

Este hack es interesante. Imaginemos que iniciamos un ordenador desde un grub, pero realmente queremos cargar otro grub.cfg, y no el que tiene por defecto asociado nuestra instalación actual. Si sabemos la etiqueta de la partición y en qué número de partición esta, podemos lograrlo.

Por ejemplo, supongamos que el archivo grub.cfg está en la carpeta /boot/grub/grub.cfg de la cuarta partición GPT y que ésta tiene como etiqueta «UbuntuNico». Entonces, en la consola de recuperación de grub ejecutaríamos:

```
search --set=root --label UbuntuNico --hint hd0,gpt4
configfile /boot/grub/grub.cfg
```

Y así, sin más, después de ejecutar esa segunda línea, grub cargará ese archivo .cfg con todas las opciones de inicio que éste tenga configurado dentro.

También se puede hacer de esta otra forma, buscando por UUID:

```
search.fs_uuid xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx root
set prefix=($root)'/boot/grub'
configfile $prefix/grub.cfg
```

Por ejemplo, con esta última forma, podemos llamar a /EFI/Debian/grub.cfg desde /EFI/Boot/grub.cfg, estando ambas carpetas en la misma partición FAT32, si lo hacemos de esta manera:

```
search.fs_uuid xxxx-xxxx vSistemaDeArchivos
set vRutaCompletaACarpetaConCfg=($vSistemaDeArchivos)'/EFI/Grub'
configfile $vRutaCompletaACarpetaConCfg/grub.cfg
```

O, directamente:

```
search.fs_uuid xxxx-xxxx vSistemaDeArchivos
configfile ($vSistemaDeArchivos)/EFI/Grub/grub.cfg
```

