

ARRANQUE REMOTO Y CLONACIÓN DE EQUIPOS CLIENTE – v45 – Noviembre 2021

PASOS INICIALES

Instaláis o actualizáis el virtualizador VirtualBox (www.virtualbox.org) y su “Extension pack” acorde a la versión (simplemente haciendo doble click sobre el archivo descargado). IMPORTANTE: En Windows la instalación suele ser más sencilla; en Linux y MAC recomendamos comprobar que la configuración de red funciona correctamente (seguid las instrucciones con detenimiento y probad conectividad, al menos, mediante NAT).

En esta práctica montaremos tres equipos sobre VirtualBox:

- Un servidor DRBL en Debian 32bits (**CZmiDebianserver**)
- Un cliente de clonación Debian 32bits (**CZmiDebian**)
- Un cliente de clonación Windows **10** con AIK (**CZmiWin**)

Puedes descargar Debian aquí (selecciona la “*imagen de instalación pequeña*” y luego arquitectura i386). En esta ocasión, utilizaremos **Debian 10.11** (En Debian 11 Clonezilla aún no funciona correctamente):

<https://cdimage.debian.org/cdimage/archive/10.11.0/i386/iso-cd/debian-10.11.0-i386-netinst.iso>

IMPORTANTE: A partir de la versión 8 de Debian, se utiliza **systemd** en lugar de **init.d** para levantar las interfaces, que además han cambiado su nombre desde Debian 9 (ya no son *ethX*). Recordad que ahora el comando para ver las interfaces es “*ip a*” (aunque hay un truco para que funcione “*ifconfig*” 😊).

Estas son las características hardware (virtual) de dichas máquinas (todas de 32 bits):

- **CZmiWin** -> 1024MB de RAM y Disco 32GB
- **CZmiDebian** -> 1024 MB de RAM Disco 8GB
- **CZmiDebianserver** -> 1024MB de RAM y Disco 8GB + opcionalmente un Disco de 8GB (depende del número de imágenes que queráis conservar en el servidor).

En cuanto a la configuración de red, en todos los casos definimos dos adaptadores:

- Adaptador 1 [enp0s3 en linux] -> Red interna (realizaremos las clonaciones). Ha de ser el primero porque PXE sólo funciona por el primer adaptador.
- Adaptador2 [enp0s8 en linux] -> NAT (sólo para acceso a Internet)

Os proponemos instalar la máquina Debian CZmiDebian, luego veremos la forma de clonarla para que su copia sea luego nuestro servidor de clonación. Instalamos cada una con su ISO correspondiente. **Todos los usuarios y claves en Linux y Windows serán “cz”, también el root, para evitar problemas de acceso a las máquinas en las prácticas (pensad que os las copiaréis de unos a otros).** Algunos pasos:

- Windows 10 (CZmiWin):
 - Esta máquina está preinstalada, disponéis de una imagen .OVA importable desde VirtualBox (<https://lia2.udc.es/xi/CZmiWin10.ova>)
- Debian Linux (CZmiDebian):
 - Con la versión Debian 32bits (CD pequeño, net install), no seleccionar el entorno de escritorio durante la instalación inicial (así la máquina a clonar será más pequeña). Más adelante indicamos como instalar, por ejemplo, un entorno X Gnome básico en el servidor.
 - Como interface de red para la instalación elegimos enp0s8 (la de NAT, acceso a Internet)
 - Durante la instalación indicamos el nombre “CZmiDebian” y el dominio “dominioCZ.es”.
 - En cuanto a las particiones, dejamos todo en una única partición (por simplificar y aprovechar el espacio en vuestros equipos).
 - Seleccionamos la réplica (tenemos una en la UDC, ftp.udc.es, normalmente funciona bien).

- En cuanto a los paquetes, seleccionamos únicamente: “Servidor SSH” y “Utilidades estándar del sistema” (eliminad la opción de “Entorno de Escritorio”).
- Al terminar la instalación, nos preguntará si queremos escribir el registro de arranque. Obviamente, le decimos que sí, en /dev/sda y reinicia.
- Desde la versión 8, existen bastantes problemas con la configuración de red, especialmente con múltiples interfaces. La clásica configuración en /etc/network/interfaces no funciona bien en algunas situaciones. Por ello, configuramos los interfaces mediante systemd en dos archivos en /etc/systemd/network/ con nombres enp0s3.network y enp0s8.network (ejecutando “ip a” podemos verlos) introduciendo, en el caso de enp0s3:

```
[Match]
Name=enp0s3

[Network]
DHCP=v4
```

... y en el de enp0s8 sólo cambiaría el nombre, enp0s3, por enp0s8. Podéis utilizar **pico**, **nano** o **vi** para editar los archivos, el que más os guste.

Para que systemd se encargue de gestionar los interfaces, ejecutamos lo siguiente (los dos primeros sólo son necesarios si está instalado el entorno gráfico, que usa NetworkManager):

```
# systemctl stop NetworkManager.service
# systemctl disable NetworkManager.service
# systemctl enable systemd-networkd
# systemctl enable systemd-resolved.service
```

Podríamos añadir las máquinas para resolución de DNS en /etc/resolv.conf con:

```
nameserver 193.144.48.30
nameserver 193.144.48.100
```

O bien hacerlo sobre el archivo nativo de systemd (que tiene prioridad sobre el anterior, si contiene los datos) en /etc/systemd/resolved.conf de la siguiente forma:

```
DNS=193.144.48.30,193.144.48.100
```

- Aunque tenemos el comando “ip a” para ver los interfaces, podemos también disponer de los clásicos comandos de manejo de la red (ifconfig, por ejemplo) instalando net-tools:

```
# apt-get install net-tools
```

- Opcionalmente, si queremos instalar el soporte de ratón en consola para poder copiar/pegar:

```
# apt-get update
# apt-get install gpm
```

- Reiniciamos y comprobamos en enp0s8 tiene IP (NAT de VirtualBox) y enp0s3 no la tiene (está en red interna y, de momento, no hay DHCP para proporcionarla)

Pero... necesitamos dos equipos Debian (CZmiDebian pero también CZmiDebianserver para luego poder clonarlo). Para ello, tenemos dos opciones:

- Instalar otra máquina exactamente igual.
- La que vamos a usar, clonar en VirtualBox la máquina.** Para ello, antes de pararla:
 - Borramos el archivo /etc/udev/rules.d/70-persistent-net.rules o 70-persistent-cd.rules, si existe alguno de ellos, en la carpeta (para eliminar todo rastro de los interfaces de red (si no,

- al clonar, pueden aparecer 4 interfaces, los dos de la antigua y los nuevos generados al clonar). En distribuciones recientes, según hemos detectado, no existe este problema.
- Paramos la máquina (“poweroff”, “shutdown –h now”, como os guste más ;-)).
 - En VirtualBox seleccionamos la máquina, botón derecho “Clonar” y **seleccionamos la opción de generar nuevas MAC para todos los adaptadores** (de ahí lo indicado en el primer paso). Lo ideal es que seleccionéis “clonación completa” aunque ocupará más espacio en disco.
 - Arrancamos la nueva máquina, que hemos bautizado como CZmiDebianserver

Ya disponemos de una máquina CZmiDebianServer, pero necesitamos “personalizarlo” para que esté configurado correctamente y actúe como servidor:

- Editamos /etc/hostname y cambiamos el nombre, para que se llame CZmiDebianserver.
- También hay que configurar la red, una vez más con systemd, para que nuestro **CZmiDebianserver** (/etc/systemd/network/enp0s3.network) tenga una IP estática en el primer interface, la red interna, donde actuará como servidor DHCP y clonará los equipos en nuestra red (IP: 10.10.24.10 y sin gateway):

```
[Match]
Name=enp0s3

[Network]
DHCP=NO
Address=10.10.24.10/24
```

Ahora reiniciamos (“reboot”) para ver que todo va bien... ya tenemos las 2 máquinas para trabajar.

Algunos elementos a instalar de manera opcional en las máquinas Debian

Podemos instalar (esto es bastante largo, no lo hagáis durante la instalación que realizaremos en el horario de prácticas) un entorno gráfico sencillo **en el servidor**, con Gnome, mediante:

```
# apt-get update
# apt-get install gnome-core xorg
```

Podríamos añadir gdm3 para disponer de un *login* gráfico. Si no lo hacemos, tendremos que ejecutar “startx” para arrancar el entorno X. NOTAS: En algunas versiones, instala el gdm3 aunque no lo pidamos (si lo queremos eliminar, ejecutamos `apt-get remove gdm3`). En algunas ocasiones, al arrancar muestra un error y debemos ejecutar `dpkg --configure -a` antes de reintentar la instalación.

IMPORTANTE: Si instaláis el entorno gráfico, deshabilitad el Network Manager para que no interfiera en la configuración de red con el “maldito” systemd:

```
# systemctl stop NetworkManager.service
# systemctl disable NetworkManager.service
```

Recomiendo instalar también algunos paquetes de herramientas:

```
# apt-get install rsync
# apt-get install zip
```

Para que funcionen las *Guest Additions* incorporadas en la distribución de Debian, tenemos que instalar algunos paquetes (esto puede variar según la versión de Debian):

```
# apt-get install build-essential
# apt-get install module-assistant
# apt-get install dkms
# m-a prepare
```

Aunque algunas de las funcionalidades de las *Guest additions* están de por sí incorporadas en Debian, para conseguir una integración completa debéis instalar las de la versión de VirtualBox correspondiente. Para ello:

```
# apt-get install linux-headers-$(uname -r)
```

Luego vais a Menú “Dispositivos” -> “Insertar imagen de CD de las Guest Additions” (y ejecutamos mount /dev/cdrom si no realiza el montaje automáticamente) y, como usuario root, en la carpeta /media/cdrom, ejecutamos:

```
# sh ./VBoxLinuxAdditions.run
```

Finalmente añadimos en /etc/modules, los siguientes módulos para cargar en el arranque: vboxguest, vboxsf y vboxvideo, escribiendo cada uno en una línea.

IMPORTANTE: Hemos detectado que en algunas distribuciones de Linux, el arranque se ralentiza de manera importante si las “Guest Additions” no están instaladas (se haya o no instalado el entorno X).

Con todo ello nos aseguramos de que el cambio automático de escritorio al modificar el tamaño de la ventana y el funcionamiento del portapapeles (recomendamos activarlo de forma bidireccional en el menú) se comportarán correctamente.

Hay algún servicio de los interfaces de red Intel (RAPL) que ralentizan también el arranque y que podríamos deshabilitar. Para ello, creamos el archivo /etc/modprobe.d/blacklist.conf y añadimos esta línea:

```
blacklist intel_rapl
```

Si quisiéramos, también opcionalmente, ampliar el espacio para almacenar imágenes, podemos añadir al servidor un disco de uso exclusivo para almacenar las imágenes, que formatearemos y montaremos de la siguiente forma:

- Con fdisk creamos la partición (que será /dev/sdb1) tipo Linux

```
# mkfs -t ext3 /dev/sdb1
# mkdir /home/partimag
```

- Metemos la línea en /etc/fstab:

```
/dev/sdb1    /home/partimag    ext3    defaults    0    2
```

También, teniendo las GuestAdditions, podríamos compartir una carpeta del anfitrión y montarla en el servidor con Debian, de la siguiente manera:

- Creamos la carpeta /mnt/Compartida
- En la configuración de la máquina desde Virtualbox, compartimos una carpeta y la “bautizamos” como “Compartida”
- Manualmente la montamos con:

```
# mount -t vboxsf Compartida /media/Compartida
```

... o podríamos incluir la línea en el fstab y montarla automáticamente

CONFIGURACIÓN DE CZmiDebianserver – Servidor DRBL

Aunque en esta pequeña guía está la información necesaria para una configuración como la planteada en prácticas, en <http://www.clonezilla.org/> disponéis de toda la documentación original de los desarrolladores.

Para instalar el DRBL server, necesitamos importar la llave del repositorio y, para eso, necesitamos instalar una herramienta y el comando de descarga e instalación:

```
# apt-get install gnupg
# wget -q http://drbl.org/GPG-KEY-DRBL -O- | apt-key add -
```

También valdría esta otra URL: <http://drbl.sourceforge.net>

Cualquiera de ellas debería responder “OK”, si todo va bien (“-O-” es una letra O mayúscula con guiones). Os indicará que apt-key “is deprecated” pero, al menos de momento, funciona sin problema.

Ahora, editaremos /etc/apt/sources.list para añadir los repositorios de DRBL:

```
deb http://free.nchc.org.tw/drbl-core drbl stable
```

NOTA: Si apareciera un error indicando que falta del paquete apt-transport-https, ejecutamos:

```
# apt-get install apt-transport-https
```

A continuación ejecutamos:

```
# apt-get update
# apt-get install drbl
```

Para dejar más espacio libre, podemos hacer limpieza:

```
# apt-get clean
```

A la hora de configurar drbl, existe una configuración “para impacientes” que utiliza unos parámetros por defecto, *drbl4imp*, pero no la vamos a usar.

Para realizar la configuración personalizada, ejecutamos:

```
# drblsrv -i
```

Veamos a continuación las diferentes preguntas que nos realizará el instalador:

- Instalar diferentes distribuciones de Linux a través de red: **N**
- Salida de consola: **N** (si seleccionamos Y no veremos nada en la pantalla de los clientes)
- Arquitectura kernel de los clientes: **2** (el mismo nivel binarios de CPU que el servidor, 32 bits)
- Actualizar sistema operativo: **Y** (actualizará los paquetes de Debian, cierto peligro)

Os aparecerá algún mensaje en rojo relacionado con el DHCP, es normal, todavía no tenemos el servidor de DHCP en marcha. No puede haber errores de descarga de archivos, si no, luego nos fallará la configuración en pasos posteriores.

Si pregunta acerca del kernel, la opción por defecto típicamente es suficiente pero en la versión actual hemos comprobado que existe un problema con ese kernel y el instalador de Debian 10.11. Por lo tanto, **seleccionad**

el kernel que incluye soporte “pae” al final del nombre (opción 2). Tendremos que activar esta característica extendida en las máquinas virtuales que queramos arrancar en remoto en VirtualBox, dentro de la configuración de *Sistema* -> *Procesador* (esta característica permite utilizar más de 3GB de RAM en una máquina de 32 bits).

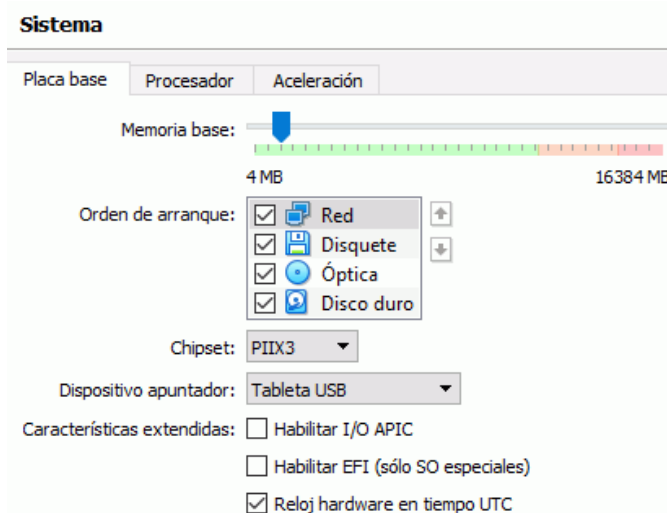
... Y con esto tenemos configurado lo básico de nuestro servidor.

Ahora tenemos que configurar la red y el entorno DRBL/Clonezilla con:

```
# drblpush -i
```

De nuevo se nos presentan diferentes preguntas:

- DNS y NIS/YP: Si no tenemos podemos poner “**dominioCZ.es**” y “**cz**”, pero es indiferente para el objetivo de la práctica.
- Prefijo nombres de cliente: Podemos poner “**CZmi-**” (luego veremos cómo podemos editarlos).
- Red para acceso a Internet: indicamos **enp0s8** y entiende ya que **enp0s3** es para clonación DRBL.
- En algunas versiones, Clonezilla nos muestra un aviso en relación nuestra dirección de red privada (por alguna razón entiende que 10.10.24.X no es una clase C privada... ¡ni caso!).
- Recopilar direcciones MAC de clientes: indicamos **Y** (y encendemos el cliente, **CZmiDebian...** o todas las que queramos manejar). **Para que sean detectadas, necesitamos configurar en VirtualBox la red como primer dispositivo de arranque en la máquina CZmiDebian de esta forma:**



- Sobre fijar las IPs a los clientes y mantenerlas según la MAC: Si indicamos **Y** y quedarán asociadas las IPs a las direcciones MAC. Indiquemos **N** y así tendremos un rango de IPs para que se conecte cualquier equipo por arranque remoto. En ambos casos se puede manualmente añadir rangos e IPs fijas.
- La primera IP a asignar: digamos **101** (¡j es de nuestra red en el interface **enp0s3**!!), se entiende **10.10.24.101**). Si hemos respondido “**N**” en la pregunta previa, se nos preguntará además de la IP inicial, cuantos clientes DRBL tenemos. Si respondemos **10**, en **/etc/dhcp/dhcpd.conf** creará un *pool* de IPs entre la 101 y la 120. Luego veremos cómo tocar manualmente ese archivo.
- Pregunta si aceptamos el rango de DHCP y decimos: **Y**
- De los tres modos seleccionamos: **1**, una única imagen común para los clientes es suficiente.
- Las mismas opciones para el servidor Clonezilla: **1**, única carpeta compartida hacia los clientes también nos llega para nuestro objetivo.
- Directorio para guardar imágenes: **/home/partimag**
- Utilización del swap: **Y** y tamaño a usar: **128MB**
- Modo de los clientes al arrancar: **2** (modo texto)
- Contraseñas de root diferentes que en el servidor: **N**

- Pedir la password en el arranque PXE: **N**
- Configuración del prompt de los clientes: **N**
- Fondo gráfico en el arranque PXE: **N**
- Permitir audio, cdrom, disquete, etc... a los usuarios: **N**
- Configurar la IP pública de los clientes: **N**
- Opción de ejecutar en modo terminal en los clientes: **N**
- NAT para el servidor DRBL: **N**
- Mantener la configuración de clientes anteriores (puede no preguntarlo)
- Guardar toda esta configuración: **Y**

Una vez hecho esto, ya tenemos el servidor preparado. Es importante remarcar las carpetas fundamentales del servidor:

Es necesario mencionar los tres directorios o carpetas importantes para la administración del DRBL/Clonezilla:

1. `/usr/share/drbl` .- Carpeta que contiene los ejecutables de configuración para lanzar la creación y despliegue de las maquinas, así como la configuración del servidor.
2. `/tftboot`
 - a. `/nbi_imag` .- Carpeta más importante, sin ella no se iniciaría el servidor, contiene el kernel inicial de arranque por PXE y la configuración de su menú.
 - b. `/nodes` .- Si escogemos una configuración completa, veremos que nos aparece una carpeta con la configuración de `/etc/` , `/dev/` ,`/root/` y `/var/` por cada equipo cliente que hayamos definido.
 - c. `/node_root` .- Aquí tenemos la definición de todas las carpetas de un Linux para un nodo maestro.
3. `/home/partimag` .- Carpeta donde se almacenan de forma comprimida o no las imágenes de los diferentes sistemas, conviene que sea una carpeta con un gran tamaño para poder almacenar un repositorio de imágenes que nos pueda ofrecer una mayor versatilidad a la hora del despliegue, es decir, hacer despliegue por partición, por disco. O incluso mantener imágenes de diferentes versiones de hardware (en caso de una empresa o corporación de un tamaño considerable).

CONFIGURACIÓN DEL SERVIDOR DHCP EN CZmiDebianserver

Esta configuración se realiza en `/etc/dhcp/dhcpd.conf`

IMPORTANTE: Siempre que cambiemos estas configuraciones (añadir y eliminar equipos), comprobad antes que el servidor está parado (comando “dcs”, opción clonezilla-stop).

Dado que por la red interna (`enp0s3`) no tenemos Internet, lo mejor es dejar comentados estos datos usando # de la siguiente forma, si existen tales líneas:

```
# option routers 10.10.24.10;
# option domain-name-servers XXX.XXX.XXX.XXX;
```

Opciones DHCP:

- `Filename` .- Especifica el fichero con el cual se arrancará por PXE.
- `Subnet` .- Rango de red del DHCP
 - `Next-server`.- servidor PXE de la subred (es necesario que el servidor este en la red de clonación)

- Pool.- Rango de direcciones libres preparadas para ser concedidas a clientes que la soliciten dentro del rango de red (su uso es a través de peticiones broadcast, aunque es posible montarlo a través de un dhcp-relay).
- Host .- Reserva de una IP para una MAC

Si hemos seguido la guía en la instalación mediante drblpush -i y hemos indicado “N” cuando se nos preguntó si queríamos fijar las MAC a las IP, lo que tendremos es un pool de IPs definido:

```
pool {
    # allow members of "DRBL-Client";
    range 10.10.24.101 10.10.24.120;
}
```

Vamos a centrarnos en introducir máquinas fijando la IP (y especificando el nombre como queramos). Sería de la siguiente forma (cuidado con las llaves):

```
host miDebian {
    hardware ethernet 08:00:27:c4:af:1a;
    fixed-address 10.10.24.100;
    # option host-name "CZmiDebian";
}

host miDebian01 {
    hardware ethernet 08:00:27:66:b2:d9;
    fixed-address 10.10.24.101;
    # option host-name "CZmiDebian2";
}

host miWinXP {
    hardware ethernet 08:00:27:8a:06:6e;
    fixed-address 10.10.24.200;
    # option host-name "CZmiWin";
}

host miWinXP01 {
    hardware ethernet 08:00:27:f3:58:69;
    fixed-address 10.10.24.201;
    # option host-name "CZmiWin01";
}
```

NOTA: Comprobad que tanto las definiciones de IPs estáticas como el pool están dentro de las llaves correspondientes a la subnet 10.10.24.0; esto es, tiene que quedaros una llave de cierre adicional a lo que arriba veis, justo debajo de todas las definiciones que se añadan.

Cuando tengamos ya todas las máquinas que queremos *manejar* con Clonezilla, tenemos que reiniciar el DHCP y generar el “perfil” para cada máquina (si no, para Clonezilla no existirán, las hemos creado manualmente sin utilizar su instalador). Ejecutamos:

```
# dcs
```

Ahí seleccionamos: All -> more -> clean-dhcpd-lease

Si no muestra ningún mensaje, ejecutamos directamente en consola `/etc/init.d/isc-dhcp-server restart` (en algunas versiones de Clonezilla se ha detectado que esa opción arriba indicada no reinicia correctamente el dhcpd, aunque debería) y ejecutamos de nuevo:

```
# dcs
```


Esta vez seleccionamos: All -> more -> gen_3N_conf

CREACIÓN DE UNA IMAGEN: Clonación PC->imagen

... Llegado este punto, clonar y restaurar imágenes es muy sencillo. Lanzamos:

```
# dcs
```

Ahí le indicamos si deseamos clonar equipos concretos o todos los equipos (si sólo tenemos un pool de IPs definido en el DHCP, no podremos elegir la MAC concreta a manejar, será cualquiera que arranque por red) y seleccionamos *Clonezilla start*.

Podemos salvar una partición o un disco completo (guardar una copia en el servidor) o bien restaurar una copia que ya está en el servidor a uno o varios clientes. En este caso, si se utilizan las opciones de clonación por *multicast* podremos concretar cuantos equipos queremos clonar, esperar un tiempo (y clonará los que arranquen en ese período) o una combinación de ambos. Fijaos en que al arrancar los clientes, los primeros en llegar se quedan en espera (fondo azul con cuadro blanco, sin más información) hasta que se completa el número de equipos a clonar o bien el tiempo límite que hemos marcado.

En el caso de elegir clonación por *unicast*, se enviará una copia separada de los datos a cada uno de los clientes y, de esta forma, se realizan clonaciones asíncronas (cada equipo inicia y finaliza a su ritmo, no hay sincronización entre ellos); no habrá pantalla de espera, van realizando el proceso independientemente.

PREPARACIÓN/ADAPTACIÓN DE UNA IMAGEN LINUX DEBIAN PARA "MULTICLONAR"

Partimos de una máquina Debian preparada para clonar (CZmiDebian). En este caso no tenemos una herramienta que nos facilite el trabajo en el primer arranque, así que tendremos que crear unos scripts que realicen ciertas tareas.

Nos situamos, como usuario root, en /root y descargamos y descomprimos los scripts mediante:

```
# wget https://lia2.udc.es/xi/scripts.linux.zip
# apt-get install zip
# unzip ./scripts.linux.zip
```

Ahí tenéis

- **nueva**: es el script que debemos ejecutar cuanto tengamos el equipo listo para ser clonado. Realiza una limpieza en el S.O. justo antes de clonarlo. Hace limpieza y genera el archivo /clonar.
- **trasclonar**: se ejecuta la primera vez que arranca el equipo clonado (chequea si existe el archivo /clonar en el disco y, si existe, comienza su ejecución).

Para dejar todo en marcha, primero les proporcionamos permisos de ejecución:

```
# chmod 700 nueva
# chmod 700 trasclonar
```

Copiamos *trasclonar* a /etc/init.d y, para que systemd lo utilice, ejecutamos:

```
# update-rc.d trasclonar defaults
```

NOTA: Por defecto, systemd no permite mostrar los mensajes de los scripts en el arranque (están en /var/log/syslog). Podéis ejecutar el siguiente comando para ver sólo los mensajes desde el último arranque:

```
# journalctl -b
```

Y en este punto paramos la máquina, realizamos la imagen con Clonezilla y la replicamos las veces que necesitemos (en el primer arranque ejecutará *trasclonar* y luego ya obviará la preparación de primer arranque).

Algunos detalles finales en relación a la clonación de imágenes Linux:

- Para evitar problemas, es recomendable montar las unidades (/etc/fstab) y el grub (/boot/grub/grub.cfg) mediante nombre físico de dispositivo (/dev/sda1, /dev/sdb2, etc.) y no UID.
- En /etc/hostname, si ponemos "(none)" y en los archivos de configuración de red en /etc/systemd añadimos "UseHostname=yes", el sistema debería tomar el nombre proporcionado por el servidor DHCP (aunque es una funcionalidad que está dando problemas en varias distribuciones).
- En las últimas versiones de systemd, la petición DHCP va ligada a un "ID de DHCP" (DUID). Para que las peticiones al servidor DHCP vayan con DUID distintos (y evitemos sorpresas de asignación de la misma IP a varios equipos), deberemos borrar en cada clon el archivo /etc/machine-id (dentro del script "nueva") con: `echo -n > /etc/machine-id`

PREPARACIÓN/ADAPTACIÓN DE UNA IMAGEN WINDOWS 8/8.1/10 PARA "MULTICLONAR"

Partiremos de una imagen importada en VirtualBox (<https://lia2.udc.es/xi/CZmiWin10.ova>). Esta máquina ya está activada y tiene instalado el Windows ADK (*Automated Assessment and Deployment Kit*) e **install.win** con todos los paquetes necesarios.

Aquí se pueden ver en detalle todas las secciones configurables:

[https://technet.microsoft.com/es-es/library/cc749307\(v=ws.10\).aspx](https://technet.microsoft.com/es-es/library/cc749307(v=ws.10).aspx)

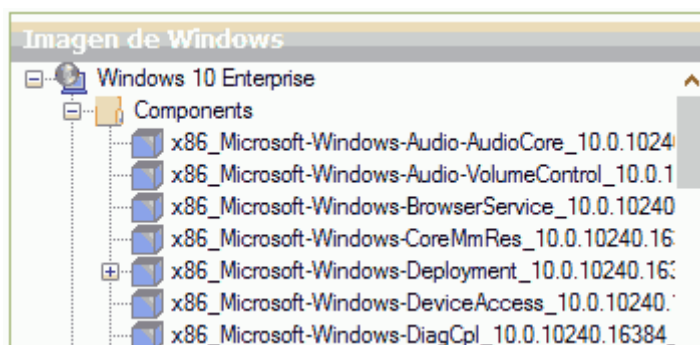
<https://technet.microsoft.com/es-es/library/cc722132%28v=ws.10%29.aspx>

Y un ejemplo sencillo de configuración aquí:

<http://blog.directionstraining.com/microsoft-windows-10/windows-10-deployments-moving-forward>

De todas formas, en este documento veremos la configuración de algunos parámetros paso a paso:

- Instalamos el software general que precisemos en el equipo (y que se replicará en los clones).
- Iniciamos "**Windows System Image Manager**" y seleccionamos como imagen de windows "**install.wim**" (casi 3GB más) de la carpeta "sources" del disco de instalación de Windows 8/8.1/10 (en la imagen CZmiWin.ova se puede encontrar en C:\SYSPREP y añadida, no es necesario este paso).



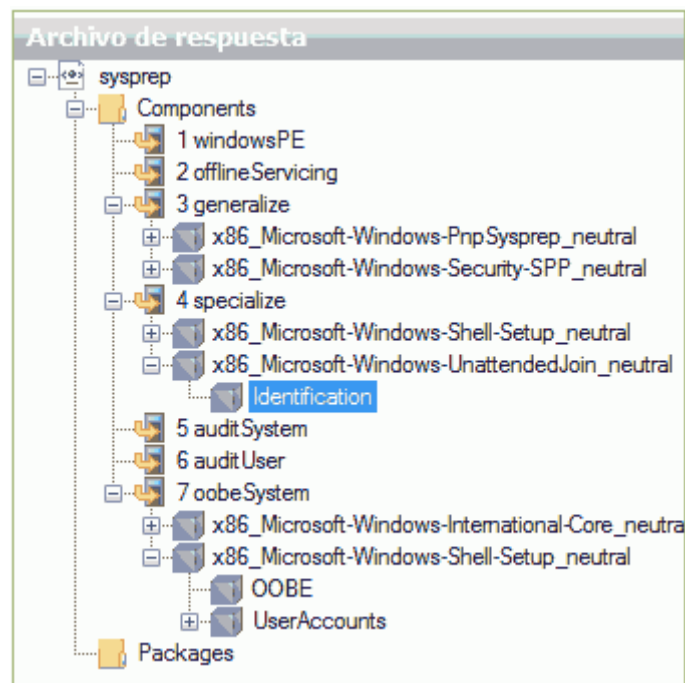
- Desde **Imagen de Windows -> Componentes** añadimos los elementos requeridos a los diferentes apartados de **Archivo de respuesta**, por comentar los que nos interesan para esta práctica:

- **Generalize:** Permite configurar mínimamente el comando '/generalize' de sysprep, así como configurar otros ajustes de Windows que debe persistir en su imagen de referencia.

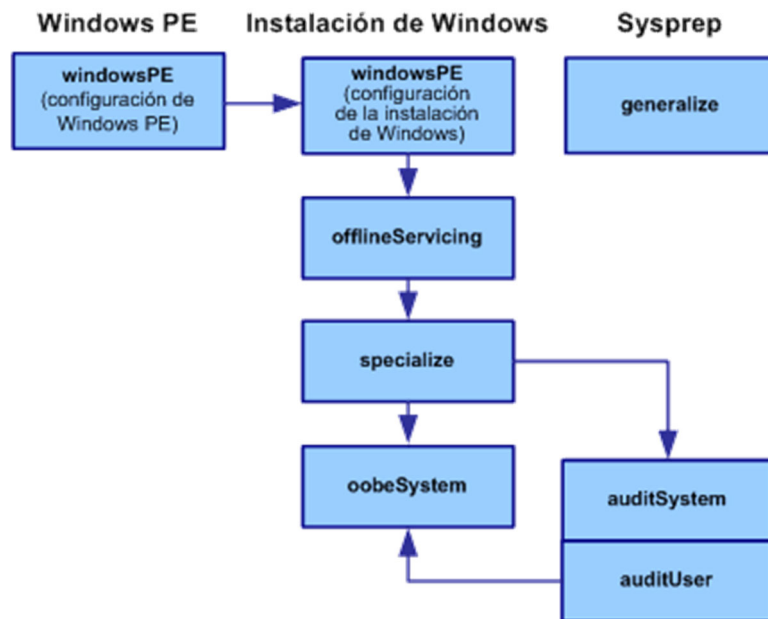
El comando '/generalize' de sysprep ejecuta la fase de configuración (similar a la de una instalación estándar). En el tipo de clonación que estamos realizando, típicamente aquí añadiremos excepciones a esta generalización.

- **Specialize:** Crea y aplica información específica del sistema (configurar los ajustes de red, ajustes internacionales de la hora y el idioma, y la información del dominio).
- **oobe System:** Aplica la configuración a Windows antes de que Windows inicie la pantalla de bienvenida (Out-Of-Box Experience).

Los componentes se añaden con el botón derecho desde la ventana “Imagen de Windows” hacia la ventana de “Archivo de respuesta” (donde podremos editar los diferentes parámetros).



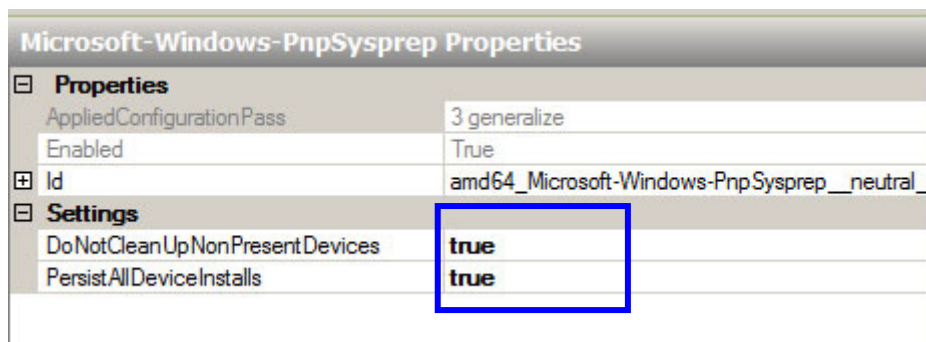
La secuencia de arranque del ordenador en Windows 10 es la siguiente:



Veamos a continuación algunos componentes interesantes y de ejemplo que nos permitirán realizar un arranque desatendido (orientado a una clonación):

- **Microsoft-Windows-PnpSysprep (Generalize)**

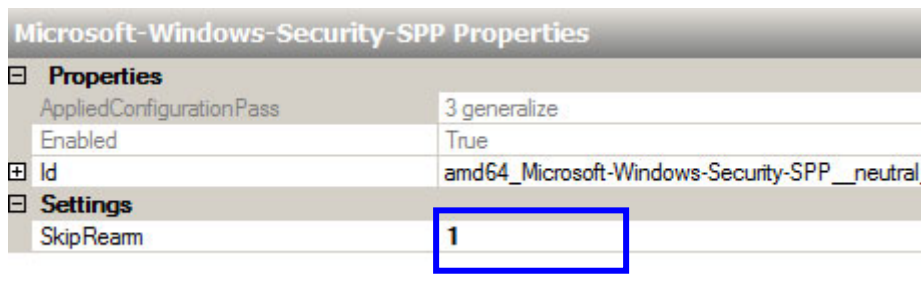
- **DoNotCleanUpNonPresentDevices:** durante la fase de configuración, plug and play guarda la información de los dispositivos. Si indicamos “true” esto se mantiene.
- **PersistAllDeviceInstalls:** los dispositivos plug and play no se desinstalarán durante esta fase si indicamos “true”.



- **Microsoft-Windows-Security-SPP (Generalize)**

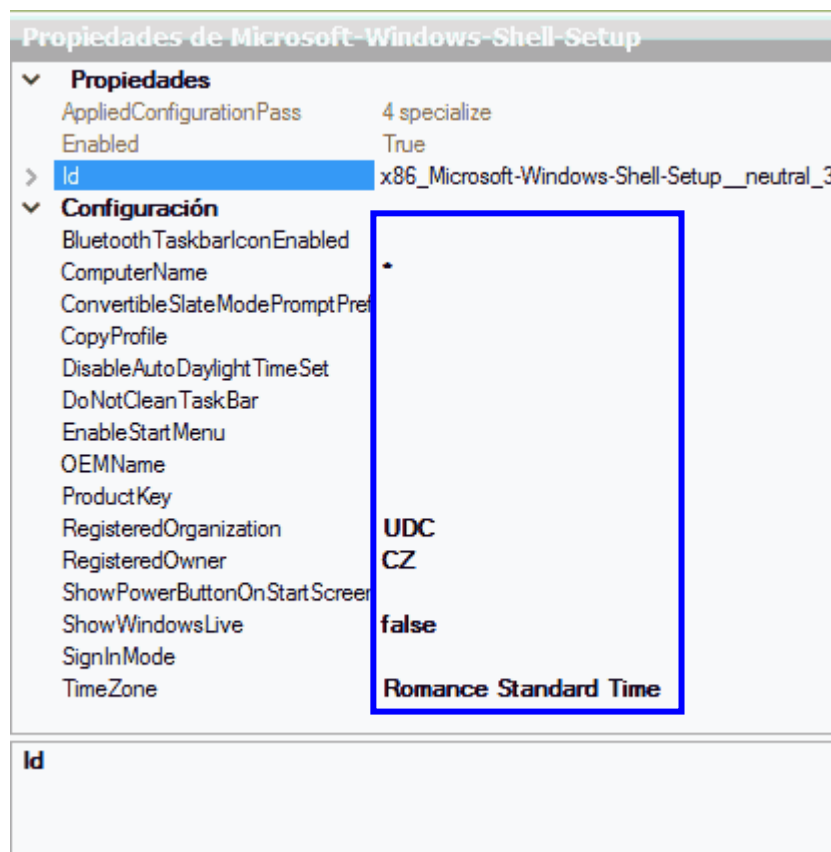
Restablece los valores de licencia que se establecieron durante la instalación del sistema y pruebas de imagen. Se restaura el equipo a un estado de licencia inicial. En Windows 7 el nombre del componente era *Microsoft-Windows-Security-Licensing-SLC*.

- **SkipRearm:** especifica que el equipo no se rearmó y que no será restaurado a la configuración original. Las licencias de activación no serán reiniciadas.



- **Microsoft-Windows-Shell-Setup (Specialize)**

- Podríamos indicar una **ProductKey** si tenemos una válida para varios puestos.
- En **ComputerName** indicamos "*", así se generará un nombre pseudo-aleatorio.
- La información sobre el propietario del ordenador.
- Información sobre la organización del propietario del ordenador.
- Si se mostrará un enlace a Windows Live en el menú Inicio (en W10 no tiene efecto)
- La zona horaria del equipo.



Hay muchas más opciones desplegando este componente en la ventana inferior izquierda (por ejemplo, podemos añadir información sobre el instalador de esta imagen, el denominado OEM)

- **Microsoft-Windows-UnattendedJoin \ Identification (Specialize)**

- Aquí nos unimos automáticamente al grupo de trabajo "GrupoCZ". Utilizando la propiedad **JoinDomain** podríamos unirnos a un dominio.

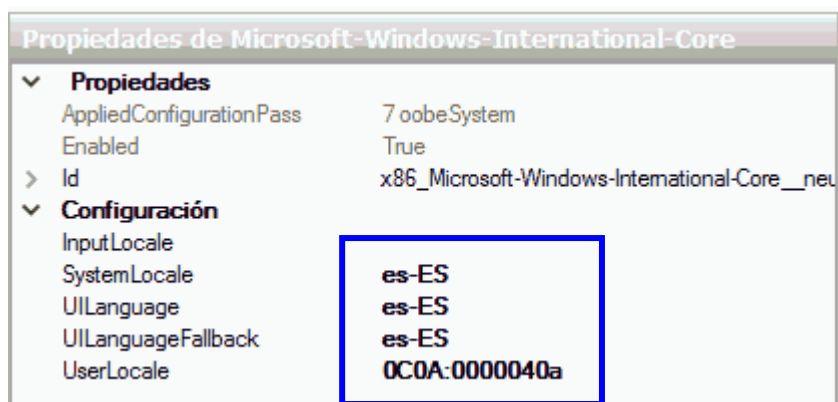


- **Microsoft-Windows-International-Core (OOBE)**

Especifica el idioma por defecto, la configuración regional, y otros ajustes internacionales para su uso durante la *Bienvenida de Windows*, evitando que lo pregunte.

La configuración de este componente, al igual que muchos otros, también se podría aplicar en la fase windowsPE aunque aquí lo haremos al final (OOBE).

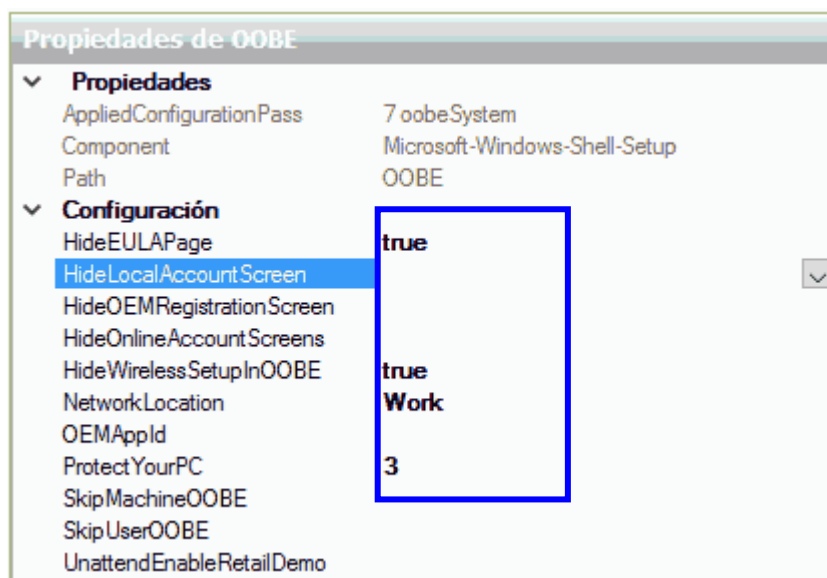
- **SystemLocale**: idioma de la zona donde está instalado el equipo
- **UILanguage**: idioma por defecto para la interfaz de usuario.
- **UILanguage Fallback**: idioma para los recursos que no están localizados en la interfaz de usuario.
- **UserLocale**: ID de validación local del idioma.



- **Microsoft-Windows-Shell-Setup \ OOBE (OOBE)**

Con esto también evitaremos algunas preguntas de la *Bienvenida de Windows* en el arranque:

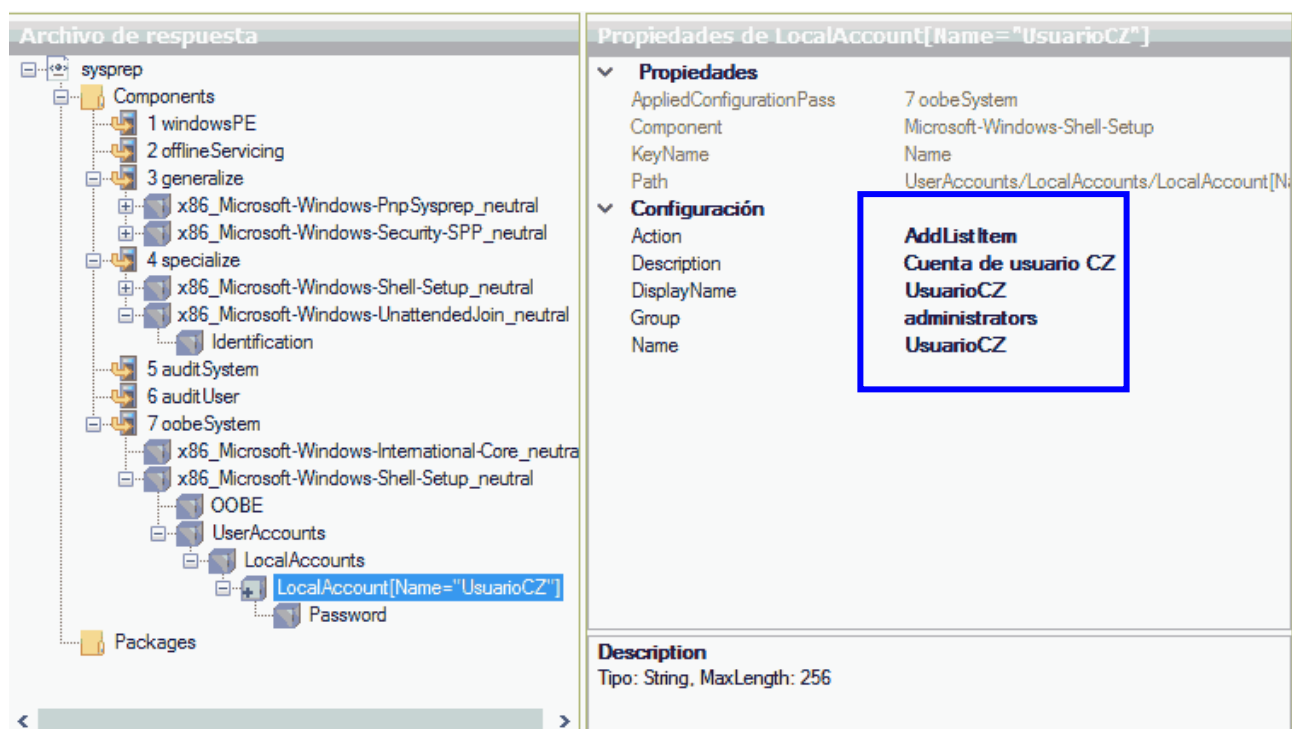
- **HideEULAPage**: aceptar automáticamente los Términos de licencia de software de Microsoft.
- **HideWirelessSetupInOOBE**: durante el inicio de Windows no muestra la pantalla de la red de conexión inalámbrica.
- **NetworkLocation**: especifica la ubicación de la red (opción no funcional en W10).
- **ProtectYourPC**: la protección de las actualizaciones de Windows están deshabilitadas por defecto.



- **Microsoft-Windows-Shell-Setup \ UserAccounts \ LocalAccounts (OOBE)**

Para añadir cuentas locales utilizamos el botón derecho indicando “Insertar nuevo LocalAccount” (Usaremos **DomainAccounts** si son cuentas del dominio).

- Insertamos los datos del usuario y desplegando el “+” podremos introducir el password “cz”.



Para terminar, cerramos la aplicación, guardamos sysprep.xml en c:\sysprep y ejecutamos sysprep como administrador, justo antes de capturar la imagen con Clonezilla, por ejemplo:

```
C:\Windows\System32\Sysprep\Sysprep.exe /generalize /oobe /shutdown /unattend:
C:\SYSPREP\sysprep.xml
```

Siendo sysprep.xml el archivo de respuestas que habremos guardado y que en WDK denominan “default user profile”.

El quipo ahora se apagará y al arrancar no preguntará nada para su configuración y leerá los valores del archivo de respuestas. Entraremos con alguno de los usuarios disponibles (si se desea, el creado ahora) y podremos comprobar en *Panel de control -> Sistema y seguridad -> Sistema* como ha tomado un nombre pseudo-aleatorio y el grupo de trabajo indicado.

NOTAS:

En algunos tipos de licencias, incluso de volumen (varios equipos), podría ser necesario lanzar la activación por comandos mediante un script en **C:\Windows\Setup\scripts** con el nombre **setupcomplete.cmd** que contenga:

```
cscript C:\windows\system32\slmgr.vbs -ipk MAK  
cscript C:\windows\system32\slmgr.vbs -ato
```

Sysprep va a fallar si se detectan aplicaciones de la Tienda de Windows que han sido instaladas/actualizadas de forma independiente por los usuarios de la máquina. Podemos eliminarlas una a una pero, si queremos eliminarlas todas, ejecutamos en el PowerShell (como administrador, opción del botón derecho):

```
Get-AppxPackage -AllUsers | Remove-AppxPackage
```

... aquí veréis algunos mensajes en rojo porque hay aplicaciones que no es posible desinstalar

Y también este otro si queremos que ningún nuevo usuario tenga nada preinstalado:

```
Get-AppXProvisionedPackage -Online | Remove-AppXProvisionedPackage -Online
```

Y ya en este punto podemos volver a ejecutar sysprep.

PREPARACIÓN/ADAPTACIÓN DE UNA IMAGEN WINDOWS XP PARA “MULTICLONAR” (antiguo)

Partimos de una máquina virtual con Windows XP (CZmiWinXP). Una vez preparada con todo su software, cuando ya la queremos clonar, ejecutamos los siguientes pasos antes de lanzar el proceso de generación de la imagen (si no se indica nada, utilizamos la opción por defecto):

1. Descargamos el archivo <https://lia2.udc.es/xi/scripts.winXP.zip> con el navegador web y copiamos el contenido del .zip en C:\ (tendremos una carpeta SYSPREP en C:\). También podemos descargar simplemente el sysprep de Microsoft (deploy.cab) directamente de Microsoft: <http://www.microsoft.com/en-us/download/details.aspx?id=11282>
2. Descomprimos su contenido en C:\SYSPREP
3. Lanzamos setupmgr.exe para generar un fichero de respuestas automáticas
4. Seleccionamos SYSPREP (2ª opción) → Windows XP -> “Yes, fully automate the installation”
5. Nombre y Organización: “CZ” y “UDC”
6. Zona horaria GMT+1
7. Introducimos la clave de Windows: DHFWC-3DY73-D33KB-4HDK-WCPDM (UDC)
8. Seleccionamos una configuración automática de nombre de equipo
9. Si indicamos “*” y “*” cuando configuramos el password, mantiene el actual
10. Configuración típica de red
11. Indicamos Workgroup “CZ”
12. En lengua, utilizamos “Europa occidental y Estados Unidos”

13. En “Identification string” ponemos “Imagen CZ” (esto quedará anotado en el registro, con *regedit* podéis verlo en “HKEY_LOCAL_MACHINE\SYSTEM\Setup\OemDuplicatorString”)
14. Al pulsar “Next” se acaba el proceso y cerramos la ventana (cerrarla, sí... ¡curioso!).

Con esto ya tenemos listo nuestro archivo de respuestas para que, al arrancar por primera vez, resuelva todo tal y como le hemos indicado (sysprep.inf). No hay que generarlo cada vez, con copiarlo a la carpeta C:\SYSPREP y hacer lo que veremos a continuación sería suficiente.

El script **nueva.bat** hace limpieza en la papelera, elimina unidades conectadas y deshabilita el servicio de escritorio remoto RDP. Finalmente lanza sysprep de la siguiente manera (obviamente, se puede también lanzar de forma manual):

```
sysprep -forcesshutdown -reseal -mini
```

NOTA: podemos cambiar `-forcesshutdown` por `-reboot` para que se reinicie el equipo al terminar...
¡PERO ENTONCES TENED PREPARADO YA EL SERVIDOR CLONEZILLA Y NO OLVIDÉIS CAMBIAR EL ORDEN DE ARRANQUE EN LA MÁQUINA WINDOWS PARA QUE ARRANQUE POR RED!

En este punto, previo al primer arranque de sysprep, realizamos la imagen con Clonezilla y la replicamos las veces que necesitemos.

MÁS INFORMACIÓN ÚTIL...

Si en Linux queremos cambiar el idioma en la terminal:

```
# export LANG=es_ES  
# export LANG=us_US
```

En las distribuciones recientes, 10 o superior, el comando que deberíamos usar para convertirnos en root manteniendo los PATH necesarios es:

```
# su -
```

Si al arrancar una máquina Linux clonada en una MV con una MAC diferente genera ethX que antes no existían (típicamente genera un eth1 cuando el original era eth0), lo más sencillo es eliminar alguno de estos archivos (puede ser uno u otro según la distribución):

```
/etc/udev/rules.d/70-persistent-net.rules  
/etc/udev/rules.d/70-persistent-cd.rules
```

Y reiniciar... y lo generará de nuevo con la interface como eth0.

Este archivo no existe en las últimas distribuciones de Debian y Ubuntu. En las más recientes, 10 o posteriores, las máquinas incorporan un archivo machine-id en:

```
/etc/machine-id  
/var/lib/dbus/machine-id
```

Si borramos el contenido de estos archivos, en el arranque de los clones todas las máquinas tendrán diferente ID y no creará problema el DHCP (al tratarse de machine-id iguales, el DHCP proporciona la misma IP aunque su dirección MAC sea distinta!).

Si en Windows alguna vez clonamos una partición y no arranca, podemos entrar en la consola de recuperación y ejecutar:

```
FIXBOOT  
FIXMBR
```

Con BOOTCFG /SCAN se pueden localizar instalaciones de Windows.

Migrar instalaciones hardware de Windows a máquina virtual en VirtualBox:

https://www.virtualbox.org/wiki/Migrate_Windows

Activar Wake on LAN en Linux y lanzar encendido:

```
# ethtool -s np0 wol g  
# etherwake las3.dirección.mac
```

Si con ethtool eth0 nos indica "d" es que está desactivado.

Hay que activarlo en la BIOS de los equipos. VirtualBox no soporta WOL para MV.

Montar carpetas compartidas de Windows (CIFS) en Linux

```
# apt-get install cifs-utils  
# mount.cifs //maquina/compartido punto.montaje.en.linux -o user=nombreusuario
```

Ahí nos preguntará el password y ya está.

Reiniciado de equipos remotos Windows / Linux

En Windows, el comando en modo consola es:

```
shutdown -r -m "nombre_equipo"
```

Comando modo gráfico

```
# shutdown -i
```

Para que esto funcione, es necesario tocar las directivas de grupo:

- 1- Accedemos al Panel de Control y luego a Herramientas administrativas.
- 2- Luego accedemos a Directiva de seguridad local.
- 3- En la nueva ventana, navegamos por Directivas locales/Asignación de derechos de usuario y sobre la sección derecha buscamos la directiva "Forzar el apagado desde un sistema remoto".
- 4- Agregar grupo Todos.

En Linux...

```
# ssh root@10.10.5.118 date
```

Si lo vas a usar para varias veces el mismo equipo y no quieres que compruebe la huella digital del servidor, utiliza la opción StrictHostKeyChecking:

```
# ssh -o 'StrictHostKeyChecking no' root@10.10.5.118 date
```

Instalación de FreeNX (escritorio remoto) en Debian:

<http://www.debianhelp.co.uk/freenx.htm>

Limitado a 2 clientes, podemos también utilizar la paquetería disponible de NoMachine:

<https://www.nomachine.com/>

Eliminar network manager

Para Debian

```
# sudo /etc/init.d/network-manager stop
# sudo update-rc.d network-manager remove
```

Para Ubuntu y “derivados”

```
# sudo stop network-manager
# echo "manual" | sudo tee /etc/init/networkmanager.override
```

Cambiar los colores del terminal

```
# setterm -foreground cyan -background black --bold on -store
```