

## **Creación de entornos de examen con LTSP**

Rafael Suárez Franco.  
Curso 2019 – 2020.

Proyecto integrado de ciclo.  
Grado Superior de Administración de Servicios Informáticos en Red.  
I.E.S. Triana.



## **Abstract - Español**

Un examen en un aula con equipos informáticos es significativamente distinto a los exámenes tradicionales, escritos en papel. El ordenador no es sólo una herramienta útil; es un elemento indispensable sin el que se concibe la realización de ciertas tareas, entre las que se encuentra la enseñanza en el ámbito de la informática. Del mismo modo que en un examen de dibujo técnico se necesitan los instrumentos de dibujo, un examen en el que el alumno debe demostrar y poner en práctica sus conocimientos en un área de la informática es difícil de concebir sin la presencia de los ordenadores. Por el contrario, los equipos informáticos pueden percibirse como un riesgo, dado que los usuarios más experimentados pueden hacer un uso indebido del ordenador para traspasar los límites de lo permitido en un examen justo. En este proyecto exponemos una solución para minimizar este tipo de riesgo, mediante Linux Terminal Server Project (LTSP), lo cual se traduce en que el alumno que realiza el examen solo dispondrá de las herramientas permitidas y de su propio conocimiento.

Palabras clave: informática, equipos informáticos, herramientas, riesgo, LTSP, exámenes.

## **Abstract - English**

Exams in classrooms with computer equipment are significantly different from paper-written, traditional exams. Computers are not just useful instruments – they are indispensable tools for the accomplishment of certain tasks, such as teaching in the computing field. Just like in a technical drawing exam, where drawing instruments are required, it is difficult to conceive a computing exam, where the student must demonstrate and put into action his knowledge, without the presence of computers. On the other hand, computer equipment may be perceived as a risk, since experienced users could make an improper usage of said computers to transgress the limits of what is allowed in a fair exam. In this project, we will present a solution to minimize this risk, using Linux Terminal Server Project (LTSP), which means that the student who takes the exam will only dispose of the allowed tools and his own knowledge.

Key words: computing, computer equipment, tools, risk, LTSP, exams.

## Objetivos

Los objetivos de este proyecto se pueden condensar en 3 puntos:

- Crear un entorno para exámenes de informática, en concreto, de gestión de bases de datos, pero adaptable a otras materias, en el que el profesor decida y controle las herramientas y recursos a los que tendrá el acceso el alumno. También deberá ser asequible, es decir, se debe poder aplicar en un aula o centro sin necesidad de invertir en software o comprar equipos adicionales.
- Exponer todo aquel software, herramienta, técnica, etc. que en conjunto se requieren para la creación de dicho entorno, además de consejos y recomendaciones acerca del uso de los mismos. Todo ello siguiendo un orden lógico, intercalando comprobaciones y aportando capturas de pantalla para mostrar al profesor qué resultados debe esperar al aplicar la solución de este proyecto.
- Garantizar que el profesor que desee llevar a la práctica este proyecto cuente con una serie de medidas de seguridad, de manera que no sea necesaria una supervisión estricta o individualizada de cada alumno durante la realización de examen.

## Tabla de contenidos

Introducción.....	9
Descripción del problema .....	9
Capítulo 1. Soluciones al problema. Base teórica .....	10
1.1 Linux Terminal Server Project (LTSP) .....	10
1.1.1 Requisitos técnicos para LTSP .....	11
1.1.2 Herramientas y protocolos de LTSP .....	11
1.2 Alternativas .....	13
Capítulo 2. Servidor y clientes .....	16
2.1 Opciones para el servidor .....	16
2.2 Definición del servidor y de la máquina virtual .....	17
2.2.1 Servidor .....	18
2.2.2 Máquina virtual .....	18
Capítulo 3. Instalación y configuración .....	20
3.1 Pasos previos .....	20
3.2 Instalación y configuración básica del servidor LTSP .....	22
3.3 Creación e inicio de un cliente ligero de pruebas .....	27
Capítulo 4 Resolución de problemas y configuración adicional .....	32
4.1 DHCP .....	32
4.2 TFTP .....	33
4.3 IPXE .....	35
4.4 Hostname .....	36
Capítulo 5. Usuarios .....	39
5.1 Análisis previo .....	39
5.2 Automatización de usuarios .....	41
5.2.1 Creación de usuarios .....	41
5.2.2 Reparto del examen .....	44
5.2.3 Recogida del examen .....	46
5.2.4 Borrado de usuarios .....	47
5.3 Restricción de ficheros .....	48
5.4 Autologin de usuarios .....	51
Capítulo 6. MySQL .....	54
6.1 Instalación del servidor MySQL .....	54
6.2 Creación de bases de datos y usuarios .....	56
6.3 Comprobación y uso de las bases de datos .....	58
6.4 Borrado de bases de datos y usuarios .....	63
Capítulo 7. Medidas de seguridad y control .....	65
7.1 Ssh y sftp .....	65
7.2 Restricted shell .....	68
7.3 Permisos de linux .....	74
7.4 Eptotes .....	78
Conclusiones.....	80
Lista de referencias.....	81

## Lista de figuras

Figura 1. Topología física del proyecto.....	9
Figura 2. Fichero de configuración de interfaces de red.....	23
Figura 3. Comando ifconfig, muestra las interfaces de red del equipo.....	24
Figura 4. Creación de una imagen con el comando ltsp image.....	25
Figura 5. Ventana de administrador de red de anfitrión.....	28
Figura 6. Adaptador sólo-anfitrión en la interfaz de red del cliente.....	28
Figura 7. Opciones de arranque del cliente.....	29
Figura 8. Menú de arranque de iPXE, muestra las imágenes que pueden arrancar los clientes....	30
Figura 9. Inicio de sesión de un cliente en la imagen de debian.....	30
Figura 10. El software específico para la realización del examen debe estar instalado en la imagen.....	31
Figura 11. Error de DHCP, el cliente no es capaz de encontrar el servidor y recibir una dirección IP.....	32
Figura 12. Error de TFTP, el servidor no es capaz de servir los archivos al cliente.....	33
Figura 13. Archivo de configuración de dnsmasq.....	34
Figura 14. Error de iPXE tras la selección de una imagen.....	35
Figura 15. Error de hostname, debido a que el cliente tiene un nombre de equipo inválido.....	37
Figura 16. Archivo de configuración de LTSP, sección de clientes.....	38
Figura 17. Creación y muestra de los nuevos usuarios.....	43
Figura 18. Uso de uno de los nuevos usuarios en el cliente.....	43
Figura 19. Reparto del examen y comprobación en los directorios de los usuarios.....	45
Figura 20. Visualización del examen en el cliente y creación de fichero de respuestas.....	45
Figura 21. Visualización del fichero de respuestas desde el servidor.....	46
Figura 22. Recogida del examen mediante copia de los directorios de los usuarios.....	47
Figura 23. Borrado de usuarios y de sus directorios.....	48
Figura 24. Fichero con lista de directorios que se ocultan a los clientes.....	49
Figura 25. Comprobación de un archivo de la lista de excluidos, desde el servidor.....	50
Figura 26. Comprobación de un archivo de la lista de excluidos, desde el cliente.....	50
Figura 27. Conversión de una cadena a base 64.....	52
Figura 28. Fichero de configuración de ltsp, sección de clientes.....	52
Figura 29. Comprobación del puerto de MySQL.....	55
Figura 30. Comprobación del demonio de MySQL.....	56
Figura 31. Creación de bases de datos y usuarios de MySQL.....	58
Figura 32. Usuarios creados.....	59
Figura 33. Bases de datos creadas.....	59
Figura 34. Comprobación del contenido de las bases de datos.....	59
Figura 35. Acceso a las bases de datos del servidor mediante el shell de MySQL.....	60
Figura 36. Uso y manipulación de la base de datos desde el cliente.....	61
Figura 37. Creación de una nueva conexión en MySQL workbench.....	61
Figura 38. Mensaje de conexión exitosa al servidor MySQL.....	62
Figura 39. Uso de MySQL workbench desde el cliente, utilizando la base de datos del servidor.....	62

Figura 40. Comprobación de borrado de bases de datos.....	64
Figura 41. Uso de ssh desde el cliente para acceder al servidor.....	66
Figura 42. Intento de uso de ssh tras configurar la restricción.....	67
Figura 43. Intento de uso de ssh y sftp.....	67
Figura 44. Comprobación de la imposibilidad del usuario de cambiar de shell.....	72
Figura 45. Uso de comandos permitidos.....	72
Figura 46. Intento de uso de comandos no permitidos.....	73
Figura 47. Comprobación de comandos del usuario desde el cliente.....	73
Figura 48. Cambio de permisos de la aplicación de firefox.....	76
Figura 49. Comprobando los nuevos permisos desde el cliente.....	76
Figura 50. Comprobación de la imposibilidad de encontrar la aplicación de firefox.....	77
Figura 51. La aplicación gráfica de epoptes muestra la actividad del cliente en tiempo real.....	78
Figura 52. Pantalla del cliente bloqueada por el profesor desde epoptes.....	79
Figura 53. Mensaje del profesor enviado al cliente.....	79



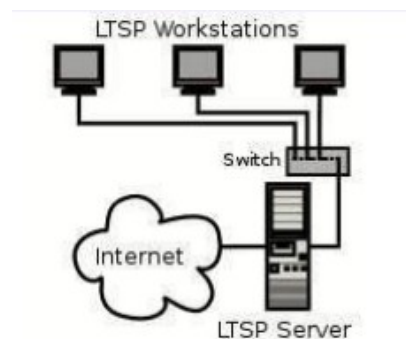
## Introducción

### Descripción del problema

En el entorno académico en el que trabajamos, normalmente debemos hacer exámenes en nuestros equipos informáticos, pero no es fácil para los profesores controlar que los alumnos no utilicen dichos equipos para hacer trampas en un examen. En un aula de informática, el ordenador del alumno está conectado a internet y a los equipos de sus compañeros, por lo que a la hora de realizar un examen, tenemos que asegurarnos de que estas vías de comunicación e información quedan cortadas.

Debemos tener en cuenta que el presupuesto y equipamiento de estas clases es, normalmente, limitado. Por tanto, no podemos permitirnos tener otra aula exclusiva para los exámenes, en la que los ordenadores solo tendrían las herramientas mínimas y necesarias, y cuya única conexión sería con un servidor, necesaria para la entrega del examen. Así pues, nuestro objetivo en este trabajo es emular dicho entorno en nuestra clase habitual. Debemos tener en cuenta que, a la hora de implementar una solución, la potencia de los equipos también es limitada, por lo que lo ideal sería utilizar la alternativa que demande los menos recursos posibles.

Buscaremos una solución que podamos implementar en un aula con unos 30 equipos, que estarán conectados al servidor mediante un switch.



*Figura 1: Topología física del proyecto*

# Capítulo 1

## Soluciones al problema. Base teórica

### 1.1 Linux Terminal Server Project (LTSP)

linux Terminal Server Project (LTSP Documentation, 2000-2020) es un software que permite la creación de servidores de clientes ligeros. En nuestro caso, nuestro servidor LTSP será el equipo del profesor, mientras que los ordenadores de los alumnos harán el papel de clientes ligeros. Como su propio nombre sugiere, los clientes ligeros son terminales cuya función principal es la entrada y salida de datos, ya que la carga de procesamiento la soportará el servidor LTSP. De este modo, podemos solventar el problema de la poca capacidad que puedan tener los clientes, siempre y cuando el servidor tenga los suficientes recursos. La capacidad del servidor, naturalmente, dependerá de la cantidad de clientes que necesitemos (en nuestro caso, unos 30).

Por otra parte, LTSP se basa en el uso de imágenes o máquinas virtuales, las cuales son necesarias para crear nuestros clientes ligeros. Dichos clientes actuarán como si trabajasen con un sistema operativo propio, el cuál realmente es una imagen o máquina virtual alojada en el servidor. LTSP permite usar plantillas para crear imágenes similares, y de este modo, tener varios clientes con copias de la misma máquina para trabajar en las mismas condiciones. Podríamos tener, por ejemplo, una plantilla para crear imágenes que tengan un servidor MySQL y MySQLWorkBench por defecto, para los exámenes de Bases de Datos.

### 1.1.1 Requisitos técnicos para LTSP

Según la página web de LTSP, para implementar el servidor de clientes ligeros, deberíamos cumplir con las siguientes especificaciones:

- Servidor: 4GB de RAM y un procesador con 3000 puntos en [cpubenchmark.net](https://cpubenchmark.net)<sup>1</sup>. Es recomendable un disco SSD. Sin embargo, como ya hemos mencionado, necesitaremos una capacidad más alta cuantos más clientes queramos añadir. Una tarjeta de red puede ser suficiente, aunque disponer de dos en el servidor nos ofrece otras opciones que veremos en la siguiente sección.
- Clientes: Dependerá de la distribución usada en las imágenes. Se estima un mínimo de 1 GB de RAM y una puntuación de 500 en [cpubenchmark](https://cpubenchmark.net), aunque lo recomendable es tener 2 GB de RAM y 2000 puntos o más.
- Red/cableado: Es recomendable que el servidor LTSP esté conectado al mismo switch al que se conectan los clientes. Por otra parte, la conexión entre servidor y switch debería ser de 1 Gigabit, mientras que bastaría con 10 Mbps para las conexiones entre clientes y switch.

### 1.1.2 Herramientas y protocolos de LTSP

LTSP hace uso de las siguientes herramientas y protocolos para cumplir su función. La instalación de LTSP implementará estas características con una configuración por defecto:

---

1 Nota: [cpubenchmark](https://cpubenchmark.net) (PassMark Software – CPU Benchmarks, 2020) es una página que realiza un ranking del rendimiento de distintas CPU. Podemos introducir nuestro modelo para saber qué puntuación tiene nuestro procesador.

- Preboot eXecution Environment (PXE): es un cargador de sistemas operativos a través de la red, nos permitirá cargar la imagen o máquina virtual en el servidor, desde el cliente ligero.
- Dynamic Host Configuration Protocol (DHCP): Sirve para asignar direcciones IP a los clientes. Si tenemos dos tarjetas de red, el servidor puede hacer la función de servidor DHCP.
- Trivial File Transfer Protocol (TFTP): Similar al protocolo FTP, se utiliza para transferir archivos entre equipos de una misma red, permitiendo arrancar los clientes ligeros desde el servidor.
- Network File System o Network Block Device (NFS o NBD) : Ambos permiten acceder a ficheros remotos en una red como si fuera locales, como si utilizáramos un disco de instalación.
- Domain Name System (DNS): Opcional. Para el uso de la caché DNS (revisar las páginas visitadas) o de las blacklists DNS (bloquear direcciones IP conocidas por hacer spam).
- Secure SHell, Secure SHell File System o Lightweight Directory Access Protocol (SSH, SSHFS o LDAP): Se puede usar cualquiera de ellos para autenticar al alumno a la hora de conectarse al servidor. Cabe destacar que necesitaremos SSHFS o NFS para que cada cliente pueda acceder a su directorio /home alojado en el servidor.
- Epopets: No es una herramienta indispensable de LTSP, pero se sugiere su instalación a la hora de montar el servidor LTSP, dado que es una software que nos permite monitorizar y controlar los clientes, por lo que es un añadido interesante, sobre todo en entornos de aulas o exámenes como el que queremos crear en este proyecto.

## 1.2 Alternativas

A continuación, expondremos una serie de opciones alternativas a LTSP. Antes de proceder a desarrollar una solución para el problema planteado, es recomendable valorar distintas posibilidades en primer lugar:

- Diskless Remote Boot in Linux: DRBL (Team. D., 2020) es muy similar a LTSP. Sirve para desplegar sistemas operativos Linux en varios clientes a la vez, en un esquema de servidor – clientes ligeros. Dichos clientes, de la misma manera, hacen uso de los recursos del servidor DRBL, el cuál necesitará mayor capacidad cuanto mayor sea el número de clientes que necesitemos. También utiliza PXE, NFS y otras herramientas similares a las que hemos visto en LTSP. En cuanto a sus requisitos mínimos, son menores, aunque se sugiere trabajar con distribuciones linux anteriores (Fedora 7, Ubuntu 7.04, etc). Además, se demanda un mínimo de 2 tarjetas de red en el servidor. No presenta ventajas evidentes frente a LTSP, y de acuerdo con la documentación, su implementación sería más complicada. No tenemos razones para preferir esta alternativa por encima de LTSP.
- ThinStation: ThinStation (Donald A. Cupp Jr., 2017-2020) es un sistema operativo de clientes ligeros basado en linux. Soporta la mayoría de protocolos de conectividad, por lo que podríamos usarlo con windows, citrix, etc. Esto también es posible gracias a que no requiere NFS, a diferencia de LTSP, sino que es opcional, por lo que es una ventaja a tener en cuenta. Por otra parte, ThinStation permite a los clientes ligeros realizar por su cuenta

ciertas tareas básicas, como la navegación en red, pero por otra parte, las aplicaciones que requieran más carga de trabajo se centralizan en un servidor, el cual administra los clientes. Por lo demás, ThinStation funciona de manera similar a las anteriores opciones, mediante el uso de imágenes en sus clientes ligeros. En cuanto a los requisitos de los equipos, dependerán de si centralizamos toda la carga de trabajo o si permitimos que cada cliente soporte ciertas tareas de manera independiente. En cualquier caso, tales requisitos no serían elevados.

- NoMachine Terminal Server: Basado en NX Linux Terminal Server, NoMachine Terminal Server (NoMachine S.à r.l., 2002) permite desplegar máquinas linux alojadas en un servidor a través de internet, usando SSH o TLS/SSL. Es una ventaja, aunque no realmente necesaria en nuestro caso, dado que los exámenes son presenciales. Otra ventaja notable es su escalabilidad, podemos crear clústers con múltiples nodos de servidores NoMachine si necesitamos mayor capacidad para servir a más clientes a la vez, o simplemente, para ofrecer alta disponibilidad, en caso de que algún nodo falle. De acuerdo con la página web, podemos utilizar casi cualquier dispositivo como cliente (ya sea windows, linux, mac, incluso android/iOS). Las aplicaciones que pueden ser utilizadas por dichos clientes están, por supuesto, alojadas y controladas desde el servidor. Podemos concluir que es una buena alternativa. Sus aportaciones y ventajas no necesarias en nuestro caso, pero amplía las posibilidades, por ejemplo, para hacer exámenes online. Por otra parte, hay que mencionar que requiere una suscripción de pago, lo cual es una desventaja frente a las demás alternativas.

Debemos concluir, antes de continuar, si deberíamos utilizar alguna de estas alternativas en lugar de LTSP. DRBL es la opción más parecida, y por tanto, no hay necesidad de elegirla para sustituir a nuestra idea original. ThinStation es una alternativa interesante, sobre todo si queremos trabajar con distintos sistemas operativos que no requieran NFS. Sin embargo, no la utilizaremos en este proyecto, dado que LTSP cuenta con más opciones a la hora de crear imágenes. Además, si queremos usar máquinas virtuales, ThinStation sólo ofrece una iso pre-configurada para funcionar como servidor. No es una mala opción, pero no nos permite elegir otras distribuciones distintas para dicho propósito. Por último, NoMachine es otra buena alternativa, aunque no realmente necesaria por las ventajas que pueda aportar. Sin embargo, es un software de pago, por lo que no tenemos más remedio que descartarlo para nuestro proyecto. Sería una opción más apropiada para academias/exámenes a distancia. En conclusión, haremos uso de LTSP para desarrollar la solución al problema que planteamos en este documento. Las alternativas son perfectamente viables, pero no aportan una utilidad adicional significativa que nos haga decantarnos por alguna de ellas, teniendo en cuenta el entorno académico en el que queremos poner en práctica este proyecto.

## Capítulo 2

### Servidor y clientes

#### 2.1 Opciones para el servidor

A la hora de crear una imagen en el servidor LTSP, tenemos 3 alternativas:

- Chrootless: Se utiliza la raíz del servidor para crear la imagen. Es decir, los clientes tendrán un sistema operativo idéntico al servidor, incluyendo sus programas, archivos, etc. Su ventaja principal es que no hay necesidad de mantener una máquina virtual independiente, por lo que podemos ahorrar recursos si las especificaciones de nuestro servidor son limitadas. Sin embargo, el hecho de desplegar copias exactas del servidor puede suponer una desventaja, ya que los clientes deberán soportar la misma arquitectura que el servidor. Además, debemos limitar el acceso a aquellos programas del servidor a los que no deseamos que accedan los clientes. Para ello, LTSP cuenta con un parámetro de configuración que permite “enmascarar” servicios.
- Chroot: Esta opción nos permite crear la imagen en un directorio específico del servidor, abarcando todo el contenido dentro de dicho directorio. Es decir, la imagen que utilizarán los clientes no se construye a partir de la raíz del servidor ( / ) sino de un directorio de nivel inferior del mismo, el cual podemos especificar (por ejemplo /home/examen). Los clientes estarán “enjaulados” en dicho directorio, por lo que no tendrán acceso a otras instancias de mayor nivel.



- Máquina virtual: Podemos elegir esta opción si preferimos generar una imagen a partir de una máquina virtual, independiente de la máquina del servidor. Es decir, los clientes tendrán un sistema operativo idéntico a una máquina virtual alojada en el servidor LTSP. Es la alternativa que más cuesta mantener, dado que hay que destinar recursos a dos máquinas distintas, pero a cambio, tenemos la ventaja de poder configurar una máquina con el sistema operativo y los programas específicos que deseemos, independientemente de la arquitectura, software, etc. del servidor.

Podemos concluir en este apartado que nuestra elección depende de los recursos de las máquinas de nuestra aula, y del grado de independencia que queremos entre el servidor y la imagen que utilizaremos en los clientes. Para este proyecto, vamos a elegir la opción de la máquina virtual, dado que nuestro servidor no tendrá problemas de escasez de recursos. Además, elegir el sistema operativo de la máquina virtual es una ventaja importante, dado que cabe la posibilidad de que algún programa necesario para los exámenes no sea compatible con el SO del servidor LTSP. Expondremos la importancia de esta ventaja en el siguiente apartado.

## **2.2 Definición del servidor y de la máquina virtual**

Debido a que hemos elegido trabajar con LTSP, y dado que nuestro objetivo final es poder hacer exámenes, en principio, de Gestión de Bases de Datos, debemos tener en cuenta que no podemos usar cualquier máquina con cualquier sistema operativo para este propósito. Ya

hemos hablado de los requisitos técnicos de los equipos, pero ahora tenemos que especificar un poco más acerca de los programas y sistemas operativos que vamos a utilizar.

### 2.2.1 Servidor

Según la página de LTSP (LTSP Documentation, 2000-2020), es muy recomendable usar un sistema operativo con escritorio para hacer el papel de servidor. Es posible tener un servidor sin entorno gráfico, pero no es la mejor opción en este caso. En concreto se recomiendan MATE y GNOME, pero en principio, cualquier entorno de escritorio debería funcionar sin problemas. Como ya hemos dicho, necesitaremos un sistema operativo linux, aunque LTSP sugiere concretamente que trabajemos con distribuciones basadas en Debian. No debería haber ningún problema dado que los profesores suelen trabajar con estas distribuciones en clase, ya sea Debian 9, Ubuntu 16.04, etc. Por ejemplo, en mi caso, estoy usando un Ubuntu 18.04 MATE para hacer pruebas en mi entorno de trabajo personal. En cuanto a los programas, dependerá de las herramientas que necesitemos a nuestra disposición, pero en un principio, ya sabemos que vamos a tener que instalar el software de LTSP y un programa de virtualización, por ejemplo, VirtualBox, dado que vamos a crear una máquina virtual para nuestros clientes.

### 2.2.1 Máquina virtual

La elección del sistema operativo para la máquina virtual es más importante, dado que es posible que alguna opción nos convenga más que otra. Como ya hemos dicho, necesitaremos instalar un servidor MySQL y MySQL Workbench, por lo que debemos tenerlo en cuenta a la hora de elegir. Por ejemplo, en primer lugar, hice pruebas con una máquina virtual Debian 10,

pero daba problemas a la hora de instalar el Workbench, por lo que es una opción desaconsejable. Por otra parte, he podido instalar y probar el Workbench en un Debian 9, por lo que podemos usar este sistema para nuestro proyecto concreto. También podemos usar un Ubuntu, pero en definitiva, la elección dependerá de las preferencias o necesidades del profesor. Recordamos que esta máquina virtual servirá para crear la imagen que utilizarán los clientes. La idea es que dicha máquina solo tenga el software necesario para la realización de un examen, por lo que si podemos evitar usar Ubuntu en un principio, sería mejor, dado que suele traer más programas por defecto que son innecesarios.

En resumen, tenemos un servidor LTSP con un SO Linux (concretamente Ubuntu 18.04) con una máquina virtual Debian 9 en VirtualBox, la cual utilizaremos para crear la imagen, que será aquella que utilicen los clientes o alumnos.

## Capítulo 3

### Instalación y configuración

#### 3.1 Pasos previos

Antes de instalar el servidor LTSP, vamos a crear la máquina virtual que describimos en el capítulo anterior. En nuestro caso, vamos a usar Oracle VM Virtualbox (Oracle Corporation, 2010), pero podemos utilizar otros softwares de virtualización (KVM, VMWare, etc) si lo deseamos. Para instalar Virtualbox tenemos dos opciones:

- Descargar la última versión desde [www.virtualbox.org](http://www.virtualbox.org). Una vez descargado, vamos a la carpeta de descargas e instalamos con el comando `dpkg -i nombre_del_paquete`.
- Instalar la versión 5 con el comando `apt-get install virtualbox`. Esta opción es perfectamente viable para nuestro proyecto.

Vamos a instalar también el extension pack de virtualbox. No es necesario para el objetivo final del proyecto, pero si queremos comprobar si el servidor ltsp funciona, podemos hacerlo con un cliente en Virtualbox. Podemos encontrar el paquete en la misma página e instalarlo al iniciar Virtualbox, o podemos usar el comando `apt-get install virtualbox-ext-pack` si hemos instalado el programa mediante la terminal. A continuación, ejecutamos `modprobe vboxdrv`.

Por otra parte, necesitaremos la imagen de la máquina que queremos crear. En mi caso, escogí una iso de debian 9.12.0 amd64, la cual podemos encontrar en [www.debian.org/distrib](http://www.debian.org/distrib) (SPI, 1997). Una vez la tengamos, podemos proceder a crear la máquina virtual. Para poder usarla con LTSP, es importante seguir una configuración específica:

- Tipo: Linux; Versión: Debian (64-bit).
- Crear disco duro virtual con tipo VMDK (Virtual Machine Disk).
- Seleccionar tamaño fijo (debemos asignar un espacio superior al mínimo exigido).

El resto de la configuración dependerá de nuestras preferencias. Una vez creada, debemos ir a Configuración → Almacenamiento y agregar la iso que hemos descargado. Podemos configurar la memoria asignada, la memoria de vídeo, etc. antes de continuar, si es preciso. Por último, iniciamos la máquina y procedemos a la instalación del sistema operativo. Seleccionamos la opción de “instalación gráfica”, nuestro idioma, región, etc. Tendremos que crear un usuario con contraseña, además de establecer la contraseña de superusuario. En cuanto al particionado de discos, seleccionaremos el disco que hemos creado anteriormente (escogemos la opción de usar todo el espacio de dicho disco). También instalaremos el cargador de arranque GRUB en el mismo. La opción de instalar el entorno de escritorio normalmente está seleccionada por defecto. Nos aseguramos de que esté marcada, ya que necesitaremos el escritorio para trabajar con MySQL Workbench.

Una vez terminada la instalación, antes de continuar con LTSP, vamos a instalar los programas que deseamos para nuestro caso particular. Podemos modificar la máquina tantas veces como haga falta, pero del mismo modo, tendremos que actualizar la imagen cada vez que queramos trasladar dichos cambios a nuestros clientes. De momento vamos a ignorar la

configuración de MySQL, simplemente ejecutaremos los siguientes comandos para tener la instalación básica terminada:

```
apt-get update
```

```
apt-get install mysql-server
```

```
apt-get install mysql-workbench
```

Una vez termine la instalación, podemos apagar la máquina y proceder con la instalación del servidor LTSP.

### 3.2 Instalación y configuración de LTSP

En primer lugar, vamos a añadir el repositorio LTSP PPA (LTSP Documentation - Personal Package Archive), necesario para instalar la última versión de LTSP (2019) en distribuciones basadas en debian. Simplemente ejecutaremos los siguientes comandos (con sudo o como root):

```
add-apt-repository ppa:ltsp
```

```
apt-get update
```

A continuación, instalamos los paquetes del servidor LTSP:

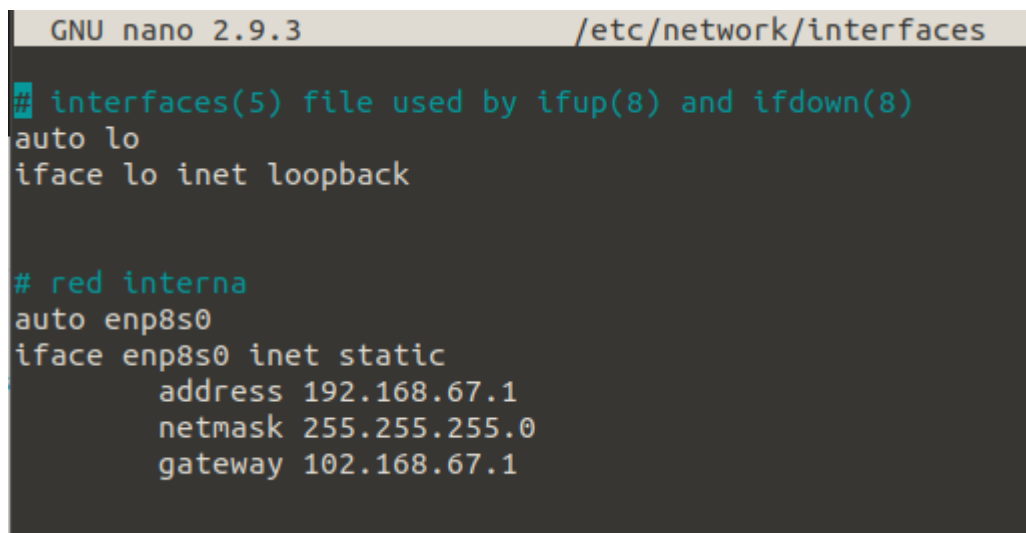
```
apt install --install-recommends ltsp ltsp-binaries dnsmasq nfs-kernel-server openssh-server squashfs-tools ethtool net-tools epoptes
```

```
gpsswd -a <nombre de administrador> epoptes
```

 (añade el administrador al grupo epoptes, el cuál podrá controlar los equipos desde el servidor.)

Como podemos ver, estos paquetes son las herramientas que utiliza LTSP, las cuales ya vimos en el primer capítulo. A continuación, vamos a realizar una configuración inicial, la cual podemos modificar si es necesario. En cuanto a la configuración de red, dependerá de si el servidor LTSP hará las funciones de servidor DHCP o no. Como ya explicamos, esto depende de si tenemos una o dos tarjetas de red en el servidor. Si tenemos solo una, debemos contar con servidor DHCP externo, por ejemplo un router. En dicho caso, ejecutamos `ltsp dnsmasq`. Por otra parte, si contamos con 2 tarjetas, podemos usar una de ellas para ofrecer el servicio DHCP. Si elegimos esta opción, debemos asignar una IP estática a la interfaz de la NIC que esté conectada a la red donde se encuentran los clientes. Es recomendable usar concretamente la IP 192.168.67.1, dado que la configuración por defecto de DHCP la utiliza como la dirección del servidor, pero podemos usar otra distinta si así lo deseamos (tendríamos que modificar los ámbitos, rangos, broadcast, etc). En cualquier caso, una vez hayamos establecido la IP de la interfaz, simplemente ejecutamos el comando de configuración, utilizando un parámetro que indica que el servidor LTSP no precisará los servicios de un servidor DHCP externo:

- Asignamos la IP en `/etc/network/interfaces`

A screenshot of a terminal window showing the contents of the file `/etc/network/interfaces` using the GNU nano 2.9.3 editor. The file contains configuration for two network interfaces: `lo` (loopback) and `enp8s0` (internal red). The `lo` interface is configured with `auto lo` and `iface lo inet loopback`. The `enp8s0` interface is configured with `auto enp8s0`, `iface enp8s0 inet static`, and static IP settings: `address 192.168.67.1`, `netmask 255.255.255.0`, and `gateway 102.168.67.1`.

```
GNU nano 2.9.3 /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

# red interna
auto enp8s0
iface enp8s0 inet static
    address 192.168.67.1
    netmask 255.255.255.0
    gateway 102.168.67.1
```

Figura 2: Fichero de configuración de interfaces de red.

```
/etc/init.d/networking restart
```

Comprobamos con ifconfig

```
root@rafael-ubuntu:/usr/share/ltsp# ifconfig
enp8s0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.67.1 netmask 255.255.255.0 broadcast 192.168.67.255
```

*Figura 3: Comando ifconfig, muestra las interfaces de red del equipo.*

```
ltsp dnsmasq --proxy-dhcp=0
```

Vamos a desarrollar el resto de la configuración suponiendo que utilizamos la configuración de doble tarjeta de red. Recordamos que los clientes idealmente están conectados al servidor mediante un switch, y que la conexión a internet de dichos clientes, por tanto, viene directamente del servidor, que dará un servicio de DHCP. En caso de que los clientes estén conectados al router, deberíamos buscar una alternativa para restringir el acceso a internet, por ejemplo, limitar dicho acceso dentro de la propia imagen que utilizan los clientes.

A continuación, debemos crear la imagen utilizando la máquina virtual. Para ello, simplemente vamos a crear un enlace simbólico del disco de la máquina en la carpeta donde se almacenan las imágenes (/srv/ltsp) y ejecutaremos otro comando de ltsp, necesario para exportar cualquier cambio en la máquina virtual a la imagen:

```
ln -rs /home/propietario_de_la_VM/VirtualBox\ VMs/debian9/debian9-flat.vmdk /srv/ltsp/  
nombre_de_la_imagen.img
```



`ltsp image nombre_de_la_imagen`

```
root@rafael-ubuntu:/home/rafael# ln -rs /home/rafael/VirtualBox\ VMs/debian9/debian9-flat.vmdk
/srv/ltsp/debian9.img
root@rafael-ubuntu:/home/rafael# ltsp image debian9
Running: losetup -rP /dev/loop0 /srv/ltsp/debian9.img
Running: mount -t tmpfs -o mode=0755 tmpfs /tmp/tmp.yBLAan1Yig/tmpfs
Running: mount -t ext4 -o ro,noload /dev/loop0p1 /tmp/tmp.yBLAan1Yig/tmpfs/0/looproot
Running: mount -t overlay -o upperdir=/tmp/tmp.yBLAan1Yig/tmpfs/0/up,lowerdir=/tmp/tmp.yBLAan1Y
ig/tmpfs/0/looproot,workdir=/tmp/tmp.yBLAan1Yig/tmpfs/0/work /tmp/tmp.yBLAan1Yig/tmpfs /tmp/tmp
.yBLAan1Yig/root/
Cleaning up debian9 before mksquashfs...
Can't load /root/.rnd into RNG
140278417289664:error:2406F079:random number generator:RAND_load_file:Cannot open file:../crypt
o/rand/randfile.c:88:Filename=/root/.rnd
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/tmp/tmp.yBLAan1Yig/root/etc/ssl/private/ssl-cert-snakeoil.key'
-----
Parallel mksquashfs: Using 4 processors
Creating 4.0 filesystem on /srv/ltsp/images/debian9.img.tmp, block size 131072.
[====- ] 7973/143291 5%
```

Figura 4: Creación de una imagen con el comando `ltsp image`.

En mi caso particular, he llamado a la imagen “debian9”. Si vamos a tener varias imágenes dependiendo del tipo de examen, es importante ponerles un nombre que nos permita diferenciarlas.

Si este último comando falla, normalmente será por un error en el enlace. Alternativamente, podemos hacer referencia a la ruta absoluta, sin necesidad de usar enlaces:

`ltsp image /home/propietario_de_la_VM/VirtualBox\ VMs/debian9/debian9-flat.vmdk`

El proceso de crear la imagen es el que más tarda con diferencia, por lo que es recomendable que instalemos/configuremos todo lo necesario en la máquina virtual antes de ejecutar el comando `ltsp image`. Una vez creada, podemos hacer cambios en la máquina y actualizar la imagen volviendo a ejecutar el comando `ltsp image nombre_de_la_imagen`. El proceso de actualización también tarda un tiempo considerable, por lo que del mismo modo, es

aconsejable asegurarse de que hemos realizado todos los cambios necesarios antes de exportarlos y actualizar la imagen.

A continuación, procedemos a configurar el resto de herramientas. Para crear un menú iPXE y exportar los binarios iPXE al servidor TFTP, utilizamos:

```
ltsp ipxe
```

Para que el servidor utilice NFS para servir las imágenes, ejecutamos:

```
ltsp nfs
```

El siguiente comando también será necesario para poder iniciar la imagen, creando un initrd secundario en /srv/tftp/ltsp/nombre\_de\_la\_imagen.img:

```
ltsp initrd
```

Cada vez que cambiemos la configuración del servidor LTSP, necesitaremos ejecutarlo. Además, sirve para actualizar los usuarios que puedan usar una imagen.

Por último, dado que estamos utilizando el método de máquinas virtuales, para copiar el kernel y el initrd de la imagen en el servidor TFTP, ejecutamos:

```
ltsp kernel
```

De este modo, la configuración básica estaría terminada, aunque es probable que el servicio de LTSP no funcione todavía en este punto. Por tanto, en apartados posteriores, explicaremos los errores y problemas que normalmente nos encontraremos, y expondremos las soluciones necesarias para solventarlos. Pero antes, vamos a definir el método con el que podemos testear el servicio de manera local.

### 3.3 Creación e inicio de un cliente ligero de pruebas

Si tenemos la estructura ya montada, o si disponemos de un equipo que pueda hacer de cliente conectado por cable a nuestro servidor, podemos probar si nuestra configuración es suficiente para empezar a servir la imagen. En caso contrario, podemos emular un cliente ligero en Virtualbox, con el que intentaremos cargar la imagen de la máquina debian.

Para ello, iniciamos Virtualbox y creamos una nueva máquina. Seleccionamos ubuntu de 64 bits, establecemos la cantidad de memoria RAM que necesitemos (al menos unos 2Gb) y a continuación no crearemos ningún disco virtual. Seleccionamos “no agregar un disco duro virtual”, dado que éste estará virtualizado en el servidor. Ignoramos la advertencia y ya tendremos el cliente en nuestra lista de máquinas virtuales.

Lo siguiente es conectarla al servidor, es decir, a la máquina anfitrión. La mejor opción para ello es el método de sólo anfitrión. Antes de poder asignarle este tipo de adaptador red, debemos crear una red solo-anfitrión. Para ello, vamos a Herramientas Globales → Administrador de red de anfitrión y crearemos una red nueva. Editamos esta nueva red en “Propiedades” y le asignaremos la IP 192.168.67.1/24. Recordamos que esta IP la asignamos a la interfaz de la red interna, donde se encuentran los clientes. Sería recomendable que vayamos a `/etc/network/interfaces` y comentemos la configuración que añadimos anteriormente.

Alternativamente, si queremos testear el método con una sola tarjeta de red, podemos seleccionar la opción de Habilitar Servidor DHCP, sin necesidad de configurar la red.

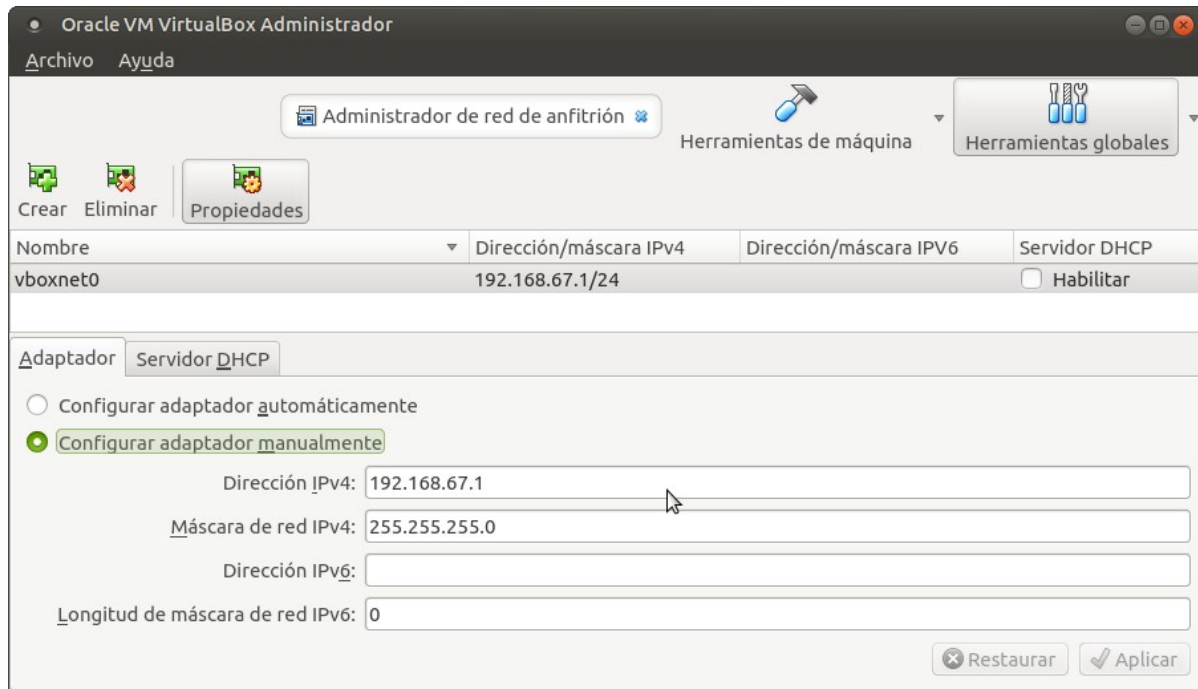


Figura 5: Ventana de administrador de red de anfitrión.

Ahora ya podemos ir a la configuración del cliente y habilitar el adaptador sólo anfitrión.

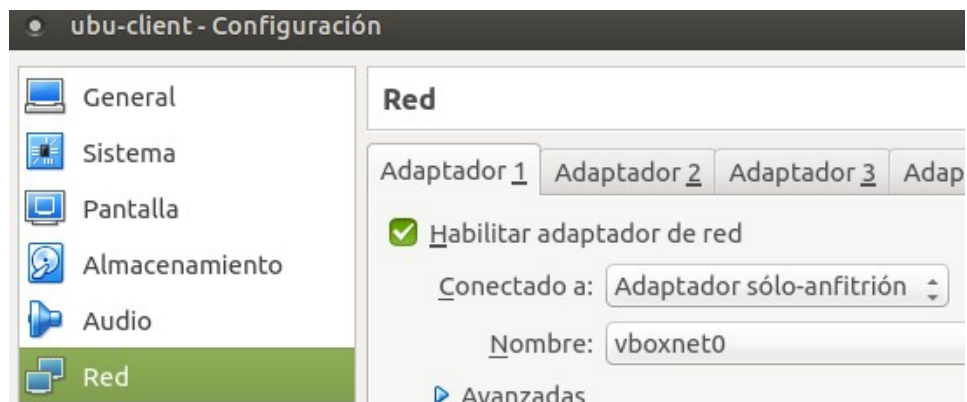


Figura 6: Adaptador sólo-anfitrión en la interfaz de red del cliente.

Adicionalmente, debemos ir a Sistema y habilitar la opción de Red en “Orden de arranque”. Podemos deshabilitar el resto para que el cliente intente iniciar desde la red automáticamente. De este modo, podemos proceder a iniciar el cliente para comprobar si nuestro servicio LTSP funciona correctamente.

En primer lugar (si no tenemos Red como única opción arranque) pulsamos F12 para mostrar las opciones de arranque.

```
VirtualBox temporary boot device selection
Detected Hard disks:
No hard disks found
Other boot devices:
f) Floppy
c) CD-ROM
l) LAN
b) Continue booting
```

*Figura 7: Opciones de arranque del cliente.*

Como podemos ver, nuestro cliente no tiene ningún disco. Pulsamos la tecla “l” para arrancar desde LAN. Si todo va bien, deberíamos ver como el cliente recibe una dirección IP de la red 192.168.67.0. A continuación, accederemos al menú de iPXE.

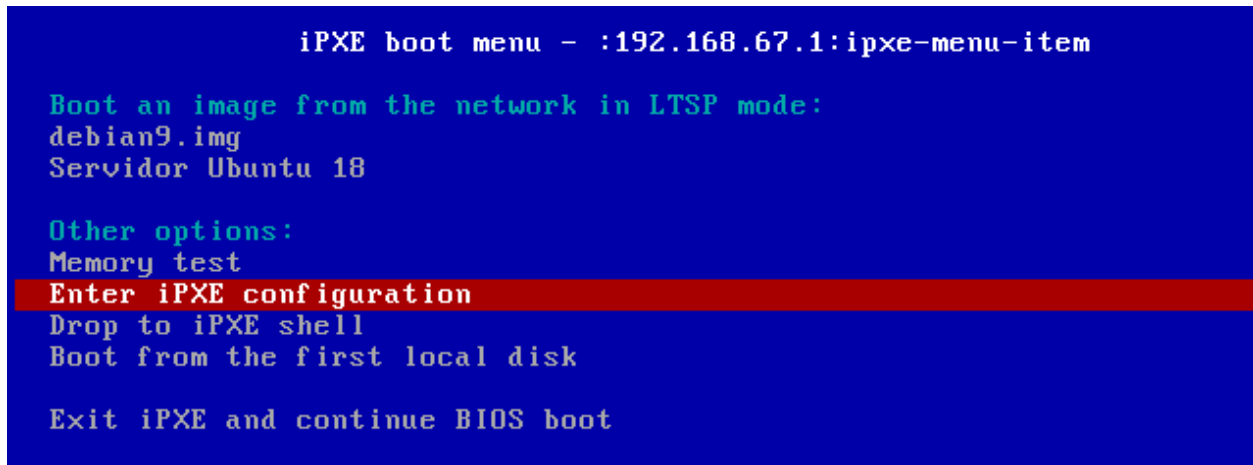


Figura 8: Menú de arranque de iPXE, muestra las imágenes que pueden arrancar los clientes.

En mi caso, tengo dos imágenes dado que también hice una imagen utilizando la raíz del servidor (método chrootless). Seleccionamos la imagen debian9.img, si funciona correctamente, deberíamos ver la pantalla de inicio de la máquina debian, con los usuarios que hubiéramos creado.

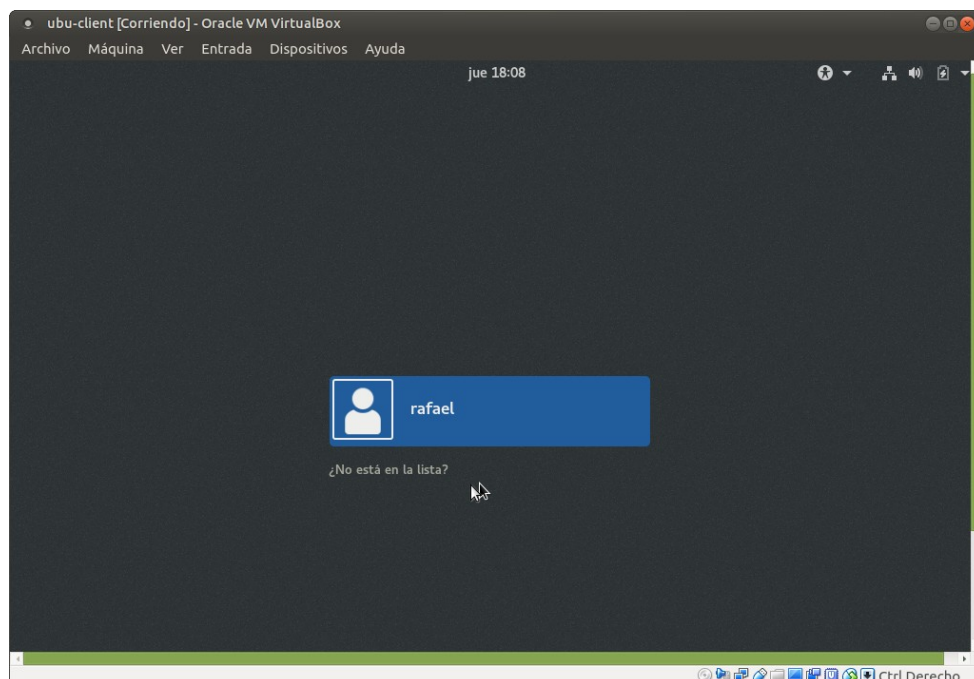
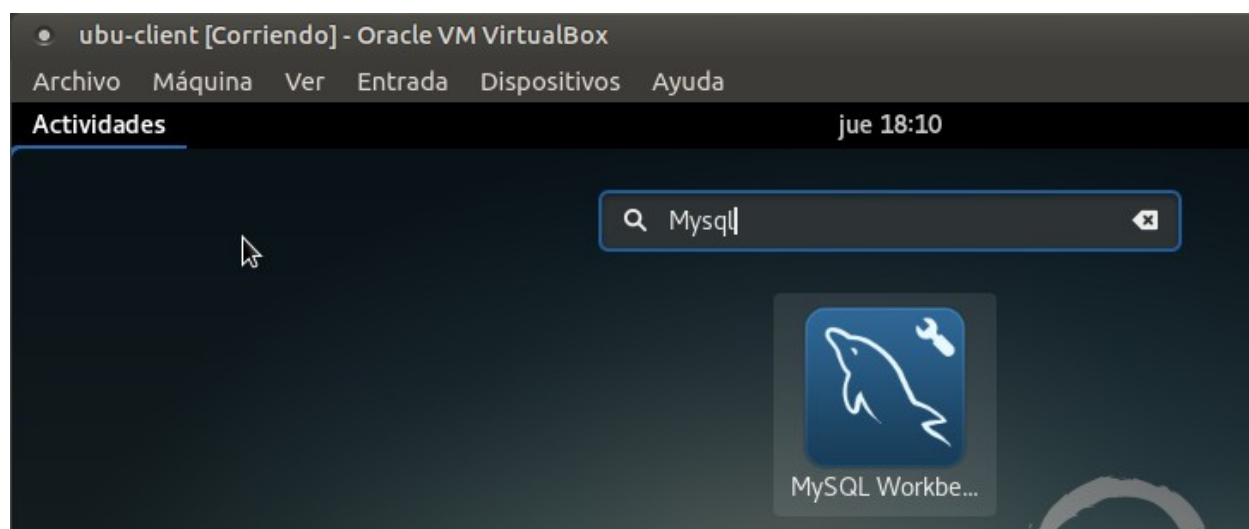


Figura 9: Inicio de sesión de un cliente en la imagen de debian.

Por supuesto, si entramos, encontraremos el servidor MySQL y el MySQL Workbench que habíamos instalado con anterioridad.



*Figura 10: El software específico para la realización del examen debe estar instalado en la imagen.*

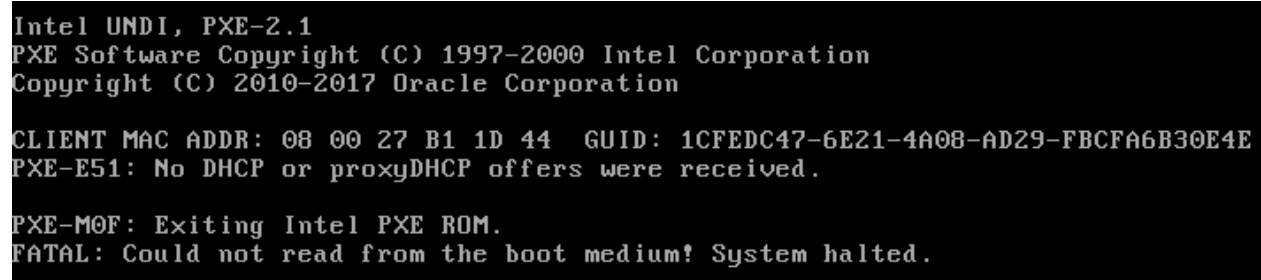
De este modo, hemos podido comprobar que nuestro servicio LTSP funciona y que los clientes que estén conectados al servidor podrán recibir la imagen de la máquina virtual. Para llevar este proyecto a la práctica con máquinas reales, los clientes deberían instalar grub-pxe para tener la opción de “iniciar desde la red”, equivalente a esa opción de LAN que tenemos en las máquinas virtuales. Podemos instalar grub-pxe con `apt install grub-pxe`. Si el cliente cuenta con un SO Windows, puede optar por win32-loader.exe, que puede ser descargado desde la misma página de documentación de LTSP.

## Capítulo 4

### Resolución de problemas y configuración adicional

#### 4.1 DHCP

El primer problema que nos podemos encontrar es que nuestro cliente no reciba una dirección IP de nuestro servidor, y por tanto, no se pueda servir la imagen a dicho cliente.



```
Intel UNDI, PXE-2.1
PXE Software Copyright (C) 1997-2000 Intel Corporation
Copyright (C) 2010-2017 Oracle Corporation

CLIENT MAC ADDR: 08 00 27 B1 1D 44 GUID: 1CFEDC47-6E21-4A08-AD29-FBCFA6B30E4E
PXE-E51: No DHCP or proxyDHCP offers were received.

PXE-M0F: Exiting Intel PXE ROM.
FATAL: Could not read from the boot medium! System halted.
```

*Figura 11: Error de DHCP, el cliente no es capaz de encontrar el servidor y recibir una dirección IP.*

Si hemos utilizado la IP 192.168.67.1/24 como dirección del servidor para dar el servicio a una red interna, en principio no deberíamos encontrarnos como este tipo de problemas. En caso contrario, es probable que un error a la hora de servir direcciones se deba a un fallo de sintaxis, al configurar las interfaces de red, etc. En primer lugar, nos aseguramos de que el servicio de dnsmasq esté funcionando correctamente.

```
/etc/init.d/dnsmasq status
```

Si no se muestran fallos, probablemente sea un error en la configuración del adaptador solo-anfitrión, por lo que deberíamos revisar que esté correctamente. Adicionalmente podemos



habilitar los logs de dhcp para obtener información que nos puede indicar dónde está el problema. Para ello, vamos a `/etc/dnsmasq.d/ltsp-dnsmasq.conf` y descomentamos la línea “log dhcp”. Recordamos que para aplicar los cambios, deberíamos reiniciar el servicio con `/etc/init.d/dnsmasq restart`. Cabe destacar que en el mismo fichero de configuración podemos editar el rango de IPs y el DNS si es necesario.

Por último, debemos mencionar que LTSP es compatible con el `isc-dhcp-server`, por lo que podemos utilizar este servicio alternativamente si lo preferimos y estamos más familiarizados con él.

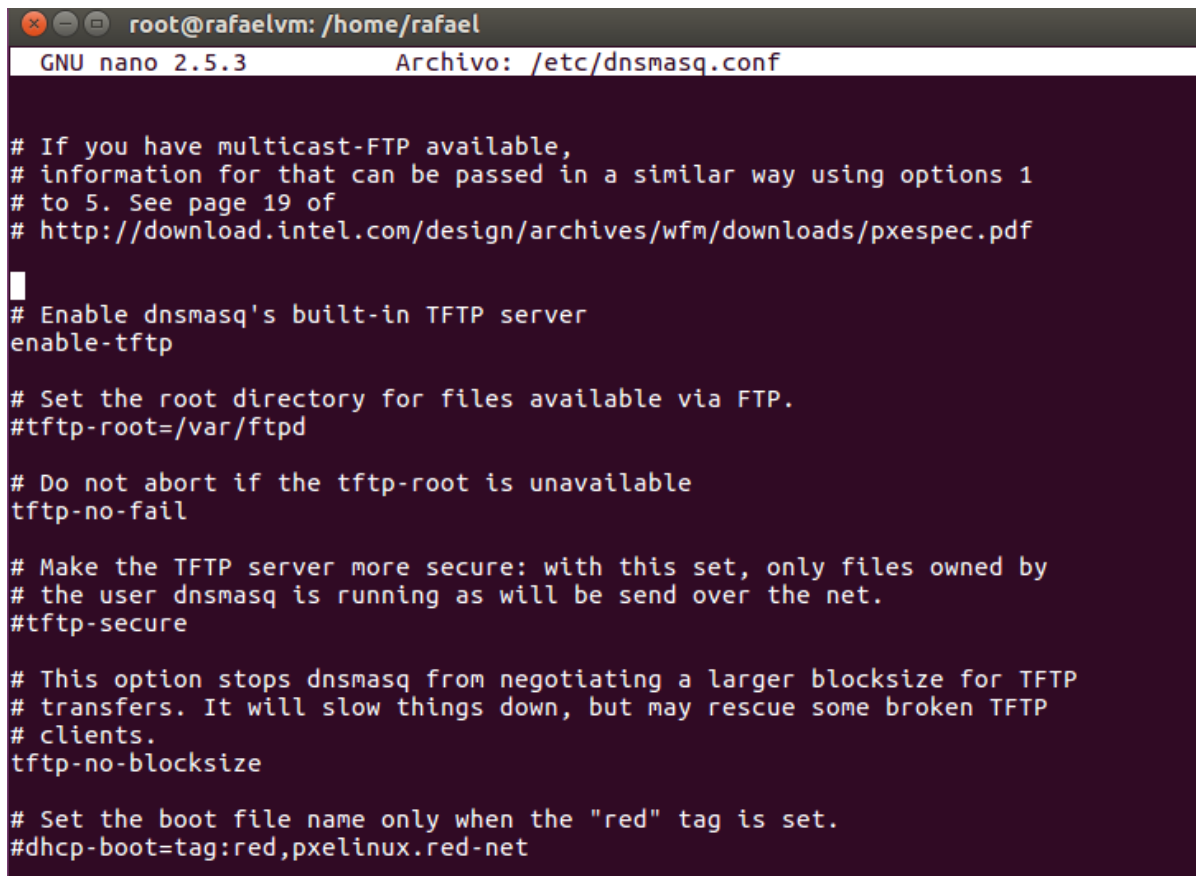
## 4.2 TFTP

Una vez el cliente recibe una dirección IP, el siguiente paso es conectar con el servidor TFTP. Es posible que con la configuración inicial, se produzca un timeout o algún otro error en la transferencia.



Figura 12: Error de TFTP, el servidor no es capaz de servir los archivos al cliente.

En primer lugar, nos dirigimos de nuevo a `/etc/dnsmasq.d/ltsp-dnsmasq.conf`. Nos aseguramos de que la línea “enable-tftp” esté descomentada. Del mismo modo, debemos buscar la misma línea en `/etc/dnsmasq.conf`. Normalmente viene comentada por defecto, por lo que nos aseguramos de que esté activa. En el mismo archivo podemos activar otras opciones, aunque no son necesarias.



```
root@rafaelvm: /home/rafael
GNU nano 2.5.3      Archivo: /etc/dnsmasq.conf

# If you have multicast-FTP available,
# information for that can be passed in a similar way using options 1
# to 5. See page 19 of
# http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf
#
# Enable dnsmasq's built-in TFTP server
enable-tftp

# Set the root directory for files available via FTP.
#tftp-root=/var/ftpd

# Do not abort if the tftp-root is unavailable
tftp-no-fail

# Make the TFTP server more secure: with this set, only files owned by
# the user dnsmasq is running as will be send over the net.
#tftp-secure

# This option stops dnsmasq from negotiating a larger blocksize for TFTP
# transfers. It will slow things down, but may rescue some broken TFTP
# clients.
tftp-no-blocksize

# Set the boot file name only when the "red" tag is set.
#dhcp-boot=tag:red,pxelinux.red-net
```

Figura 13: Archivo de configuración de dnsmasq.

Recordemos que hay que reiniciar el servicio antes de volver a intentar servir la imagen al cliente, con `/etc/init.d/dnsmasq restart`. Si el problema persiste, deberíamos comprobar que nuestro servidor tftp esté escuchando en el puerto `udp/69`

```
netstat -nulp | grep 69
```

Del mismo modo, es recomendable asegurarse de que el firewall permita las conexiones en este puerto. En ubuntu, podemos usar iptables para permitir esta conexión:

```
iptables -A INPUT -p udp -s 192.168.67.0/24 --dport 69 -j ACCEPT
```

Por último, como alternativa al servidor tftp del dnsmasq, podemos utilizar otras opciones, como tftp-hpa, si así lo preferimos. Recordamos que dnsmasq integra varias herramientas necesarias para dar el servicio ltsp, por lo que siempre podremos utilizar herramientas independientes pero que cumplan la misma función.

### 4.3 IPXE

Si el servicio tftp funciona correctamente, a continuación intentaremos acceder al menú iPXE, que nos permitirá iniciar la imagen a través de la red. Sin embargo, es posible que nos encontremos un error tras seleccionar la imagen, como el siguiente:

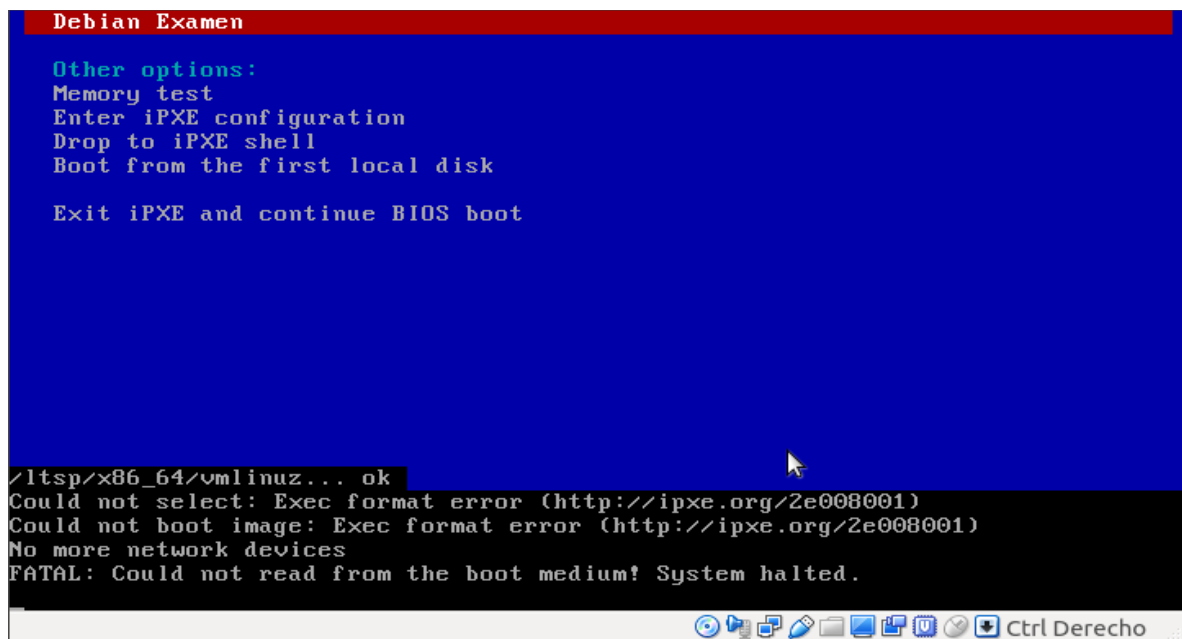


Figura 14: Error de iPXE tras la selección de una imagen.

Los dos errores más comunes que nos encontraremos son el 2e008001 (exec format error) y el 4c1216 (connection timed out). El primero de ellos no debería aparecer si hemos instalado el extension pack de Virtualbox, como recomendamos antes, dado que sin dicho paquete es posible que no se reconozca el formato de la imagen. La página web de ipxe.org (iPXE, 2010) también recomienda asegurarnos de tener instalada la última versión de iPXE:

```
apt-get install gcc binutils make perl liblzma5 mtools mkisofs syslinux
```

```
git clone git://git.ipxe.org/ipxe.git
```

```
cd ipxe/src
```

```
make
```

Por otra parte, el otro error que hemos mencionado normalmente está relacionado con tftp, por lo que no debería ocurrir si hemos configurado este servicio correctamente. Aún así, si el error persiste, podemos ir a /etc/dnsmasq.d/ltsp-dnsmasq.conf y descomentar la línea “tftp-root=/srv/tftp”, lo cual debería ser suficiente para solventar este problema.

## 4.4 Hostname

El siguiente error se puede producir una vez se está iniciando la imagen, interrumpiendo el proceso de arranque:

```

IP-Config: enp0s3 complete (dhcp from 192.168.67.1):
address: 192.168.67.123 broadcast: 192.168.67.255 netmask: 255.255.255.0
gateway: 192.168.67.1 dns0 : 8.8.8.8 dns1 : 208.67.222.222
rootserver: 192.168.67.1 rootpath:
filename :
[ 8.775717] random: fast init done
done.
Begin: Running /scripts/nfs-bottom ... done.
Begin: Running /scripts/init-bottom ... Running /usr/sbin/ltsp initrd-bottom
Running: mount -t tmpfs -o mode=0755 tmpfs /run/initramfs/ltsp
Running: mount -t squashfs -o ro /root/images/x86_64.img /run/initramfs/ltsp/0/looproot
Running: mount -t overlay -o upperdir=/run/initramfs/ltsp/0/up,lowerdir=/run/initramfs/ltsp/0/looproot,workdir=/run/initramfs/ltsp/0/work /run/initramfs/ltsp /root/
done.
hostname: the specified hostname is invalid
LTSP command failed: hostname (none)
Aborting ltsp
LTSP boot error! Enable DEBUG_SHELL to troubleshoot!

```

*Figura 15: Error de hostname, debido a que el cliente tiene un nombre de equipo inválido.*

Este error lo podemos encontrar al intentar poner un nombre que contenga caracteres no permitidos en el hostname de un equipo (en /etc/hostname y /etc/hosts). En nuestro caso, se produce debido a que la máquina cliente no tiene un hostname. Para solucionar esto, en primer lugar, consultamos la dirección MAC de la interfaz de red del cliente. En el caso de una máquina virtual, entramos en la configuración de la máquina, pestaña de red, desplegamos las opciones avanzadas y copiamos la dirección. A continuación, necesitamos crear un archivo de configuración. Simplemente ejecutamos el siguiente comando:

```
install -m 0660 -g sudo /usr/share/ltsp/common/ltsp/ltsp.conf /etc/ltsp/ltsp.conf
```

De este modo, tendremos el archivo de configuración /etc/ltsp/ltsp.conf. Lo editamos con nano y nos vamos a la sección [clients]. Nos encontraremos líneas de configuración comentadas, entre ellas, un ejemplo de como configurar un cliente en particular utilizando la MAC que hemos consultado anteriormente.

```
[clients]
# Specify an /etc/fstab line for NFS home; note
# FSTAB_HOME="server:/home /home nfs defaults,no
# MAC address, IP address, or hostname sections
# to specific clients.
[08:00:27:a6:0b:b8]
HOSTNAME=alumno1
```

*Figura 16: Archivo de configuración de LTSP, sección de clientes.*

Como podemos ver, debemos indicarla entre corchetes y con las letras en minúscula. A continuación, le asignamos un hostname al cliente, como podemos ver en la captura. Es importante no usar comillas, y escribir la palabra hostname en mayúsculas. Guardamos la configuración y reiniciamos el servicio dnsmasq con `/etc/init.d/dnsmasq restart`. Con esto debería quedar solucionado el error del hostname. Si el error persistiera, es posible que debamos actualizar la imagen con `ltsp image nombre_de_la_imagen`.

## **Capítulo 5**

### **Usuarios**

La creación y configuración de usuarios es una parte muy importante de este proyecto, dado que es el principal elemento que nos permitirá que los alumnos estén aislados. Recordamos que tendremos una cantidad de alumnos considerable, entre 20 y 30, por lo que automatizar la creación de alumnos es un añadido valioso en nuestro caso.

#### **5.1 Análisis previo**

En este apartado vamos a analizar y exponer los aspectos fundamentales del funcionamiento los usuarios a la hora de trabajar con LTSP. En primer lugar, la máquina cliente que usa una imagen del servidor puede acceder al sistema de ficheros de dicho servidor. Es decir, si entramos en nuestra imagen debian y utilizamos un usuario del servidor, podemos acceder, por ejemplo, al directorio home de dicho usuario. Esto nos lleva a las siguientes conclusiones:

- Cualquier usuario del servidor LTSP puede ser utilizado en la imagen por los clientes (salvo el administrador).
- Cualquier fichero o directorio al que tenga acceso cierto usuario del servidor puede ser accedido, de igual manera, por el mismo usuario dentro de la imagen.

- Del mismo modo, un fichero o directorio creado en la imagen será accesible desde el servidor, ya sea por el mismo usuario creador o por el administrador del servidor.

Estas características tienen sus ventajas: no hay necesidad de crear usuarios dentro de la máquina virtual base, sino que podemos crearlos en el servidor para posteriormente utilizarlos en la imagen. Se facilita la entrega de exámenes, dado que cada usuario puede, por ejemplo, dejar su fichero sql en el directorio home de su usuario, al cual puede acceder el root del servidor. Y por supuesto, cualquier enunciado o fichero de examen puede ser colocado previamente en los mismos directorios para que los alumnos dispongan de ellos, sin necesidad de conexión a internet, repartir enunciados en folios, etc.

Por otra parte, hay desventajas por las deberíamos tomar precauciones: un alumno que disponga de un usuario y contraseña que no le corresponda podría acceder a los archivos de dicho usuario, por lo que las credenciales deberán ser entregadas al inicio del examen para evitar que los alumnos copien. Del mismo modo debemos borrar esos usuarios una vez hayamos recopilado las respuestas de los exámenes, dado que sería una imprudencia reutilizar estos usuarios para futuros exámenes. Otro punto a tener en cuenta es que no deberíamos tener usuarios con los mismos nombre tanto en la máquina original como en el servidor. Es decir, si nuestro usuario habitual en el servidor es “rafael” y en la máquina virtual creamos otro usuario con el mismo nombre, sería recomendable eliminar el segundo. Es importante porque en un principio, la imagen no muestra los usuarios del servidor, sino los usuarios que tuviera la máquina a la hora de crear la imagen. Por precaución, pues, sería mejor no tener ese mismo usuario “rafael”, dado que en caso de que un alumno lograra acceder con dicho usuario, tendría acceso a nuestros ficheros, etc. Del mismo modo, debemos evitar tener algún usuario en la máquina que sea



predecible, como el típico “usuario usuario”, dado que todos los alumnos podrían acceder a dicho usuario. Si podemos tener una máquina virtual sin más usuario que el administrador, sería lo ideal. Esto debería aplicarse también a los usuarios que creamos para los alumnos. Sería imprudente crear usuarios como “antonio antonio” o “alumno01 contraseña01” que puedan facilitar que algún alumno acceda al perfil de otro usuario que no le corresponde.

## 5.2 Automatización de usuarios

A la hora de crear o borrar usuarios para los exámenes, no es viable crearlos o borrarlos de uno en uno. Dado que necesitamos realizar estas operaciones repetidamente, recurriremos a crear unos scripts en bash. Estos scripts deberán ir acompañados de un fichero extra donde almacenaremos las credenciales de los usuarios, para facilitar su posterior manipulación. Recordemos que a cada script hay que darle permisos de ejecución:

```
chmod u+x script.sh
```

### 5.2.1 Creación de usuarios

La idea del siguiente script es que podamos introducir el número de usuarios que se desean crear y, opcionalmente, el fichero donde queremos plasmar los nuevos usuarios. Dichos usuarios serán creados con un nombre y contraseña totalmente aleatorio, utilizando mayúsculas, minúsculas y números. Las primeras 10 líneas simplemente nos sirven para definir las variables, por lo que no hay que comentar demasiado sobre ellas.

```

#!/bin/bash
defaultfile="usuarios.txt"
read -p "Cantidad de usuarios a crear: " usuarios
read -p "Crear fichero donde almacenar los usuarios (dejar en blanco -> usuarios.txt):" fichero
if [ -z $fichero ]
then
    fichero=$defaultfile
else
    touch $fichero
fi

for a in $(seq 1 $usuarios)
do
    user="$(cat /dev/urandom | tr -dc '[:alnum:]' | head -c 14; echo)"
    password="$(cat /dev/urandom | tr -dc '[:alnum:]' | head -c 14; echo)"
    echo "nombre,$user,contraseña,$password" >> $fichero
    useradd -m -p $(openssl passwd -1 $password) $user
done

```

La segunda mitad del script es la que crea tantos usuarios como hayamos indicado, utilizando un bucle for. Como podemos ver, hacemos uso de /dev/urandom (o /dev/random), que genera números aleatorios, con el comando tr podemos cambiar los números por letras y números, y finalmente, arrojamos los 14 primeros caracteres. Podemos sustituir “alnum” por “alpha” si preferimos sólo letras, “lower” o “upper” para obtener solo minúsculas o mayúsculas, etc. La longitud de ambas cadenas también puede ser modificada cambiando el 14 por la cantidad de caracteres deseada. La siguiente línea guarda las credenciales en el fichero que hayamos especificado, y por último, se hace uso del comando useradd para crear el usuario con contraseña, además de su directorio home. Estos usuarios usan por defecto el shell sh, por lo que si preferimos crearlos con bash, deberíamos ejecutar previamente `sudo useradd -D -s /bin/bash`, o bien añadir al comando useradd “-s /bin/bash” antes de \$user.

A continuación, vamos a probar su funcionamiento.

```

root@rafael-ubuntu:/home/rafael/script_usuarios# cat usuarios.txt
root@rafael-ubuntu:/home/rafael/script_usuarios# ./crear_usuario.sh
Cantidad de usuarios a crear: 5
Crear fichero donde almacenar los usuarios (dejar en blanco -> usuarios.txt):
root@rafael-ubuntu:/home/rafael/script_usuarios# cat usuarios.txt
nombre,OOTSkUfwr4mwzR,contraseña,78YnVIEZfpuFOC
nombre,unQhtxXzAy44y8,contraseña,L97d01NbSi2wzk
nombre,risGwLe62BmRH6,contraseña,wwzqv611Lw665e
nombre,plCVCEng2oPngB,contraseña,9iuUCEQripvNUx
nombre,p8jjMWebAwxAgi,contraseña,hbd1oZJuVL4CTZ
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/
alumnodeejemplo  OOTSkUfwr4mwzR  plCVCEng2oPngB  risGwLe62BmRH6  unQhtxXzAy44y8
franco           p8jjMWebAwxAgi  rafael          suarez          usuario
root@rafael-ubuntu:/home/rafael/script_usuarios# exit
exit
sb3AfQKJC4IzUd@rafael-ubuntu:/home/rafael/script_usuarios$ su risGwLe62BmRH6
Contraseña:
risGwLe62BmRH6@rafael-ubuntu:/home/rafael/script_usuarios$ cat usuarios.txt

```

Figura 17: Creación y muestra de los nuevos usuarios.

Vamos a entrar en la imagen e iniciar sesión con uno de estos usuarios. No olvidemos ejecutar el comando `ltsp initrd` para exportar los nuevos usuarios (de hecho, podemos añadir dicho comando al final del script).

The screenshot shows a terminal window titled "client [Corriendo] - Oracle VM VirtualBox". The terminal output shows a user login for "OOTSkUfwr4mwzR@alumno1: ~". The user runs the command `pwd`, which returns `/home/OOTSkUfwr4mwzR`. Then, the user runs `ls /home/`, which lists the directories: `alumno OOTSkUfwr4mwzR rafael`. Finally, the user runs `ls`, which lists the contents of the home directory: `Descargas Escritorio Música Público Documentos Imágenes Plantillas Vídeos`.

```

client [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Actividades  Terminal  mar 19:22

OOTSkUfwr4mwzR@alumno1: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
OOTSkUfwr4mwzR@alumno1:~$ pwd
/home/OOTSkUfwr4mwzR
OOTSkUfwr4mwzR@alumno1:~$ ls /home/
alumno  OOTSkUfwr4mwzR  rafael
OOTSkUfwr4mwzR@alumno1:~$ ls
Descargas  Escritorio  Música  Público
Documentos  Imágenes  Plantillas  Vídeos
OOTSkUfwr4mwzR@alumno1:~$ █

```

Figura 18: Uso de uno de los nuevos usuarios en el cliente.

Como podemos ver, podemos utilizarlo perfectamente. Dicho usuario sólo tendrá acceso a su propio directorio home. Podrá ver que hay otros, pertenecientes a usuarios de la máquina virtual, pero solo tendrá acceso al suyo propio. Si probamos a crear un fichero en dicho directorio desde el servidor, podremos ver dicho fichero desde la imagen, y viceversa.

### 5.2.2 Reparto del examen

Queremos situar el fichero del examen en cada uno de los directorios de la nueva lista de usuarios, para lo cual, haremos uso de otro script. Este script simplemente leerá la lista y copiará dicho fichero en los directorios home de cada uno de los usuarios que se encuentren en esa lista.

```
#!/bin/bash
defaultfile="usuarios.txt"
read -p "Fichero del examen a repartir: " examen
read -p "Fichero de usuarios a los que se reparte el examen (dejar en blanco -> usuarios.txt):"
fichero
if [ -z $fichero ]
then
    fichero=$defaultfile
fi

while IFS= read -r line; do
    user="$(echo $line |cut -f2 -d,)"
    cp $examen /home/$user
done < $fichero
```

Como podemos ver, pedimos el nombre o la ruta del examen y la lista de usuarios que recibirán ese examen. En este caso usamos un bucle while para leer la lista, y por cada línea que hay en dicha lista, leemos el segundo campo, que es el nombre del usuario. De este modo, copiamos el examen en el directorio home que se llama igual que el nombre del usuario. Vamos a comprobar que funciona.

```

root@rafael-ubuntu:/home/rafael/script_usuarios# cat examen.txt
pregunta 1

pregunta 2
root@rafael-ubuntu:/home/rafael/script_usuarios# cat usuarios.txt
nombre,00TSkUfwr4mwzR,contraseña,78YnVIEZfpuFOC
nombre,unQhtxXzAy44y8,contraseña,L97d01NbSi2wzk
nombre,risGwLe62BmRH6,contraseña,wwzqv611Lw665e
nombre,plCVCEng2oPngB,contraseña,9iuUCEQripvNUx
nombre,p8jjMWebAwxAgi,contraseña,hbd1oZJuVL4CTZ
root@rafael-ubuntu:/home/rafael/script_usuarios# ./repartir_examen.sh
Fichero del examen a repartir: examen.txt
Fichero de usuarios a los que se reparte el examen (dejar en blanco -> usuarios.
txt):
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/
alumnodeejemplo 00TSkUfwr4mwzR  plCVCEng2oPngB  risGwLe62BmRH6  unQhtxXzAy44y8
franco           p8jjMWebAwxAgi  rafael          suarez          usuario
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/plCVCEng2oPngB/
examen.txt
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/unQhtxXzAy44y8/
examen.txt
root@rafael-ubuntu:/home/rafael/script_usuarios# █

```

Figura 19: Reparto del examen y comprobación en los directorios de los usuarios.

Como podemos ver, cada usuario tiene su copia del examen. Vamos a comprobarlo en la imagen.

```

O0TSkUfwr4mwzR@alumno1: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
O0TSkUfwr4mwzR@alumno1:~$ pwd
/home/O0TSkUfwr4mwzR
O0TSkUfwr4mwzR@alumno1:~$ ls /home/
alumno  O0TSkUfwr4mwzR  rafael
O0TSkUfwr4mwzR@alumno1:~$ ls
Descargas  Escritorio  Música  Público
Documentos  Imágenes  Plantillas  Vídeos
O0TSkUfwr4mwzR@alumno1:~$ ls
Descargas  Escritorio  Imágenes  Plantillas  Vídeos
Documentos  examen.txt  Música  Público
O0TSkUfwr4mwzR@alumno1:~$ cat examen.txt
pregunta 1

pregunta 2
O0TSkUfwr4mwzR@alumno1:~$ touch respuesta.txt
O0TSkUfwr4mwzR@alumno1:~$ echo "esta es mi respuesta" >> respuesta.txt
O0TSkUfwr4mwzR@alumno1:~$ ls
Descargas  Escritorio  Imágenes  Plantillas  respuesta.txt
Documentos  examen.txt  Música  Público  Vídeos
O0TSkUfwr4mwzR@alumno1:~$ cat respuesta.txt
esta es mi respuesta
O0TSkUfwr4mwzR@alumno1:~$ █

```

Figura 20: Visualización del examen en el cliente y creación de fichero de respuestas.

De igual manera, vamos a ver que la respuesta del usuario es visible desde el servidor.

```
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/plCVCEng2oPngB/
examen.txt
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/unQhtxXzAy44y8/
examen.txt
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/00TSkUfwr4mwzR/
Descargas  Escritorio  Imágenes  Plantillas  respuesta.txt
Documentos examen.txt  Música    Público     Videos
root@rafael-ubuntu:/home/rafael/script_usuarios# cat /home/00TSkUfwr4mwzR/respuesta.txt
esta es mi respuesta
root@rafael-ubuntu:/home/rafael/script_usuarios#
```

Figura 21: Visualización del fichero de respuestas desde el servidor.

### 5.2.3 Recogida del examen

Este script será similar al anterior. Simplemente vamos a copiar todos los directorios home con su contenido en una carpeta para que el profesor pueda disponer de las respuestas de los alumnos.

```
#!/bin/bash
defaultfolder="respuestas"
defaultfile="usuarios.txt"
read -p "Carpeta donde almacenar los exámenes: " carpeta
read -p "Fichero de usuarios a los que recogemos el examen (dejar en blanco -> usuarios.txt):"
fichero
if [ -z $fichero ]
then
    fichero=$defaultfile
fi
if [ -z $carpeta ]
then
    carpeta=$defaultfolder
else
    mkdir $carpeta
fi

while IFS= read -r line; do
    user="$(echo $line |cut -f2 -d,)"
    cp -r /home/$user/ $carpeta/
```

```
done < $fichero
```

Si ejecutamos este script, este es el resultado:

```
root@rafael-ubuntu:/home/rafael/script_usuarios# ./recoger_examen.sh
Carpeta donde almacenar los exámenes: respuestas_alumnos
Fichero de usuarios a los que recogemos el examen (dejar en blanco -> usuarios.txt):
root@rafael-ubuntu:/home/rafael/script_usuarios# cd respuestas_alumnos/
root@rafael-ubuntu:/home/rafael/script_usuarios/respuestas_alumnos# ls
00TSkUfwr4mwzR  p8jjMWebAwxAgi  plCVCEng2oPngB  risGwLe62BmRH6  unQhtxXzAy44y8
root@rafael-ubuntu:/home/rafael/script_usuarios/respuestas_alumnos# cd 00TSkUfwr4mwzR/
root@rafael-ubuntu:/home/rafael/script_usuarios/respuestas_alumnos/00TSkUfwr4mwzR# ls
Descargas  Escritorio  Imágenes  Plantillas  respuesta.txt
Documentos examen.txt  Música    Público    Videos
root@rafael-ubuntu:/home/rafael/script_usuarios/respuestas_alumnos/00TSkUfwr4mwzR# cat re
spuesta.txt
esta es mi respuesta
root@rafael-ubuntu:/home/rafael/script_usuarios/respuestas_alumnos/00TSkUfwr4mwzR#
```

Figura 22: Recogida del examen mediante copia de los directorios de los usuarios.

Podemos ser más específicos y pedir a los alumnos que nombren su fichero de respuesta con el mismo prefijo, por ejemplo, examen\_mayo\_nombre\_del\_alumno, de manera que podemos recoger el examen usando expresiones regulares, en este caso /home/\$user/examen\_mayo\*.

#### 5.2.4 Borrado de usuarios

Una vez concluya el examen y hayamos recogido las respuestas de los alumnos, como ya hemos dicho, debemos borrar los usuarios para evitar que puedan ser reutilizados en posteriores exámenes. Para ello, usaremos el siguiente script.

```
#!/bin/bash
defaultfile="usuarios.txt"
read -p "Seleccionar fichero con usuarios a borrar (dejar en blanco -> usuarios.txt):" fichero
if [ -z $fichero ]
then
    fichero=$defaultfile
fi
```

```
while IFS= read -r line; do
    user="$(echo $line |cut -f2 -d,)"
    userdel -f $user
    rm -rf /home/$user
done < $fichero
cat /dev/null > $fichero
```

Como podemos ver, utilizamos `userdel` para borrar el usuario, además de eliminar los directorios `home` y el contenido del fichero donde almacenamos los usuarios

```
root@rafael-ubuntu:/home/rafael/script_usuarios# cat usuarios.txt
nombre,00TSkUfwr4mwzR,contraseña,78YnVIEZfpuFOC
nombre,unQhtxXzAy44y8,contraseña,L97d01NbSi2wzk
nombre,risGwLe62BmRH6,contraseña,wwzqv611Lw665e
nombre,pLCVCEng2oPngB,contraseña,9iuUCEQripvNUx
nombre,p8jjMWebAwxAgi,contraseña,hbd1oZJuVL4CTZ
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/
alumnodeejemplo 00TSkUfwr4mwzR pLCVCEng2oPngB risGwLe62BmRH6 unQhtxXzAy44y8
franco           p8jjMWebAwxAgi rafael          suarez          usuario
root@rafael-ubuntu:/home/rafael/script_usuarios# ./borrar_usuarios.sh
Seleccionar fichero con usuarios a borrar (dejar en blanco -> usuarios.txt):
userdel: user 00TSkUfwr4mwzR is currently used by process 8366
userdel: user unQhtxXzAy44y8 is currently used by process 6381
root@rafael-ubuntu:/home/rafael/script_usuarios# ls /home/
alumnodeejemplo franco rafael suarez usuario
root@rafael-ubuntu:/home/rafael/script_usuarios# su 00TSkUfwr4mwzR
No existe clave de entrada para el usuario «00TSkUfwr4mwzR»
root@rafael-ubuntu:/home/rafael/script_usuarios#
```

Figura 23: Borrado de usuarios y de sus directorios.

Aunque nos indiquen que ciertos usuarios están ocupados en algún proceso, se borrarán igualmente, dado que hemos incluido el parámetro “-f” en `userdel` para forzar que se eliminen. Si intentamos hacer log in con alguno de esos usuarios, veremos que ya no existen. Recordemos ejecutar el comando `ltsp initrd` para actualizar los usuarios en la imagen.

### 5.3 Restricción de ficheros



LTSP nos ofrece la posibilidad de restringir el acceso a ciertos ficheros y directorios de manera global para cualquier usuario de la imagen. Para dichos usuarios, los archivos de acceso restringido ni siquiera serán visibles.

En primer lugar, vamos a crear el fichero `/etc/ltsp/image.excludes`. Podemos crear el archivo y empezar a incluir directorios y ficheros en él, pero es recomendable que copiemos el contenido de `/usr/share/ltsp/server/image/image.excludes` y lo peguemos en él.

```
root@rafael-ubuntu:/etc/ltsp# nano image.excludes
root@rafael-ubuntu:/etc/ltsp# ls
image.excludes  ltsp.conf  ltsp-update-image.excludes
root@rafael-ubuntu:/etc/ltsp# cat image.excludes
# This file is part of LTSP, https://ltsp.org
# Copyright 2019 the LTSP team, see AUTHORS
# SPDX-License-Identifier: GPL-3.0-or-later
# To customize, see ADD_IMAGE_EXCLUDES and OMIT_IMAGE_EXCLUDES in ltsp.conf(5)
cdrom/*
cdrom/*
dev/*
etc/epoptes/server.key
etc/.*(java)
etc/mysql/debian.cnf
etc/NetworkManager/system-connections/*
etc/ssh/ssh_host_*
etc/udev/rules.d/??-persistent-*.rules
home/*
lost+found/*
media/*
```

Figura 24: Fichero con lista de directorios que se ocultan a los clientes.

A continuación, debemos mencionar dicho archivo en `/etc/ltsp/ltsp.conf`, incluyendo debajo de la sección `[server]` la siguiente línea:

```
ADD_IMAGE_EXCLUDES="/etc/ltsp/image.excludes"
```

Para que los cambios se efectúen en la imagen, debemos actualizarla con el comando `ltsp image` (seguido del mismo nombre que le pusimos a la imagen). El proceso de actualización puede tardar unos minutos, pero en cualquier caso, tardará mucho menos que la creación inicial

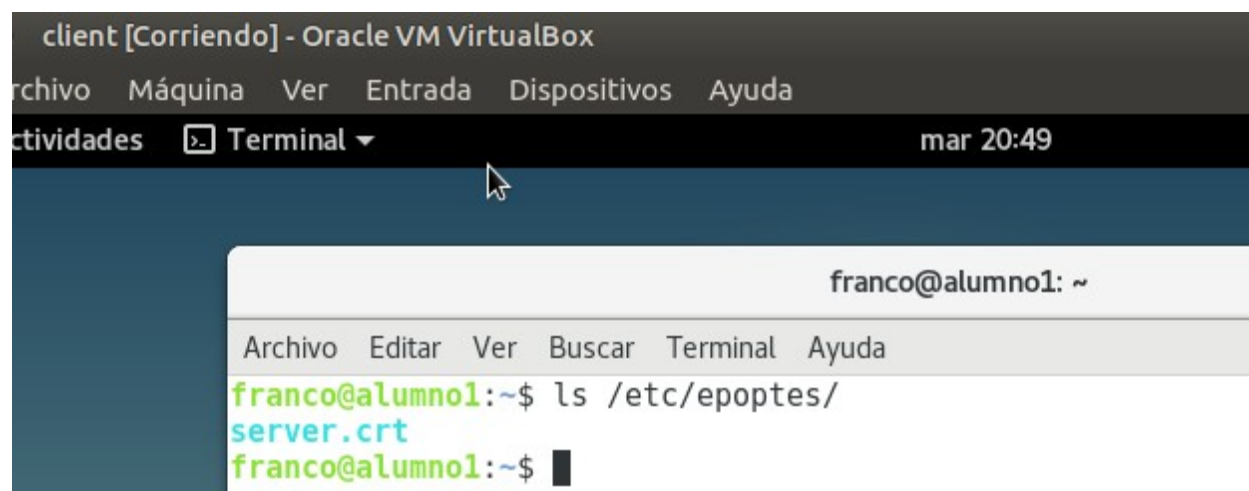
de la imagen. Una vez actualizada la imagen, vamos a intentar acceder desde la imagen a algún directorio de los que aparecen en la captura.

Por ejemplo, en nuestro archivo hemos excluido el fichero `/etc/epoptes/server.key` que se encuentra en nuestro servidor.

```
rafael@rafael-ubuntu:/etc/ltsp$ su franco
Contraseña:
franco@rafael-ubuntu:/etc/ltsp$ ls /etc/epoptes/
server.crt  server.key
franco@rafael-ubuntu:/etc/ltsp$
```

*Figura 25: Comprobación de un archivo de la lista de excluidos, desde el servidor.*

Vamos a comprobar si podemos ver el mismo fichero `server.key` desde la imagen, con el mismo usuario.



*Figura 26: Comprobación de un archivo de la lista de excluidos, desde el cliente.*

Para el usuario es virtualmente inexistente. Por otra parte, podemos ver que hemos incluido `/home/*` en nuestra lista de directorios excluidos. Sin embargo, siempre podremos acceder al home del usuario en cuestión, por lo que esta restricción se aplica a los home de otros usuarios del servidor (no de la máquina). Por esto resaltamos la importancia de que los usuarios

que hubiera en la máquina no deben ser fácilmente accesibles, para que varios alumnos no puedan utilizarlos para comunicarse entre ellos. Como vimos en capturas de apartados anteriores, a parte del usuario utilizado, podíamos ver los directorios home de “rafael” y “alumno”, que son usuarios de la máquina virtual original. Sin embargo, comprobamos que no podemos ver los home del resto de usuarios del servidor, lo cual ocurre gracias a incluir `/home/*` en `image.excludes`.

## 5.4 Autologin de usuarios

LTSP nos ofrece la posibilidad de establecer inicios de sesión automáticos en ciertas máquinas cliente, utilizando un usuario y contraseña que tengamos en el servidor. Sin embargo, un usuario que ha iniciado sesión mediante autologin no tiene acceso, en principio, a su directorio home en el servidor, por lo que debemos buscar una alternativa para el intercambio de exámenes y respuestas, por ejemplo haciendo uso del servidor tftp. Es una alternativa más costosa de poner en marcha, pero en caso de que la encontremos preferible, la podemos aplicar de la siguiente manera.

En primer lugar, necesitaremos conocer la dirección MAC de cada equipo al que queramos asignar un usuario con autologin. Crearemos una sección por cada dirección a la que queremos aplicar este método, pero antes, debemos transformar la contraseña del usuario a base 64.

`base64`

contraseña\_del\_usuario

Presionamos Ctrl+D y copiamos el output.

```
rafael@rafael-ubuntu:~$ base64  
ejemplotriana  
ZWplbXBsb3RyaWFuYQo=
```

Figura 27: Conversión de una cadena a base 64.

A continuación, vamos a /etc/ltsp.ltsp.conf y añadimos la sección con la MAC debajo de la sección [clients]:

```
[clients]  
# Specify an /etc/fstab line for NFS home; note this is insecure  
# FSTAB_HOME="server:/home /home nfs defaults,nolock 0 0"  
  
# MAC address, IP address, or hostname sections can be used to  
# to specific clients.  
[08:00:27:a6:0b:b8]  
HOSTNAME=alumno1  
AUTOLOGIN=alumnodeejemplo  
PASSWORD_ALUMNO1=alumnodeejemplo/ZWplbXBsb3RyaWFuYQo=
```

Figura 28: Fichero de configuración de ltsp, sección de clientes.

Recordamos que la MAC debe ir en minúsculas. Añadimos las líneas:  
AUTOLOGIN=nombre\_de\_usuario

PASSWORD\_NOMBRE\_DEL\_HOST=nombre\_de\_usuario/contraseña\_base64

Guardamos la configuración y ejecutamos `ltsp initrd`. Si intentamos acceder a la imagen con el cliente que tiene esa dirección MAC, entraremos automáticamente con el usuario 'alumnodejemplo'.

Este tipo de configuración puede ser interesante si no queremos entregar las credenciales a los alumnos, pero es mucho más costoso de configurar/automatizar, además de que debemos buscar otro modo de intercambiar archivos con el servidor.

## **Capítulo 6**

### **MySQL**

Tal y como explicamos en el capítulo anterior, los clientes accederán al sistema de ficheros del servidor. Del mismo modo que crear los usuarios en el servidor es más ventajoso y rápido que crearlos en la máquina virtual, implantar y gestionar los servicios de MySQL en el servidor LTSP será la mejor opción. Por lo tanto, las bases de datos que se utilicen en los exámenes estarán centralizadas y controladas en nuestro servidor. Sin embargo, surge la misma problemática: los alumnos no deberían tener acceso a la misma base de datos en el examen, tanto para impedir que se utilice como conducto de comunicación entre clientes, como para garantizar que las operaciones realizadas por algunos alumnos no afecten a otros (updates, inserts, deletes, etc). Así pues, cada alumno debe tener acceso a una copia de la base de datos del examen. Consecuentemente, será necesario crear usuarios con privilegios sobre dichas bases de datos, por lo que vamos a desarrollar esta parte del proyecto siguiendo la misma dinámica del capítulo anterior: automatizando las tareas con scripts. Nos ayudaremos de la documentación de MySQL para desarrollar este capítulo (MySQL Documentation, 2020).

#### **6.1 Instalación del servidor MySQL**

Los clientes ya cuentan con MySQL server y MySQL workbench, por lo que ahora debemos instalar el servicio en el servidor LTSP también. Simplemente ejecutamos:

```
apt-get update
```

```
apt-get install mysql-server
```

Si una vez completada la instalación no se nos pide poner la contraseña de root, podemos ejecutar `mysql_secure_installation utility` para establecerla. También podemos cambiarla entrando al shell de mysql y ejecutando un update:

```
UPDATE mysql.user SET password = 'nueva_contraseña' WHERE user = 'root';
```

```
FLUSH PRIVILEGES;
```

Hay que mencionar que en versiones más recientes no tenemos el campo 'password', sino 'authentication\_string'. Recordamos que para acceder como root, ejecutamos `mysql -u root -p`, aunque no será necesario especificar el usuario si estamos trabajando como root.

Antes de proceder a crear bases de datos, vamos a hacer unas comprobaciones y cambios de configuración para asegurarnos de que no nos encontraremos con ningún error. En primer lugar, verificamos que el puerto 3306 está escuchando:

```
netstat -lnp | grep mysql
```

```
root@rafael-ubuntu:/home/rafael# netstat -lnp | grep mysql
tcp6      0      0 :::3306          :::*              ESCUCHAR      1174/mysqlld
unix  2      [ ACC ]     FLUJO          ESCUCHANDO      33365      1174/mysqlld      /var/run/mysqlld/mysqlld.sock
root@rafael-ubuntu:/home/rafael#
```

*Figura 29: Comprobación del puerto de MySQL.*

Comprobamos que el demonio de mysql está corriendo:

```
ps -Af | grep mysqld
```

```

root@rafael-ubuntu:/home/rafael# ps -Af | grep mysqld
mysql    1174      1  0 11:20 ?        00:00:05 /usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.pid
root     9129   8073  0 12:58 pts/0    00:00:00 grep --color=auto mysqld
root@rafael-ubuntu:/home/rafael#

```

Figura 30: Comprobación del demonio de MySQL.

Por último, editamos el archivo de configuración `/etc/mysql/mysql.conf.d/mysqld.cnf` y comentamos la línea `bind-address = 127.0.0.1`. Esto es necesario para poder acceder desde los clientes. Guardamos los cambios y reiniciamos el servicio con `/etc/init.d/mysql restart`. Con todo esto, no debería haber ningún problema o error a la hora de conectarnos desde un cliente.

## 6.2 Creación de bases de datos y usuarios

Vamos a hacer uso del listado de usuarios que tenemos para facilitar propósito de este apartado, de manera que los usuarios de MySQL serán los mismos que los del servidor. De igual manera, las bases de datos contendrán el nombre de usuario para que los alumnos puedan identificarlas con facilidad. Así pues, vamos a exponer y analizar el script:

```

#!/bin/bash
read -p "prefijo de la base de datos: " bd
read -p "fichero sql a importar: " ficherosql
read -p "usuarios que tendrán una base de datos (usuarios.txt por defecto): " ficherausuario

defaultfile="./usuarios.txt"
if [ -z $ficherausuario ]
then
    ficherausuario=$defaultfile
fi

while IFS= read -r line; do
    user="$(echo $line |cut -f2 -d,)"
    password="$(echo $line |cut -f4 -d,)"

```



```
userdatabase=$bd$user

mysql -e "create database $userdatabase;"
mysql -u root $userdatabase < $ficherosql
mysql -e "CREATE USER $user@"192.168.67.%' IDENTIFIED BY '$password';"
mysql -e "GRANT ALL PRIVILEGES ON $userdatabase.* TO '$user'@"192.168.67.%';"
mysql -e "FLUSH PRIVILEGES;"

done < $ficherosuario
```

El funcionamiento de este código es similar a los que hemos creado anteriormente, con la particularidad de que estamos añadiendo comandos mysql. En primer lugar, pedimos un prefijo para los nombres de las bases de datos, aunque no es necesario. En este caso, vamos a utilizar un fichero sql para importar la base de datos a cada una de las copias. Por último, debemos especificar también ese listado de usuarios, si no estamos utilizando el que hemos dejado por defecto.

A continuación, el bucle while se encargará de leer cada línea de la lista de usuarios y de ejecutar las siguientes tareas en el orden correcto:

- Extraer y definir el usuario y contraseña, además del nombre de la base de datos.
- Crear la base de datos vacía con su nombre.
- Importar el fichero.sql, que volcará las tablas y datos del examen en la base de datos.
- Crear un usuario, con el mismo nombre y contraseña que el usuario del servidor.

Recordamos que el usuario no se va a utilizar de manera local, por lo que detrás del @ debemos escribir '192.168.67.%' en vez de 'localhost' para que un cliente de la red pueda usar dicho usuario.

- Otorgar privilegios al usuario sobre la base de datos. Podemos conceder todos los privilegios o especificar solo los justos y necesarios, si queremos evitar que el alumno

pueda realizar acciones como alterar o borrar una tabla, etc. Por último se ejecuta `flush privileges;` para que estos cambios tengan efecto.

### 6.3 Comprobación y uso de las bases de datos

A continuación, vamos a comprobar el funcionamiento del script con un ejemplo. Dependiendo del tamaño de la base de datos, los procesos de importación pueden tardar más o menos, por lo que es importante tener esto cuenta a la hora de preparar el examen.

```
root@rafael-ubuntu:/home/rafael/script_usuarios# ./crear_usuario.sh
Cantidad de usuarios a crear: 2
Crear fichero donde almacenar los usuarios (dejar en blanco -> usuarios.txt):
root@rafael-ubuntu:/home/rafael/script_usuarios# cat usuarios.txt
nombre,gY6GNd7YdRdyxE,contraseña,1JyAf0tK8d6yNt
nombre,ABHFiynefq1i1p,contraseña,6GfFgkria3uNBq
root@rafael-ubuntu:/home/rafael/script_usuarios# cd mysql/
root@rafael-ubuntu:/home/rafael/script_usuarios/mysql# ./create_database.sh
prefijo de la base de datos: bbdd_de_
fichero sql a importar: examenbbdd.sql
usuarios que tendrán una base de datos (usuarios.txt por defecto):
root@rafael-ubuntu:/home/rafael/script_usuarios/mysql#
```

*Figura 31: Creación de bases de datos y usuarios de MySQL.*

Si todo está correcto, no veremos ningún tipo de mensaje una vez finalice el script. En nuestro caso vamos a crear solamente 2 bases de datos para 2 usuarios. Podemos entrar al shell de mysql para comprobar que se han creado tanto las bases como los usuarios:

```
mysql> select user from mysql.user;
+-----+
| user          |
+-----+
| ABHFiynefq1i1p |
| gY6GNd7YdRdyxE |
| debian-sys-maint |
| mysql.session |
| mysql.sys      |
| root           |
| usuario        |
+-----+
7 rows in set (0.00 sec)
```

Figura 33: Usuarios creados.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| bbdd_de_ABHFiynefq1i1p |
| bbdd_de_gY6GNd7YdRdyxE |
| mysql         |
| performance_schema |
| sys           |
+-----+
6 rows in set (0.00 sec)
```

Figura 32: Bases de datos creadas.

Realizamos algún select, show tables, etc. para comprobar que se han importado los datos correctamente.

```
mysql> use bbdd_de_ABHFiynefq1i1p;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_bbdd_de_ABHFiynefq1i1p |
+-----+
| disc_info          |
| film_info          |
| general_info       |
| image_info         |
| song_info          |
+-----+
5 rows in set (0.01 sec)

mysql> select * from song_info;
+-----+-----+-----+-----+-----+
| title      | duration | artist  | genre  | disc_title |
+-----+-----+-----+-----+-----+
| cancion1   | 10:10:10 | artista1 | genero1 | NULL       |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Figura 34: Comprobación del contenido de las bases de datos.

A continuación, iniciamos sesión en el cliente con uno de los usuarios que acabamos de crear (recordamos que hay que ejecutar `ltsp initrd` para actualizar los usuarios). Para conectarnos desde la terminal, ejecutamos el siguiente comando:

```
mysql -u nombre_usuario -p -h 192.168.67.1
```

En este caso hay que especificar que es un host remoto con el parámetro ‘-h’. En caso de que utilicemos un puerto distinto al 3306, lo podemos indicar con el parámetro -P (por ejemplo “-P 3306”).

```
ABHFiynefqlilp@alumnol:~$ mysql -u ABHFiynefqlilp -p -h 192.168.67.1
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.30-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| bbdd_de_ABHFiynefqlilp |
+-----+
2 rows in set (0.00 sec)
```

*Figura 35: Acceso a las bases de datos del servidor mediante el shell de MySQL.*

Si hacemos un `show databases`, solo veremos la base de datos que le corresponde al usuario, además de `information_schema`, para la cual, el usuario no tiene ningún privilegio, por lo que no debemos preocuparnos. Vamos a comprobar, no obstante, que podemos acceder y realizar cambios en la base de datos.

```

MySQL [(none)]> use bbdd_de_ABHFiynefqlilp;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [bbdd_de_ABHFiynefqlilp]> drop table disc_info;
Query OK, 0 rows affected (0.13 sec)

MySQL [bbdd_de_ABHFiynefqlilp]> show tables
-> ;
+-----+
| Tables_in_bbdd_de_ABHFiynefqlilp |
+-----+
| film_info                          |
| general_info                       |
| image_info                         |
| song_info                          |
+-----+
4 rows in set (0.00 sec)

```

Figura 36: Uso y manipulación de la base de datos desde el cliente.

En este caso, podemos borrar una tabla, por ejemplo, dado que el usuario tiene todos los privilegios sobre esta base de datos. Si podemos acceder y manipular la base por terminal, no habrá ningún problema a la hora de hacer lo mismo desde MySQL workbench.

Por tanto, cada alumno deberá crear una nueva conexión que se parecerá a la siguiente:

Connection Name: examen de gbd

Connection: Remote Management System Profile

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters SSL Advanced

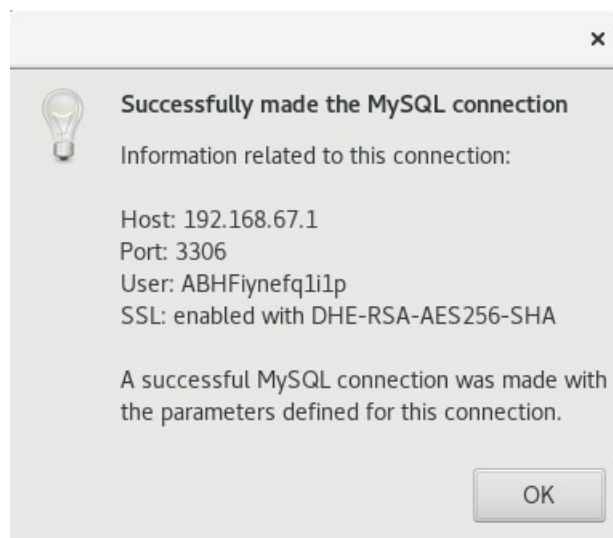
Hostname: 192.168.67.1 Port: 3306 Name or IP address of the server host - and TCP IP port.

Username: ABHFiynefqlilp Name of the user to connect with.

Password: Store in Keychain ... Clear The user's password. Will be requested later if it's not set.

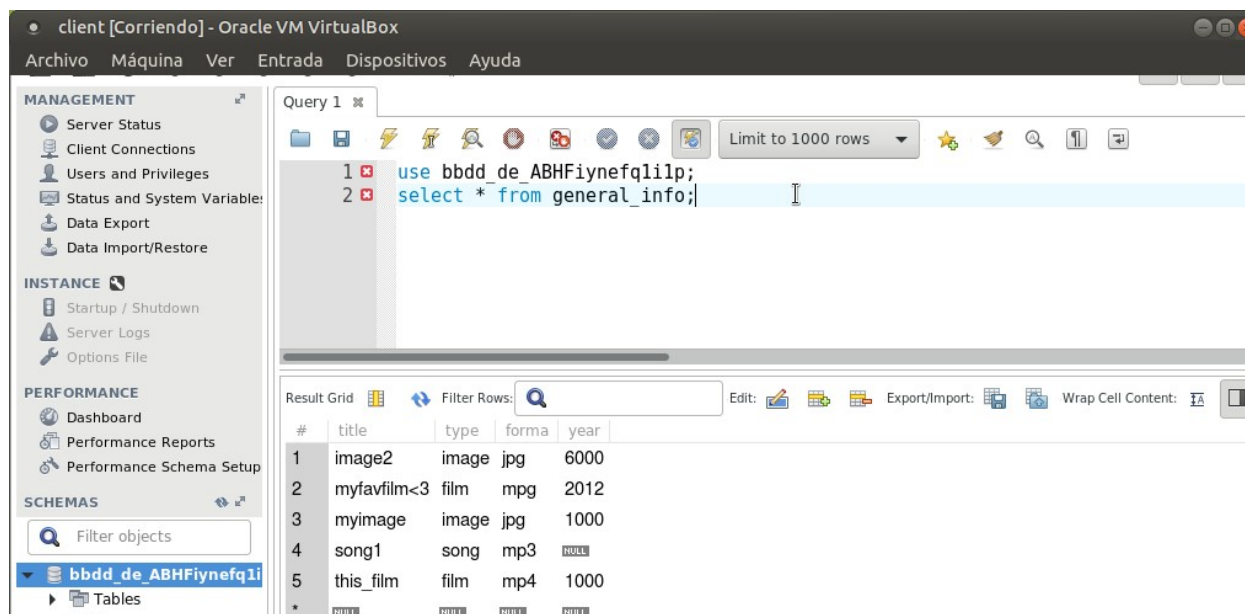
Figura 37: Creación de una nueva conexión en MySQL workbench.

Simplemente debemos introducir las credenciales, la IP del servidor y un nombre. Si hacemos un test de la conexión, debe funcionar correctamente.



*Figura 38: Mensaje de conexión exitosa al servidor MySQL.*

Ahora el alumno podrá abrir la conexión y realizar el examen con la comodidad que brinda el workbench.



*Figura 39: Uso de MySQL workbench desde el cliente, utilizando la base de datos del servidor.*

## 6.4 Borrado de bases de datos y usuarios

Una vez terminado el examen, debemos borrar las bases de datos y sus respectivos usuarios, del mismo modo en el que nos deshacemos de los usuarios del servidor. Simplemente ejecutaremos el siguiente script:

```
#!/bin/bash
read -p "prefijo de las bbdd a borrar: " bd
read -p "usuarios a los que borramos la bbdd (usuarios.txt por defecto): " ficherosusuario

defaultfile="../usuarios.txt"
if [ -z $ficherosusuario ]
then
    ficherosusuario=$defaultfile
fi

while IFS= read -r line; do
    user="$(echo $line |cut -f2 -d,)"
    userdatabase=$bd$user

    mysql -e "drop database $userdatabase;"
    mysql -e "drop user $user@'192.168.67.%';"
done < $ficherosusuario
```

Como podemos ver, es muy similar al anterior, simplemente basta con cambiar los comandos mysql para borrar las bases de datos y los usuarios.

```

root@rafael-ubuntu:/home/rafael/script_usuarios/mysql# ./delete_database.sh
prefijo de las bbdd a borrar: bbdd_de_
usuarios a los que borramos la bbdd (usuarios.txt por defecto):
root@rafael-ubuntu:/home/rafael/script_usuarios/mysql# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 33
Server version: 5.7.30-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database                |
+-----+
| information_schema       |
| mysql                    |
| performance_schema       |
| sys                      |
+-----+
4 rows in set (0.00 sec)

```

*Figura 40: Comprobación de borrado de bases de datos.*

Podemos entrar de nuevo al shell de mysql para comprobar que ya no existen las bases de datos y los usuarios.



## Capítulo 7

### Medidas de seguridad y control

Ya hemos hablado de la importancia de construir este proyecto de manera que se minimice el riesgo de que un alumno pueda hacer trampas en los exámenes, sobre todo en los dos últimos capítulos, con una separación estricta de usuarios y bases de datos. Sin embargo, debemos tomar una serie de medidas adicionales para asegurarnos de que el alumno no pueda copiar, ya sea restringiendo el uso de conexiones a otros equipos, el acceso a internet, etc. Debemos mencionar que nuestros clientes no disponen de herramientas como samba o ftp, dado que la instalación básica de la máquina virtual debían no las incorporaba por defecto. Por tanto, dado que los clientes no pueden adquirir privilegios de administrador para instalar software, no sería necesario hacer nada respecto a ellas. Aún así, algunas medidas que explicaremos a continuación son aplicables a cualquier programa, por lo que no tendremos problemas en limitar su uso, por ejemplo, si utilizamos el método chrootless, en el cual los clientes tendrían acceso a los mismos programas del servidor.

#### 7.1 Ssh y sftp

Los clientes, en un principio, pueden hacer uso de ssh (secure shell) o sftp (ssh file transfer protocol) para conectarse al servidor. Esto no debería suponer un riesgo, siempre y cuando el alumno no cuente con un usuario del servidor, con el que podría acceder a ficheros o recursos que le puedan ayudar a hacer trampas.

```
usuario@alumno1:~$ ssh rafael@192.168.67.1
rafael@192.168.67.1's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 5.3.0-53-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

Pueden actualizarse 113 paquetes.
0 actualizaciones son de seguridad.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Thu May 28 17:57:53 2020 from 192.168.67.20
rafael@rafael-ubuntu:~$ ls
Descargas      ipxe          script_usuarios      'VirtualBox VMs'
Documentos     Música        script_usuarios.zip
Escritorio     Plantillas    Vídeos
Imágenes       Público       virtio-net.rom
```

Figura 41: Uso de ssh desde el cliente para acceder al servidor.

Preventivamente, vamos a realizar un cambio en la configuración de ssh que impedirá el uso de estos servicios. En primer lugar, editamos el fichero /etc/ssh/sshd\_config. Encontraremos la línea ‘ #ChrootDirectory none’, podemos descomentarla o crear otra a parte, de manera que tengamos la línea ‘ChrootDirectory /srv/sshroot’. Si quisiéramos restringir solo ssh pero no sftp,

podemos añadir la línea 'ForceCommand internal-sftp'. A continuación, ejecutamos los siguientes comandos:

```
mkdir -p /srv/sshroot/home
```

```
mount --bind /home /srv/sshroot/home
```

```
/etc/init.d/ssh restart
```

Si intentamos acceder al servidor por ssh, obtendremos el siguiente mensaje:

```
usuario@alumno1:~$ ssh rafael@192.168.67.1
rafael@192.168.67.1's password:
This service allows sftp connections only.
Connection to 192.168.67.1 closed.
```

*Figura 42: Intento de uso de ssh tras configurar la restricción.*

Si no incluimos la línea 'ForceCommand internal-sftp', podemos comprobar que ninguna de las dos conexiones funcionan.

```
usuario@alumno1:~$ ssh rafael@192.168.67.1
rafael@192.168.67.1's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 5.3.0-53-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

Pueden actualizarse 113 paquetes.
0 actualizaciones son de seguridad.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Sat May 30 12:37:35 2020 from 192.168.67.20
/bin/bash: No such file or directory
Connection to 192.168.67.1 closed.
usuario@alumno1:~$ sftp rafael@192.168.67.1
rafael@192.168.67.1's password:
Connection closed
```

*Figura 43: Intento de uso de ssh y sftp.*

En este caso, nos indica que no se encuentra el `/bin/bash`, debido a que no hay ningún binario en el chroot. Por tanto, la conexión se cierra automáticamente. Si queremos ser más específicos y restringir solamente a los usuarios que utilizarán los alumnos, podemos añadir la línea `'Match group nombre_de_grupo'` para que las restricciones se apliquen a los usuarios del grupo que hemos indicado, en cuyo caso deberíamos incluir dicho grupo en el script de creación de usuarios<sup>2</sup>. Por último, hay que mencionar que estos cambios son fácilmente reversibles, en caso de que queramos rehabilitar las conexiones ssh y sftp. Bastaría con eliminar o comentar las líneas que hemos añadido y reiniciar el servicio con `/etc/init.d/ssh restart`.

## 7.2 Restricted shell

En linux podemos hacer uso del shell restringido, `/bin/rbash` (The Restricted Shell - Bash Reference Manual, 2019). Entre sus características, podemos destacar que impide el movimiento entre directorios con el comando `cd`, el uso de comandos que contengan una `'/'`, de manera que no se puede ejecutar un programa si no está en el directorio en el que nos encontramos, o el cambio del valor de variables como `$SHELL` y `$PATH`, entre otras restricciones. El shell restringido puede no ser demasiado útil por sí mismo, pero podemos utilizarlo para crear listas de comandos permitidos (whitelist) para que los alumnos solo puedan ejecutar ciertos comandos,

---

2 Bastaría con crear dicho grupo con `'groupadd nombre_de_grupo'` y añadir `'-G nombre_de_grupo'` antes de `$user` en el comando `useradd`.

por ejemplo, nano, gedit, ls, cat, etc. Consecuentemente, se impediría el uso de otros comandos como ssh, ftp, samba, mail, o cualquier otro que pudiera utilizarse para hacer trampas.

En primer lugar, comprobamos si existe el shell restringido. Si no existe, lo creamos a partir de una copia de bash:

```
cp /bin/bash /bin/rbash
```

Para que los usuarios tengan este shell por defecto, cambiaremos la variable \$SHELL de cada usuario. No es recomendable añadir el shell en el script de creación de alumnos, dado que puede dar problemas a la hora del inicio de sesión del usuario. Más tarde explicaremos cómo cambiar las variables, pero de momento, crearemos un directorio en el que almacenaremos enlaces a aquellos comandos que queramos permitir:

```
mkdir /bin/comandos_permitidos
```

```
ln -s /bin/ls /bin/comandos_permitidos
```

```
ln -s /bin/cat /bin/comandos_permitidos
```

```
ln -s /bin/nano /bin/comandos_permitidos
```

```
ln -s /usr/bin/mysql /bin/comandos_permitidos
```

Es importante recalcar que si esta nueva carpeta está dentro de un directorio excluido (es decir, incluido en /etc/ltsp/image.excludes) es probable que en el cliente no se reconozca dicha carpeta, y por tanto, no se puedan usar los comandos que incluyamos en ella. Una alternativa es crear la carpeta en el directorio home de cada usuario (/home/nombre\_usuario/comandos\_permitidos) la cual no podría ser manipulada por el usuario en cuestión. Es una buena opción dado que el alumno podría ver los comandos que puede utilizar, pero no podría crear enlaces simbólicos o modificar la carpeta.

Podríamos decir que en un principio, el alumno no debería usar comandos de terminal, ya que dispondrá del workbench. Aun así, podemos incluir algunos comandos básicos, además del shell de mysql. Es importante combinar el rbash con esta lista de comandos restringidos, ya que ambas restringen acciones que permiten deshabilitar la otra parte. Por ejemplo, si solo utilizamos el rbash, el alumno podría simplemente ejecutar 'bash' y así tendría acceso al shell normal. Hemos mencionado que '/bin/bash' no puede ser ejecutado en el shell restringido, sin embargo, no se prohíbe la ejecución de 'bash' dado que no contiene ninguna barra o carácter restringido. Sin embargo, al hacer uso de esta lista de comandos permitidos, el enlace simbólico de 'bash' no es reconocido, y por tanto, el alumno no puede cambiar de shell. Por otra parte, si utilizáramos únicamente la lista de comandos, el alumno podría cambiar la variable \$PATH e incluir /bin, /usr/bin, etc, por lo que potencialmente tiene acceso a todos los comandos, incluyendo ssh, ftp, etc. No obstante, si está utilizando el rbash, no se permite la escritura o modificación de variables como \$PATH y \$SHELL, entre otras. De este modo, el alumno no podría cambiar esa lista de comandos y acceder al resto de binarios. En definitiva, estas dos medidas deben tomarse en conjunto si queremos lograr y garantizar que el alumno solo pueda ejecutar un número limitado de comandos de terminal.

Aún debemos cambiar las variables \$SHELL y \$PATH de cada alumno. Como hemos indicado, no vamos a especificar el shell en el propio proceso de creación de usuarios. En nuestro caso, editaremos o añadiremos el fichero .bashrc en el directorio de cada usuario, el cual modificará el valor de estas variables tras el inicio de sesión. De esta manera, prevenimos cualquier error a la hora de hacer login si el shell predefinido es rbash. Así pues, creamos o modificamos el fichero .bashrc y añadimos la variable \$PATH y \$SHELL de manera que sus

valores sean, respectivamente, `‘/home/nombre_de_usuario/comandos_permitidos’`<sup>3</sup> y `‘/bin/rbash’`. El contenido del fichero debería ser el siguiente:

```
PATH=/home/nombre_de_usuario/comandos_permitidos
export PATH
SHELL=/bin/rbash
export SHELL
```

En nuestro caso, los usuarios se crean sin `.bashrc` por defecto, por lo que tendremos que crear el fichero en el script.

```
#script de creación de usuarios.
useradd -m -p $(openssl passwd -1 $password) $user

#copiamos la carpeta de comandos en el home de cada usuario
cp /bin/comandos_permitidos /home/$user/comandos_permitidos

fichero_prof="/home/$user/.bashrc"
touch $fichero_prof

echo "PATH=/home/$user/comandos_permitidos" >> $fichero_prof
echo "export PATH" >> $fichero_prof
echo "SHELL=/bin/rbash" >> $fichero_prof
echo "export SHELL" >> $fichero_prof

done
```

Recordamos que la creación de usuarios la hacemos como root, por tanto, no hay riesgo de que el alumno pueda cambiar el contenido de este fichero o de la copia de la carpeta de comandos permitidos, dado que el propietario es el administrador, no el usuario del home en el

---

3 Nota aclaratoria: en las capturas tomadas a continuación, se utiliza la carpeta `‘/bin/comandos_permitidos’`, dado que estamos comprobando su funcionamiento en el servidor. Como ya hemos dicho, es recomendable tener esa carpeta en el home de cada usuario, dado que así nos aseguramos de que la variable `$PATH` pueda encontrarla. En definitiva, no es recomendable usar una carpeta única del servidor para todos los clientes.

que se encuentran estos archivos. Vamos a comprobar que un usuario del servidor solo puede utilizar los comandos que hemos especificado.

```
rafael@rafael-ubuntu:/home/restricted_user$ su restricted_user
Contraseña:
restricted_user@rafael-ubuntu:~$ echo $SHELL
/bin/rbash
restricted_user@rafael-ubuntu:~$ echo $PATH
/bin/comandos_permitidos
restricted_user@rafael-ubuntu:~$ PATH=/bin
rbash: PATH: variable de sólo lectura
restricted_user@rafael-ubuntu:~$ SHELL=/bin/bash
rbash: SHELL: variable de sólo lectura
restricted_user@rafael-ubuntu:~$ /bin/bash
rbash: /bin/bash: restringido: no se puede especificar «/» en nombres de órdenes
restricted_user@rafael-ubuntu:~$ bash
rbash: /usr/lib/command-not-found: restringido: no se puede especificar «/» en nombres de órdenes
```

*Figura 44: Comprobación de la imposibilidad del usuario de cambiar de shell.*

Como podemos ver, no es posible cambiar las variables o utilizar un shell distinto. En cambio, podemos usar todos aquellos comandos que enlacemos en el directorio ‘/bin/comandos\_permitidos’ (o ‘/home/nombre\_de\_usuario/comandos\_permitidos’):

```
restricted_user@rafael-ubuntu:~$ nano respuesta
restricted_user@rafael-ubuntu:~$ cat respuesta
respuestas
restricted_user@rafael-ubuntu:~$ ls
respuesta
restricted_user@rafael-ubuntu:~$ mysql -u usuario -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.7.30-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

*Figura 45: Uso de comandos permitidos.*



```
restricted_user@rafael-ubuntu:~$ ssh
rbash: /usr/lib/command-not-found: restringido: no se puede especificar «/» en nombres de órdenes
restricted_user@rafael-ubuntu:~$ ftp
rbash: /usr/lib/command-not-found: restringido: no se puede especificar «/» en nombres de órdenes
restricted_user@rafael-ubuntu:~$ tftp
rbash: /usr/lib/command-not-found: restringido: no se puede especificar «/» en nombres de órdenes
restricted_user@rafael-ubuntu:~$ mail
rbash: /usr/lib/command-not-found: restringido: no se puede especificar «/» en nombres de órdenes
restricted_user@rafael-ubuntu:~$ su
```

*Figura 46: Intento de uso de comandos no permitidos.*

Vamos a probar ahora desde una máquina cliente. Recordemos que debemos cambiar ‘/bin/comandos\_permitidos’ por ‘/bin/restricted\_user/comandos\_permitidos’ para lograr que funcione (además de la variable PATH en su .bashrc).

```
restricted_user@alumnol:~$ echo $SHELL
/bin/rbash
restricted_user@alumnol:~$ echo $PATH
/home/restricted_user/comandos_permitidos
restricted_user@alumnol:~$ ls
comandos_permitidos  Documentos  Imágenes  Plantill
Descargas            Escritorio  Música    Público
restricted_user@alumnol:~$ ls comandos_permitidos/
cat echo login ls mysql nano su
restricted_user@alumnol:~$ cat respuesta
respuestas
restricted_user@alumnol:~$ gedit
rbash: gedit: no se encontró la orden
restricted_user@alumnol:~$ ssh
rbash: ssh: no se encontró la orden
restricted_user@alumnol:~$ mail
rbash: mail: no se encontró la orden
restricted_user@alumnol:~$ █
```

*Figura 47: Comprobación de comandos del usuario desde el cliente.*

No olvidemos el propósito de este capítulo. Esta medida es muy contundente y puede eliminar gran parte de las posibilidades de un alumno de hacer trampas. Por otra parte, debemos procurar que estas medidas no lastren a los alumnos en la realización del examen, por lo que es

recomendable comprobar e incluso hacer un pequeño simulacro para verificar que no hemos despojado de ninguna herramienta necesaria al alumno.

### 7.3 Permisos de linux

Los permisos de linux pueden servir para denegar el acceso a ciertos programas. Este método es el opuesto al anterior, dado que en este caso, estaríamos creando una lista negra de comandos o programas, en lugar de una lista blanca de comandos permitidos. Por tanto, no es una medida tan intrusiva, pero por el contrario, si hay una gran cantidad de software al que tenemos que limitar el acceso, requerirá de mucho más tiempo y esfuerzo.

La idea de este apartado es simple: revocar permisos de ejecución en todos aquellos programas, binarios, etc. a los que no deben acceder los alumnos. Como sabemos, se le pueden otorgar distintos permisos al propietario, al grupo y al resto de usuarios sobre un determinado archivo, binario, etc. Los programas suelen ser propiedad del usuario root y del grupo root, dado que es el administrador el que suele instalar software en el sistema. Por tanto, en la mayoría de casos, nos bastaría con eliminar los privilegios de ejecución del “resto de usuarios”, dado que por norma general, el usuario root es el único miembro del grupo root. En caso de que necesitemos consultar los usuarios que pertenecen a un grupo, podemos hacerlo con el comando `getent group nombre_de_grupo`. No sería necesario cambiar el grupo propietario de un binario, dado que los usuarios que creamos para los alumnos no son añadidos a ningún grupo que hubiera previamente

en el sistema (cada usuario pertenece por defecto a un grupo que tiene el mismo nombre de ese usuario), pero en caso de que queramos cambiar el grupo o el propietario, por ejemplo, si queremos tener un grupo de usuarios del sistema que tenga acceso al archivo en cuestión, podemos usar el comando `chown nuevo_propietario:nuevo_grupo ruta_del_archivo`.

En nuestro caso particular, debemos aplicar los permisos en la máquina virtual, no en el servidor, dado que la imagen que utilizan los clientes hace uso del software de dicha máquina (por eso debíamos instalar el workbench en la máquina debian y no en el servidor). El procedimiento a seguir es el siguiente:

- En primer lugar, localizamos los programas y binarios que queremos limitar. Para ello, podemos hacer uso del comando `which`, que nos arrojará la ruta absoluta del binario indiquemos.

`which nombre_del_programa`

- A continuación, consultamos los permisos del programa, además del usuario y grupo propietario del mismo. Como hemos dicho, por norma general simplemente quitaremos permiso de ejecución al “resto de usuarios”, pero es recomendable examinar cada uno de los programas, en caso de que alguno requiera de un tratamiento distinto (cambiar grupo propietario, quitar permisos de escritura o lectura en caso de querer restringir el acceso a un fichero, etc.). Podemos utilizar el siguiente comando:

`ls -l ruta_del_programa`

- Suprimimos el permiso de ejecución del conjunto del resto de usuarios, identificado por la ‘o’ de ‘others’.

`chmod o-x ruta_del_programa`

Repetiremos este proceso con todos aquellos programas, carpetas, ficheros, etc. que queramos. Si probamos con un usuario no propietario, comprobaremos que no podemos acceder al programa.

```
rafael@debianrafael: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@debianrafael:/home/rafael# which firefox  
/usr/bin/firefox  
root@debianrafael:/home/rafael# ls -l /usr/bin/firefox  
-rwxr-xr-x 1 root root 113 abr  8 00:54 /usr/bin/firefox  
root@debianrafael:/home/rafael# chmod o-x /usr/bin/firefox  
root@debianrafael:/home/rafael# ls -l /usr/bin/firefox  
-rwxr-xr-- 1 root root 113 abr  8 00:54 /usr/bin/firefox  
root@debianrafael:/home/rafael# exit  
exit  
rafael@debianrafael:~$ firefox  
bash: /usr/bin/firefox: Permiso denegado  
rafael@debianrafael:~$ su
```

Figura 48: Cambio de permisos de la aplicación de firefox.

Una vez hayamos cambiado los permisos necesarios, debemos actualizar la imagen.

`ltsp image ruta_de_la_imagen`

Vamos a comprobar ahora que tampoco podemos acceder desde un cliente a firefox ni a otros binarios a los que hemos restringido el acceso.

```
franco@alumno1: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
franco@alumno1:~$ firefox  
bash: /usr/bin/firefox: Permiso denegado  
franco@alumno1:~$ firefox-esr  
bash: /usr/bin/firefox-esr: Permiso denegado  
franco@alumno1:~$ ssh  
bash: /usr/bin/ssh: Permiso denegado  
franco@alumno1:~$ passwd  
bash: /usr/bin/passwd: Permiso denegado
```

Figura 49: Comprobando los nuevos permisos desde el cliente.

De hecho, ni siquiera podremos encontrar el icono de firefox en la lista gráfica de aplicaciones.



*Figura 50: Comprobación de la imposibilidad de encontrar la aplicación de firefox.*

Como podemos ver, es un método más sencillo de aplicar que el anterior. Puede que requiera de más tiempo, pero dado que la máquina virtual solo cuenta con mysql y mysql-workbench a parte la instalación inicial, es una opción perfectamente viable. Recordamos que el proceso de actualización de la imagen tarda un poco, por lo que se recomienda identificar todos los programas a los que queramos limitar el acceso y asegurarnos de que hemos suprimido todos los permisos necesarios antes de proceder a actualizar.

## 7.4 Epopetes

Hicimos una breve mención sobre epoptes (Epopetes Documentation, 2010) en el primer capítulo de este proyecto. Esta herramienta es muy útil en entornos de aulas con ordenadores, dado que permite al profesor monitorizar, controlar, bloquear o incluso apagar los equipos de los alumnos, entre otras opciones. Ya instalamos en su momento el programa junto con el resto de herramientas de ltsp, y añadimos el usuario administrador al grupo de epoptes. Sólomente con esto, ya podemos lanzar la aplicación, la cual detectará cualquier cliente del servidor LTSP que haya en la red.

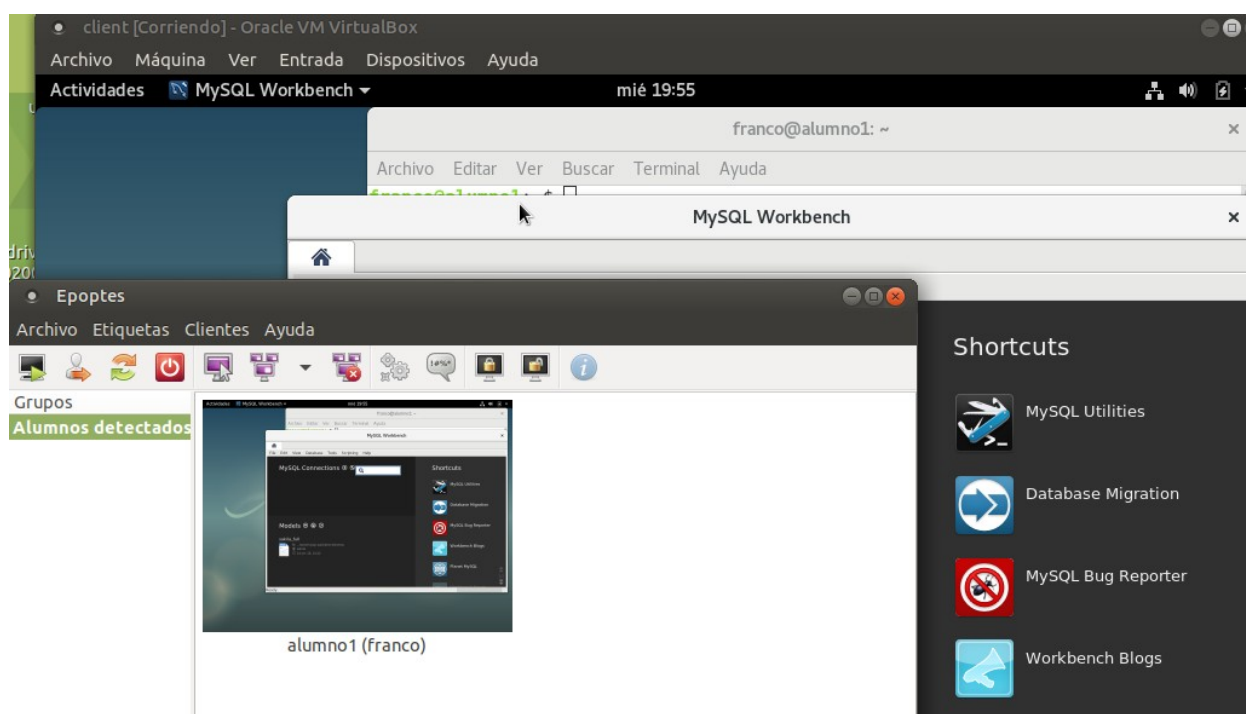


Figura 51: La aplicación gráfica de epoptes muestra la actividad del cliente en tiempo real.

Como podemos observar, el profesor puede monitorizar todos los clientes en tiempo real, por lo que si el alumno realiza alguna acción sospechosa, su equipo puede ser intervenido o

bloqueado en el momento. También nos ofrece opciones interesantes, como enviar un mensaje a los clientes o hacer una evaluación de la red. Puede ser útil para garantizar la igualdad de condiciones en el examen, por ejemplo, bloqueando las pantallas en cuanto termina el tiempo del examen. En resumen, es un programa muy intuitivo, útil y fácil de utilizar.

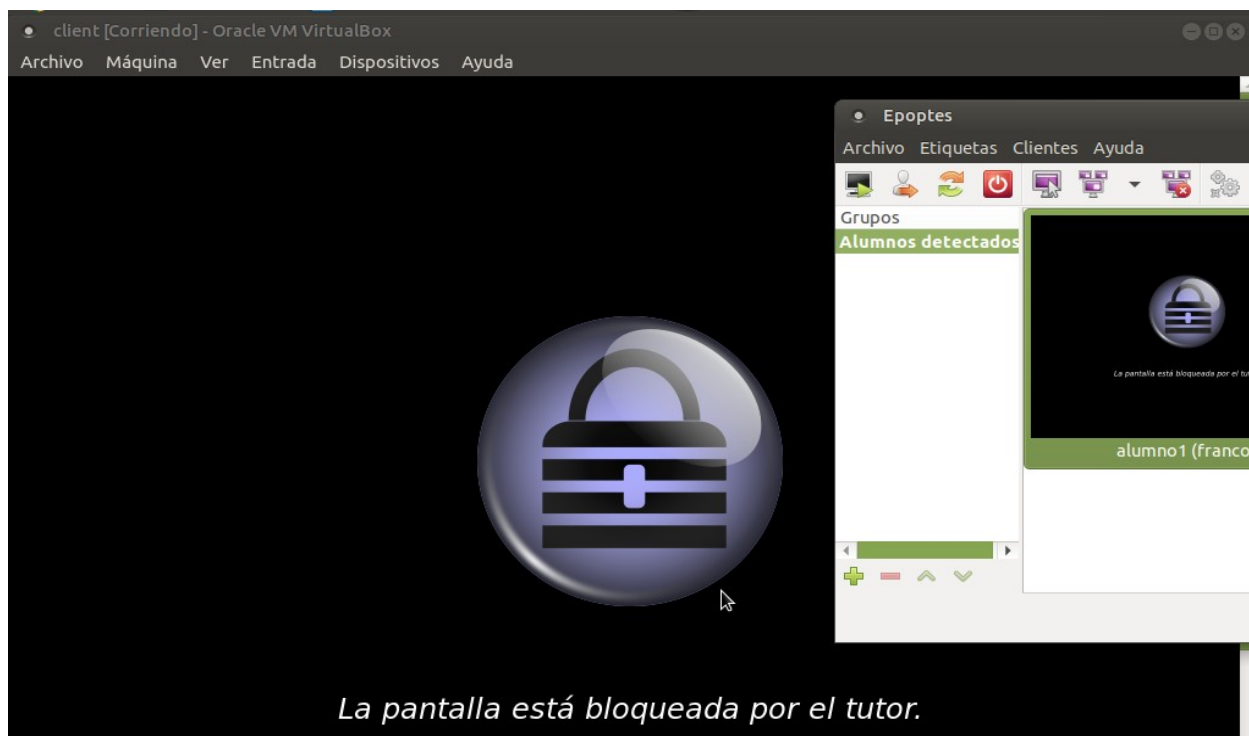


Figura 52: Pantalla del cliente bloqueada por el profesor desde eprotes.

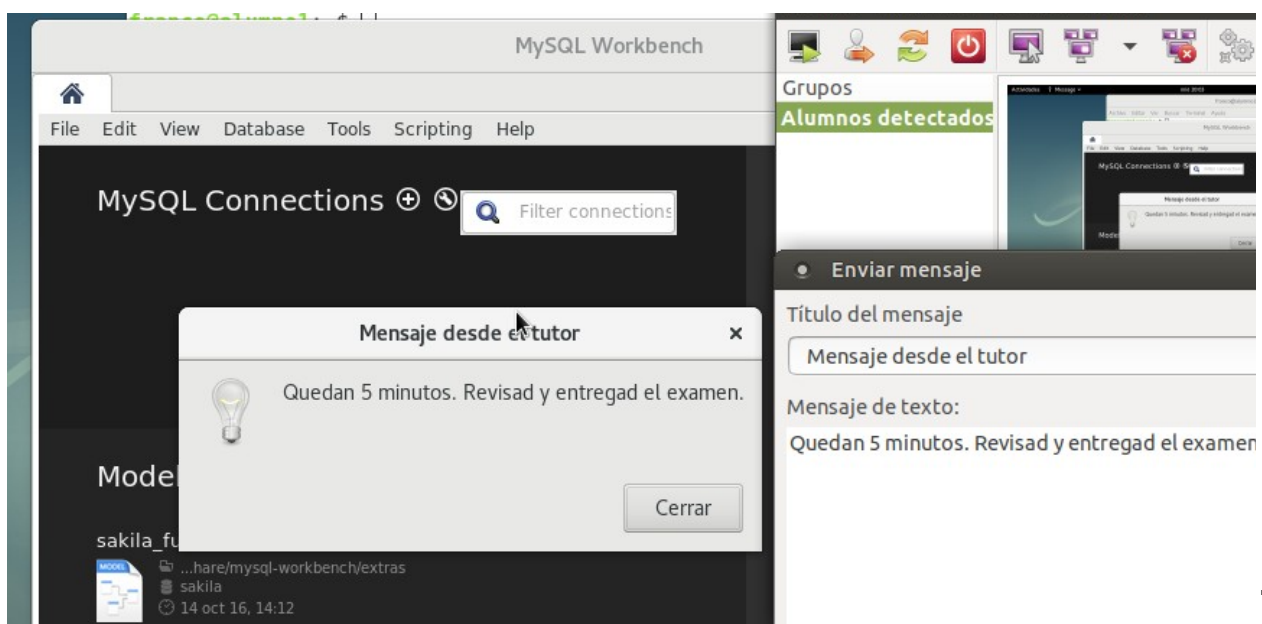


Figura 53: Mensaje del profesor enviado al cliente.

## Conclusiones

Como hemos visto a lo largo de este proyecto, se puede crear un entorno para exámenes de informática con software y equipos al alcance de cualquier centro de enseñanza, sin necesidad de hacer un desembolso importante ni de tener unos conocimientos muy específicos para ello. La solución que hemos propuesto y desarrollado pasa por centralizar las tareas y recursos en el servidor de LTSP, por lo que es recomendable que dicho servidor, en la práctica, cuente con unas especificaciones que puedan soportar la carga de trabajo estimada (como ya hemos dicho, entre 20 y 30 clientes). Este entorno que hemos desarrollado se caracteriza por tener un proceso de instalación y configuración bastante largo, pero una vez terminado, su utilización es bastante sencilla, sobre todo con la ayuda de los scripts. Por tanto, es más lógico instalar un servidor LTSP si vamos a hacer una cantidad de exámenes considerable, ya sea en uno o varios años, o incluso en distintas asignaturas, para lo cual podemos preparar distintas imágenes. Esta solución también garantiza la igualdad de condiciones entre los alumnos. No importa el sistema operativo que usen normalmente, o si tienen instalado o no el software necesario para el examen, dado que en el examen todos usarán el mismo sistema, software, recursos, etc. sin dar opción a que un alumno pueda intentar hacer trampas. En definitiva, hemos cumplido los objetivos propuestos al principio de este proyecto, dado que nuestro entorno para exámenes de informática es adaptable a distintas asignaturas, asequible, fácil de entender y de aplicar, y seguro frente a cualquier estratagema que puedan intentar los alumnos para obtener un resultado mayor.



## Lista de referencias

- LTSP Documentation. (2000). Recuperado de <https://ltsp.org/docs/>
- LTSP github community. (2019). Recuperado de <https://github.com/ltsp/community/wiki>
- PassMark Software - CPU Benchmarks. (2020). Recuperado de <https://www.cpubenchmark.net/>
- Team., D. (2020). DRBL - About. Recuperado de <https://drbl.org/>
- Cupp Jr., D. A. (2017). ThinStation. Recuperado de <http://thinstation.org/>
- NoMachine S.à r.l. (2002). NoMachine - Servidor de Terminal Linux del Creador de NX. Recuperado de <https://www.nomachine.com/es/terminal-server>
- Oracle Corporation. (2010). Oracle VM VirtualBox. Recuperado de <https://www.virtualbox.org/>
- SPI (Software in the Public Interest). (1997). Debian -- Getting Debian. Recuperado de <https://www.debian.org/distrib/>
- iPXE - open source boot firmware [download]. (2010). Recuperado de <https://ipxe.org/download>
- MySQL :: MySQL Documentation. (2020). Recuperado de <https://dev.mysql.com/doc/>
- The Restricted Shell (Bash Reference Manual). (2019). Recuperado de [http://www.gnu.org/software/bash/manual/html\\_node/The-Restricted-Shell.html](http://www.gnu.org/software/bash/manual/html_node/The-Restricted-Shell.html)
- How can I restrict the normal user to run only limited set of commands? (2019). Recuperado de <https://access.redhat.com/solutions/65822>
- Eptotes Documentation. (2010). Recuperado de <https://eptotes.org/documentation/>