursos que ayuden a la comprensión del documento. **No se debe descuidar la redacción de la memoria**, ya que podría afectar negativamente a la nota final.

**IMPORTANTE:** La memoria **NO** debe contener el código del *script*, que se entregará aparte.

## 4. Entrega

El alumno deberá entregar la práctica a través de Moodle, en el apartado habilitado a tal efecto. La entrega deberá constar de **un único fichero comprimido** (zip, rar, tgz, ...). El contenido de dicho fichero debe ser:

- *script* `configurar_cluster.sh`.
- Memoria de la práctica (en pdf).
- Cualquier otro fichero auxiliar del que haga uso el *script* anterior.



## 5. Evaluación

La evaluación de la práctica se llevará a cabo usando un conjunto de instancias de la misma máquina virtual puesta a disposición del alumno para su realización. La calificación otorgada será de 0 a 10 puntos, repartidos de la siguiente forma:

Parte	Puntuación máxima
Montaje	0.25 puntos
Configuración RAID	1.25 puntos
Configuración LVM	1.25 puntos
Configuración NIS	1.25 puntos
Configuración NFS	1.25 puntos
Configuración backup	1.75 puntos
Estructuración, legibilidad y mantenibilidad del código	1 punto
Calidad de la memoria	2.00 puntos

La calificación obtenida supondrá un 60 % de la nota total de la asignatura.

## A. Ejemplos de entradas y salidas esperadas

En este apéndice se incluyen algunos ejemplos de ficheros de entrada que serán proporcionados a los *scripts* que entregar y el tipo de salida que se espera obtener para dichas entradas en el proceso de corrección. Esta documentación **no pretende ser una lista exhaustiva** de todos los casos que serán probados, pero sí proporcionar una idea de cómo es la batería de pruebas.

### A.1. Montaje (mount)

1. *Perfil de servicio recibido:*

```
/dev/sdb1
/un/directorio/que/no/existe
```

*Salida esperada:* Asumiendo que /dev/sdb1 existe, no debe producirse ningún error y el valor devuelto debe ser 0. El directorio /un/directorio/que/no/existe ha sido creado y /dev/sdb1 está montado sobre el mismo. Al reiniciar, /dev/sdb1 se monta automáticamente.

2. *Perfil de servicio recibido:*

```
/este/dispositivo/no/existe
/directorio
```

*Salida esperada:* Error de montaje, el dispositivo no existe o no puede ser montado. El directorio de montaje puede haber sido creado. Código de error distinto de 0.

3. *Perfil de servicio recibido:*

```
/dev/sdb1
```

*Salida esperada:* Error de sintaxis, falta el punto de montaje. Código de error distinto de 0.

### A.2. RAID (raid)

1. *Perfil de servicio recibido:*

```
/dev/md0
5
/dev/sdb1 /dev/sdc1 /dev/sdd1
```

*Salida esperada:* Se ha creado un dispositivo RAID nivel 5 llamado /dev/md0 que debe aparecer con ese nombre al reiniciar la máquina. El código de retorno debe ser 0.

2. *Perfil de servicio recibido:*

```
/dev/md0
17
/dev/sdb1 /dev/sdc1 /dev/sdd1
```

*Salida esperada:* Nivel de RAID inválido. El valor devuelto debe ser distinto de 0.

3. *Perfil de servicio recibido:*

```
array1
0
/dev/sda /dev/sdc1 /dev/sdd1
```

*Salida esperada:* Asumiendo que `/dev/sda` es un dispositivo que ya contiene un sistema de ficheros, el mandato fallará. Esta configuración es correcta en lo que respecta al nombre del dispositivo RAID que debe crearse. El valor devuelto debe ser distinto de 0.

### A.3. LVM (lvm)

1. *Perfil de servicio recibido:*

```
grupo1
/dev/sdb /dev/sdc
vol1 2G
vol2 4G
```

*Salida esperada:* Se ha creado un grupo LVM llamado grupo1 que abarca los dispositivos `/dev/sdb` y `/dev/sdc`. Dentro de ese grupo se han creado los volúmenes lógicos vol1 de 2G y vol2 de 4G. La información puede ser verificada con los mandatos `pvdisplay`, `vgdisplay` y `lvdisplay`. El valor devuelto debe ser 0.

2. *Perfil de servicio recibido:*

```
grupo1
/dev/sdb /dev/sdc
```

*Salida esperada:* Falta al menos un volumen lógico que crear. El valor devuelto debe ser distinto de cero.

3. *Perfil de servicio recibido:*

```
grupo1
/dev/sdb /dev/sdc
vol1 4G
vol2 20000G
```

*Salida esperada:* El tamaño de `vol2` es mayor que el disponible en el grupo de dispositivos. El volumen `vol1` puede haber sido creado o no. El valor devuelto debe ser distinto de cero.

#### A.4. Client de NIS (`nis_client`)

##### 1. *Perfil de configuración recibido:*

```
dominio
192.168.200.1
```

*Salida esperada:* Asumiendo que la IP pertenece al servidor de domino NIS, se configurará el cliente para poder usar dicho servidor para la autenticación. Por ejemplo, el superusuario será capaz de hacer `su usuario`, siendo `usuario` un nombre de usuario presente en el servidor. El *script* no tiene porque detectar que el servidor de dominio existe y puede ser contactado. El valor de retorno debe ser igual a cero.

##### 2. *Perfil de configuración recibido:*

```
dominio
esto.no.es.una.ip
```

*Salida esperada:* Impresión del error de sintaxis y devolución de un valor distinto de cero.

##### 3. *Perfil de configuración recibido:*

```
dominio
```

*Salida esperada:* Error por configuración incompleta y devolución de un valor distinto de cero.

#### A.5. Servidor de NFS (`nfs_server`)

##### 1. *Perfil de configuración recibido:*

```
/directorio/valido
```

*Salida esperada:* El servidor de NFS queda instalado y configurado para exportar en todas las subredes a las que está conectado. Debe ser posible montar cualquiera de los sistema de ficheros exportados desde cualquier máquina conectada a alguna de las subredes del servidor.

##### 2. *Perfil de configuración recibido:*

```
/directorio/invalido  
/directorio/valido
```

*Salida esperada:* Si un directorio no existe, debe imprimirse un error y no añadirlo a `/etc/exports`. El servidor de NFS puede quedar instalado o no. Queda a discreción del alumno decidir si continuar procesando el fichero o no, pero el *script* debe devolver un valor distinto de 0 en cualquier caso.

#### A.6. Cliente de NFS (`nfs_client`)

##### 1. *Perfil de configuración recibido:*

```
192.168.0.1 /export/home /home
```

*Salida esperada:* Asumiendo que el host 192.168.0.1 está exportando `/export/home`, el directorio se montará en `/home` y será accesible por cualquier usuario (los permisos de acceso de cada fichero y directorio se corresponderán con los exportados por el servidor). El valor de retorno debe ser 0. Al reiniciar la máquina cliente, los sistemas de ficheros deben ser montados al arranque.

##### 2. *Perfil de configuración recibido:*

```
192.168.0.1 /home
```

*Salida esperada:* Error por falta de parámetros de configuración. El valor de retorno debe ser distinto de 0.

##### 3. *Perfil de configuración recibido:*

```
host.invalido /home /home  
192.168.0.1 /shared /shared
```

*Salida esperada:* Error debido que el primero parámetro no es un nombre de host existente y el directorio remoto no puede ser montado (un error similar es que el servidor no esté exportando el directorio solicitado). Queda a discreción del *script* detenerse en el primer error o continuar, en cualquier caso el valor de retorno debe ser distinto de 0.

#### A.7. Backup en el servidor (`backup_server`)

##### 1. *Perfil de configuración recibido:*

```
/ruta_a_directorio_existente/
```

*Salida esperada:* Si el directorio no está vacío, error por intentar usar un directorio con contenido previo como almacén de backup (devuelve un código de salida distinto de cero). Si el directorio está vacío, el servidor de backup queda configurado correctamente.

2. *Perfil de configuración recibido:*

```
/ruta_a_directorio_inexistente/
```

*Salida esperada:* Error por directorio inexistente (código de salida distinto de cero).

## A.8. Backup en el cliente (backup\_client)

1. *Perfil de configuración recibido:*

```
/ruta_a_directorio_local_existente/  
10.0.0.1  
/ruta_a_directorio_remoto_existente/  
24
```

*Salida esperada:* Asumiendo que la IP pertenece al servidor de backup, si el directorio del servidor puede ser escrito desde el cliente, éste queda correctamente configurado. En caso contrario, error por falta de permisos de escritura en el directorio de backup (código de salida distinto de cero).

2. *Perfil de configuración recibido:*

```
/ruta_a_directorio_local_inexistente/  
10.0.0.1  
/ruta_a_directorio_remoto_existente/  
24
```

*Salida esperada:* Error por directorio local inexistente (código de salida distinto de cero).

3. *Perfil de configuración recibido:*

```
/ruta_a_directorio_local_existente/  
10.0.0.1  
/ruta_a_directorio_remoto_inexistente/  
24
```

*Salida esperada:* Error por directorio remoto inexistente (código de salida distinto de cero).

4. *Perfil de configuración recibido:*

```
/ruta_a_directorio_local_existente/  
10.0.0.355  
/ruta_a_directorio_remoto_existente/  
24
```

*Salida esperada:* Error por IP del servidor de backup incorrecta (código de salida distinto de cero).

#### A.9. Errores en el fichero principal

1. Perfil de configuración inexistente, no accesible o vacío.

*Salida esperada:* Impresión del error pertinente y devolver un valor distinto de 0.

2. *Perfil de configuración recibido:*

```
# Este fichero contiene errores de sintaxis  
  
# La línea en blanco no es un error  
192.168.0.1 raid raid.conf  
# En la línea siguiente falta el fichero de configuración  
192.168.0.1 mount
```

*Salida esperada:* Falta el fichero de configuración del servicio mount, debe imprimirse la línea errónea y devolver un valor distinto de 0.

3. *Perfil de configuración recibido:*

```
# Este fichero contiene un servicio que no existe  
# La línea en blanco no es un error  
192.168.0.1 trocola
```

*Salida esperada:* El servicio solicitado no existe. El valor devuelto debe ser distinto de 0.

## B. Configuración de VirtualBox

Esta sección cubre algunas pequeñas indicaciones para la llevar a cabo la configuración de VirtualBox necesaria para la elaboración de la práctica.

### B.1. Configuración de medios de almacenamiento

La adición de discos nuevos a una máquina virtual se hace a través del menú de configuración de dicha máquina (botón derecho sobre el nombre de la máquina). En el diálogo de configuración se selecciona la sección *Almacenamiento* en la lista de la izquierda y en el panel de la derecha se pincha en *Controlador: SATA* y luego en el icono de la derecha para añadir un disco duro.

Para eliminar un disco duro se selecciona el dispositivo que se quiere eliminar y o bien se pincha con el botón derecho para seleccionar la opción de eliminar o se pincha en el icono con un menos en la parte inferior.

### B.2. Configuración de red

En el enunciado indica explícitamente que las interfaces de red de las máquinas deben configurarse en modo *Red NAT*. Este modo, a diferencia del modo *Red interna* usado en el ejercicio IV de clase, requiere un paso de configuración adicional para poder asignar la red interna que usar. Este paso es la creación de la red, lo cual se puede hacer accediendo al diálogo de configuración global de red desde *Archivo → Preferencias*. En este diálogo, pinchar en la pestaña de *Redes NAT* y ahí añadir la red NAT con el nombre que se desee. Una vez completado este paso será posible asignar esa red a cualquier interfaz que configuremos en modo *Red NAT*.

Como ayuda a la elaboración más rápida de la práctica se recomienda trabajar con las máquinas virtuales en modo *headless*, es decir, sin acceder a ellas a través de su ventana de VirtualBox, si no estableciendo conexiones SSH directamente a las mismas. Para ello, configurar una interfaz adicional en las máquinas en modo *Adaptador sólo-anfitrión* y arrancar la máquina por la línea de mandatos (según se indica abajo). Cuando la máquina está arrancada debería aparecer una interfaz de red nueva en el anfitrión llamada *vboxnet0* (si no se ha cambiado el nombre por defecto), con una IP ya asignada y el enrutado configurado. Es posible que el servidor de SSH no esté instalado en la máquina cliente, en cuyo caso habría que hacerlo a través de una sesión con acceso gráfico.

En cuanto a la IP de una máquina cliente hay dos alternativas fundamentales: configurar la IP a mano utilizando la información de la red sólo-anfitrión usada, o dejar la configuración por defecto y averiguar qué dirección le está asignando el servidor DHCP.

La configuración manual es la opción más segura. El rango de IPs válido y la máscara que utilizar se pueden consultar en *Archivo → Preferencias →*, sección de *Red*, pestaña *Redes solo-anfitrión* y haciendo doble *click* sobre la red *vboxnet0*. Observar que esta es una red de acceso directo, por lo que no es necesario configurar ningún encaminador.



Para averiguar qué dirección asigna el servidor DHCP hay varias formas. Si se trata de la primera máquina que se conecta en dicho modo al anfitrión, su IP será probablemente 192.168.56.3. Esta IP es la primera del rango de direcciones del servidor de DHCP configurado para la red por defecto `vboxnet`. Las direcciones deberían asignarse incrementalmente en las siguientes máquinas que son arrancadas con una interfaz de este tipo. El rango exacto se puede configurar en la configuración de la red sólo anfitrión igual que antes. Otro modo de averiguar la IP es sencillamente arrancar la máquina en modo normal y consultar la IP en su configuración de red. Por último, el modo más inmediato es ejecutar el siguiente mandato desde el anfitrión: `VBoxManage guestproperty get nombre_de_maquina /VirtualBox/GuestInfo/Net/X/V4/IP`", donde *X* es el número de la interfaz sólo-anfitrión. Este último modo no necesita haber arrancado la máquina virtual en modo gráfico en ningún momento, pero es imprescindible que las extensiones de cliente estén funcionando, de lo contrario `guestproperty` no proporcionará dicha información.

En principio el tiempo de préstamo de las IPs es suficientemente alto para asumir que en combinación con la renovación, las direcciones otorgadas por el servidor de DHCP no cambian.

### B.3. Mandatos auxiliares

Esta sección recoge algunos mandatos que son útiles para gestionar las máquinas virtuales desde la consola sin necesidad de usar la interfaz gráfica de VirtualBox.

- Arrancar una máquina sin ventana gráfica: `VBoxManage startvm nombre_de_maquina --type headless`
- Apagar una máquina: `VBoxManage controlvm nombre_de_maquina poweroff`
- Crear una instantánea de una máquina: `VBoxManage snapshot nombre_de_maquina take nombre_de_instantánea`. Para que esto funcione no debe existir ninguna instantánea con el mismo nombre.
- Restablecer el estado de una máquina al de una instantánea: `VBoxManage snapshot nombre_de_maquina restore nombre_de_instantánea`. Si la máquina está arrancada primero será necesario apagarla.