

Si utilizas máquinas virtuales **UEFI** en **ProxmoxVE** te interesará saber como es la partición mínima y los archivos mínimos para que cargue el UEFI y no te quedes en el shell.

La partición tiene que tener estas características:

- Tiene que ser una partición FAT32 dentro de un disco con sistemas de particiones GPT.
- Debe tener dentro la siguiente estructura mínima de carpetas y archivos:

```
/EFI/boot/bootx64.efi
```

En la configuración por defecto de OVMF, el archivo **bootx64.efi** es lo primero que se busca si no hay ninguna configuración personal ya realizada. Y debe estar exactamente en esa carpeta y exactamente con ese nombre.

Es decir, si OVMF no tiene configurado cargar algún .efi especificado previamente, lo que hará será intentar cargar **/EFI/boot/bootx64.efi** desde la particion FAT32 especial. Si no encuentra el archivo acabarás directamente en la Shell UEFI donde podrás escribir comandos o salirte con «exit», lo que te dejará directamente en las opciones de la BIOS UEFI de OVMF.

¿ Y qué es ese archivo **bootx64.efi** realmente? Pues no es otra cosa que un archivo .efi programado por alguien. De hecho, puede ser el mismo **grubx64.efi**, de Grub, renombrado a **bootx64.efi**. O cualquier otro, como algún .efi de Windows o de Clover (hackintosh).

Ahora bien, aquí hay que entender algo: una cosa es que OVMF llame al archivo **/EFI/boot/bootx64.efi** y otra cosa es a quién llama el archivo **bootx64.efi** una vez cargado. Puede ser que, para bootear correctamente, no te alcance sólo con el archivo bootx64.efi en esa ubicación si éste está programado para, a su vez, buscar otros archivos que complementan su inicio. Como Grub, que además buscará grub.cfg en alguna carpeta específica.

## EJEMPLOS

### Debian

La instalación de Debian en un disco GPT crea en la partición EFI una estructura de carpetas y archivos siendo los mínimos necesarios para arrancar éstos:

```
/EFI/debian/grubx64.efi  
/EFI/debian/grub.cfg
```

Es decir, el archivo **grubx64.efi** de Debian busca el archivo **grub.cfg** en **/EFI/debian/**, de forma que si quieres preparar una partición mínima para bootear Debian renombrando grubx64.efi a bootx64.efi, la estructura mínima de carpetas y archivos de la partición EFI debería ser:

```
/EFI/boot/bootx64.efi  
/EFI/debian/grub.cfg
```

### Ubuntu

Con Ubuntu pasa algo parecido. La instalación de Ubuntu en un disco GPT crea en la partición EFI una estructura de carpetas y archivos siendo los mínimos necesarios para arrancar éstos:

```
/EFI/ubuntu/grubx64.efi  
/EFI/ubuntu/grub.cfg
```

Es decir, el archivo **grubx64.efi** de Ubuntu busca el archivo **grub.cfg** en **/EFI/ubuntu/**, de forma que si quieres preparar una

partición mínima para bootear Ubuntu renombrando grubx64.efi a bootx64.efi, la estructura mínima de carpetas y archivos de la partición EFI debería ser:

```
/EFI/boot/bootx64.efi  
/EFI/ubuntu/grub.cfg
```

## Windows 10

Windows 10 es un poco diferente. Por defecto, la instalación en español de Windows 10 en un disco GPT crea en la partición EFI una estructura de carpetas y archivos siendo los mínimos necesarios para arrancar éstos:

```
\EFI\Microsoft\Boot\BCD  
\EFI\Microsoft\Boot\bootmgfw.efi
```

Es decir, parece que el archivo **bootmgfw.efi** de Windows 10 busca el archivo **BCD** en **\EFI\Microsoft\Boot\**, pero en realidad lo busca en la misma carpeta donde se encuentra el mismo **bootmgfw.efi**, de forma que si quieres preparar una partición mínima para bootear Windows 10 renombrando bootmgfw.efi a bootx64.efi, la estructura mínima de carpetas y archivos de la partición EFI debería ser:

```
\EFI\boot\BCD  
\EFI\boot\bootx64.efi
```

A este respecto cabría decir que Windows también instala muchos otros archivos en la carpeta **\EFI\Microsoft** de la partición EFI. Yo dejaría esa carpeta sin borrar si sólo quisiera tener windows 10 en esa máquina virtual OVMF

## Clover (Hackintosh)

La instalación de Clover en un disco GPT crea en la partición EFI una estructura de carpetas y archivos siendo los mínimos necesarios para arrancar éstos:

