Quake I&II Mini Como

Álvaro Villalba Poncet, alvarovp@mad.servicom.es ver 0.7, 22 de Enero de 1999.

Guia para configurar el Quake I y el II en linux.

Índice General

1 Introducción. **2**

2 Requisitos. **2**

2.1 Software. 2

2.2 Hardware: 2

2.2.1 Configuración de la tarjeta de video. 2

2.2.2 Configuración de tarjetas aceleradoras. 3

2.2.3 Configuración del ratón. 5

2.2.4 Configuración de la tarjeta de sonido. 5

3 Quake : **5**

3.1 Principales problemas: 6

3.2 Juego en Red (Quake World). 7

3.3 Añadidos al quake. 7

4 Quake II **8**

4.1 Juego en Red 9

4.2 Principales Problemas 9

4.3 Principales modificaciones. 9

5 Cómo montar un server dedicado. **9**

5.1 Quake 10

5.2 QuakeWorld. 10

5.3 Quake 2 10

6 Problemas con las tarjetas aceleradoras. **10**

7 Servidores españoles de quake. **12**

8 Agradecimientos: **13**

9 Anexo: El INSFLUG **13**

1 Introducción.

Actualmente no hay muchos juegos "populares" portados a linux, de hecho solo está el doom, el quake (en todas sus versiones) y el InnerWorlds, pero todo se andará (actualmente hay fabricantes que han dicho que nunca portarán sus juegos a linux, algún día se comerán sus palabras ... :)). El quake en linux funciona perfectamente, incluso más rápido que en windows, además si se quiere montar un server de quake es la plataforma perfecta, el único problema que puede haber es con las tarjetas de video modernas (especialmente con las 3dfx de última generación), ya que no todas tienen drivers para linux (pero cada vez son más las soportadas).

2 Requisitos.

2.1 Software.

Nos hacen falta los ejecutables para linux. Se pueden encontrar en muchos sitios , el principal es en idsoftware (`ftp.idsoftware.com`) pero este server va muy lento, el mejor sitio para bajarse todo lo necesario es la página de quake en linux (`captured.com/threewave/linux/index.html`). Ahí se puede bajar todo lo necesario sin tener que sumergirse en los oscuros directorios de idsoftware ... Otro sitio sobre el Quake y Linux es quake.medina.net. Otros sitios donde se puede obtener casi todo el software necesario para el quake son:

- `ftp.cdrom.com` (muy lento, es mejor el mirror que hay en `ftp.doc.is.ac.uk`).
- `ftp.sci.com/quake`.
- `www.quake2.com`

Para descomprimir un fichero `tgz` o `tar.gz` ponemos `tar zxfv nombre_del_fichero`.

2.2 Hardware:

Los que pide el propio juego, osea, una tarjeta de video y un Pentium a 90 (yo he visto el quake corriendo en un 486 bajo linux y funcionaba, así que todo es cuestión de probar ;)).

2.2.1 Configuración de la tarjeta de video.

Se puede jugar al quake desde las X o desde la SVGA, recomiendo la SVGA porque es mas rápida, pero si es un servidor con varios usuarios recomiendo la versión de las X (principalmente porque para correr la versíon SVGA hace falta que el ejecutable sea setuid root, con los consiguientes riesgos de seguridad que esto conlleva), pero cada uno que use la que más le guste ...

Si vas a jugar en las X tendrá que tenerlas configuradas (cualquier duda sobre cómo hacerlo mirela en el XFree86-Como)

Para la SVGA normalmente no hay que tocar nada, el archivo de configuración está en `/etc/vga/libvga.config` (En RedHat) o en `/etc/libvga.config` (Slack). Si algo no va bien con la vga tendrá que editar el fichero y poner los parámetros de su tarjeta (tipo de chipset, RamDac, etc) y del monitor (mírelas en los manuales del ordenador).

2.2.2 Configuración de tarjetas aceleradoras.

Lo primero que necesitaremos es una tarjeta aceleradora; hasta la fecha solo están soportadas las basadas en el chip `3dfx` (en el momento de escribir esto estaban portando los drivers para la `banshee`). Para poder usar una 3dfx necesitaremos bajarnos los siguientes ficheros:

- `glide` son las librerías que acceden directamente a la tarjeta, se pueden obtener de http://glide.xxedgexx.com/3DfxRPMS.html , ahí podemos encontrar versiones en formato `rpm` (para RedHat, OpenLinux, SuSe) así como para glibc (RedHat 5.X, debian 2.X, SuSe 6.X) o, libc5. En esta página se puede obtener tambien información sobre las tarjetas soportadas. Necesitaremos como mínimo dos archivos:
  1. `Glide2x_SDK-2.1` es el kit de desarrollo de la librería glide.
  2. `Glide` son las librerías propiamente dichas, dependiendo del tipo de 3dfx que tengamos necesitaremos:
     (a) `Glide_VG-2.46` para las tarjetas `voodoo 1`.
     (b) `Glide_VR-2.46` para las tarjetas `voodoo rush`.
     (c) `Glide_V2-2.51` para las tarjetas `voodoo 2`.
- `Mesa` es la versión *free* de las librerías `OpenGL`, se pueden obtener de http://www.ssec.wisc.edu/ brianp/Mesa.html. La última versión estable es la 3.0 y de desarrollo la 3.1, es recomendable bajarse la última estable (en este caso la 3.0). De ahí cojeremos dos ficheros:
  1. `MesaLib` las librerías propiamente dichas.
  2. `MesaDemos` demos de la librería.

  Existen binarios de las librerías en todos los formatos, pero en este caso necesitaremos los fuentes porque las tenemos que recompilar.

Una vez obtenidos los ficheros necesarios lo primero que instalaremos son las `glide`, si nos las hemos bajado en rpm lo haremos con

```
[root@unsCAred][~]# rpm -Uhv Glide_V[tipo de tarjeta]-[versi\'{o}n].rpm
```

Si no disponemos del programa rpm podemos usar la utilidad `rpm2cpio` de la siguiente manera (desde el directorio / para que nos cree bien los directorios necesarios):

```
[root@unsCAred][/]# cd /
[root@unsCAred][/]# rpm2cpio archivo.rpm | cpio -i --make-directories
[root@unsCAred][/]# ldconfig
```

Para comprobar que las librerías están correctamente instaladas ejecutamos en el shell el comando:

```
[root@unsCAred][~]# ldconfig -p|grep libglide
[root@unsCAred][~]# ldconfig -p|grep libtexus
```

Ahora instalamos el fichero `Glide2x_SDK-2.1`, del que solo necesitamos los includes Si tenemos rpm ponemos

```
[root@unsCAred][~]# rpm -Uhv Glide2x_SDK-2.1-3.i386.rpm
```

Si no disponemos de rpm usamos nuevamente el rpm2cpio de la siguiente manera:

```
[root@unsCAred][~]# cd /tmp
[root@unsCAred][/tmp]# rpm2cpio Glide2x_SDK-2.1-3.i386.rpm | cpio -i --make-directories
[root@unsCAred][/tmp]# cd /usr/local/
[root@unsCAred][/usr/local]# mkdir glide
[root@unsCAred][/usr/local]# cd glide
[root@unsCAred][/usr/local/glide]# mkdir include
[root@unsCAred][/usr/local/glide]# cd include
[root@unsCAred][/usr/local/glide/include]# cp /tmp/include/* .
[root@unsCAred][/usr/local/glide/include]# ls
3dfx.h      glidesys.h  gump.h      sst1vid.h
glide.h     glideutl.h  linutil.h   texus.h
```

En caso de haber usado el rpm tenemos que hacer un enlace simbólico ya que las librerías Mesa buscarán la librería `libglide` en `/usr/local/glide/lib`, lo hacemos con lo siguiente:

```
[root@unsCAred][~]#cd /usr/local/glide
[root@unsCAred][/usr/local/glide]# mkdir lib
[root@unsCAred][/usr/local/glide]# cd lib
[root@unsCAred][/usr/local/glide/lib]# ln -s /usr/lib/libglide2x.so .
[root@unsCAred][/usr/local/glide/lib]# ls
libglide2x.so@
```

Ahora que ya hemos instalado correctamente las librerías glide nos vamos a por las Mesa. Haremos lo siguiente:

```
[root@unsCAred][~]# cd /tmp
[root@unsCAred][/tmp]# tar zxfv MesaLib*
[...]
[root@unsCAred][/tmp]# tar zxfv MesaDemos*
[...]
[root@unsCAred][/tmp]# cd Mesa-3.0
[root@unsCAred][/tmp/Mesa-3.0]# make realclean
```

Ahora es cuando nos leemos el README que viene con las librerías, una vez hecho eso las compilamos, dependiendo del tipo de tarjeta aceleradora que tengamos pondremos lo siguiente:

- `make linux-386-opt-glide` si tenemos una voodoo 1.
- `make linux-386-opt-V2-glide` si tenemos una voodoo 2.
- si tenemos una voodoo rush podemos hacer uno de estos tres:
  1. `make linux-glide` para linux glide.
  2. `make linux-386-glide` para micros i386.
  3. `make linux-386-glide-mits` micros i386 con multiproceso.

Despues de un rato y si no hemos tenido ningun problema las librerías ya estarán compiladas, ahora lo único que queda es instalarlas. Si ya las teníamos instaladas antes lo único que tenemos que hacer es copiar la librería nueva que hemos generado, para eso hacemos:

```
[root@unsCAred][/tmp/Mesa-3.0]# cd lib
[root@unsCAred][/tmp/Mesa-3.0/lib]# cp -a * /usr/lib
```

He puesto el `/usr/lib` porque es donde las tenía yo instaladas, si ya las teníamos en `/usr/local/lib` las copiaremos ahí. Si no las teníamos instaladas antes lo hacemos poniendo:

```
[root@unsCAred][/tmp/Mesa-3.0]# make install
```

Para concluir, ejecutamos como root el comando `ldconfig` y ya tendremos todo lo necesario para que funcione la 3dfx (bueno, aún nos faltan los ejecutables para el quake, el quakeworld y el quake2).

2.2.3 Configuración del ratón.

Para que funcione el ratón tenemos que editar otra vez el fichero de configuración de la vga (ver 2.2.1) y poner el tipo de ratón que tenemos. Es necesario tener el `gpm` funcionando, que es el que se encarga de controlar el ratón en la consola. Si usa el `gpm` con el parámetro `-R` es recomendable cambiarlo antes de jugar con un script o algo parecido, ya que se nota bastante la velocidad de respuesta del mouse.

En el quake se puede especificar el dispositivo en el que se encuentra el ratón mediante los parámetros `-mdev [device]` y la velocidad con `-mrate [speed]`. Los parámetros por defecto son `/dev/mouse` para el `-mdev` y `1200` para el `-mrate`.

Lo errores con el ratón normalmente se deben a fallos en la configuración del gpm o porque no se ha especificado el tipo de ratón en el `libvga.config`. Además es muy común que cuando se está jugando al quake y se cambia a otra consola el ratón deje de responder, esto se puede arreglar de dos maneras:

- Reiniciando el gpm (en RedHat con `/etc/rc.d/init/gpm restart`)
- Poniendo `echo "*n" >/dev/mouse` en una consola.

Para cualquier duda o problema con un ratón de tres botones es recomendable consultar el `Raton_3_botones-Como`.

2.2.4 Configuración de la tarjeta de sonido.

No es extrictamente necesaria, pero jugar al quake sin sonido es como una primavera sin flores ;) . Para configurar la tarjeta de sonido consultar el `Sonido-Como`.) Para jugar sin sonido ponemos el parámetro `-nosound`. Además de este hay otros parametrós que son:

- `sndbits` [8 or 16]
- `sndspeed [speed]` (8000, 11025, 22051 and 44100), por defecto 11025.
- `sndmono`.
- `sndstereo` (por defecto, siempre que sea posible).

En el quake2 he experimentado problemas con el OSS y el driver para mi tarjeta (una AWE32), ya que cuando se reinicia el sistema de sonido deja de funcionar (esto no ocurre con el driver normal del kernel). El problema estaba en el compilador, si está usando el `OSS`, el `egcs` o el `pgcc` como compilador y un kernel de la serie `2.0.X` puede que tenga problemas, la solución pasa por instalarse el gcc-2.7.2.3 y recompilar el sndconfig con él (además los kernels de esa versión es recomnedable compilarlos con el `gcc` y no con el `egcs/pgcc`).

3 Quake :

Bueno, se acabaron las configuraciones, ahora toca jugar :)

Necesitamos los archivos `pak0.pak` (para la share) y el `pak0.pak` y `pak1.pak` para la registrada. Si tenemos el quake ya instalado en otra partición (por ejemplo en windows) no hace falta copiarlos a la partición linux, podemos usar enlaces simbólicos, lo único que tenemos que hacer es montar la particion de windows, por ejemplo en `/mnt/dos`.

Vamos a instalar el quake en `/usr/local/games/quake` (se puede instalar en cualquier sitio).

- Lo primero es crear ese directorio (`mkdir /usr/local/games/quake`). Un vez hecho eso creamos el directorio `id1` (a partir de ahora todos los directorios van referenciados al `/usr/local/games/quake`) dentro del directorio quake.
- Copiamos el `pak0.pak` y el `pak1.pak` al directorio `id` (si lo hacemos con enlaces simbólicos, y suponiendo que tenemos la partición donde está el `pak0.pak` y el `pak1.pak` montada en `/mnt/dos` ponemos como root:

  ```
  ln -s /mnt/dos/[ruta_al_quake]/quake/id/pak* /usr/local/games/quake/id
  ```

- Descomprimimos el ejecutable para linux en `/usr/local/games/quake`.
- Y ahora ... a jugar, ya no hace falta nada más ... vamos al directorio `/usr/local/games/quake` y ponemos `./squake` (o `xquake` si estamos en las X) y cruzamos los dedos. Si todo a ido bien ya está funcionando el quake. Si usamos una tarjeta aceleradora ejecutaremos `./glquake`.

Para cambiar la resolución de la pantalla se usan dos comandos, nos ponemos en la consola del quake y ponemos `vid_describemodes` que nos da una descripción de todos los modos de video posible (cada uno con un número).

Ahí luego ponemos `vid_mode N` siendo `N` el número que corresponde a la resolución que queremos. Esto no funciona con las tarjetas aceleradoras, para cambiar la resolución en ellas hay que especificarlo en el arranque mediate los parámetros `-width ANCHO -height ALTO`, sino arrancará por defecto en 640x480 (otra resolución muy utilizada es 512x384).

Si queremos medir los *fps* a una resolución podemos hacerlo con `timedemo demo1` desde consola, que intenta ejecutar lo más rápido posible la `demo1` y nos dice los *fps*.

Si queremos comprobar los *fps* durante el juego lo podemos hacer con `timerefresh` (pero depende de la parte de la fase en la que nos encontremos).

3.1 Principales problemas:

Pueden ser:

- La tarjeta de sonido no está configurada y hemos intentado arrancar con sonido. Solución: arrancamos con `./squake -nosound`.
- No tenemos puesto el `HOSTNAME`. Solución: editamos el fichero `/etc/hosts` y añadimos el nombre de nuestra máquina, por ejemplo:

  ```
  127.0.0.1       localhost       localhost.localdomain       unsCAred
  ```

  y nos aseguramos de que en `/etc/HOSTNAME` pone unsCAred (en mi caso claro ;))

- `can't get I/O permisions`. El squake necesita ser *setuid root* , ejecutamos como root

  ```
  chown root.root squake
  chmod 4755 squake/
  ```

  (No me responsabilizo de nada que pase por tener el ejecutable *setuid*) (No se asuste!!!, esto tiene problemas de seguridad, si es la máquina de casa no se te va a romper el ordenador ni nada parecido, simplemente, la svga no es muy segura en cuanto a *seguridad del sistema* se refiere).

- No nos arranca en las X. Solución arrancamos el servidor a 8 bits de profundidad de color con (`startx -bpp 8`).

3.2 Juego en Red (Quake World).

El quake es un buen juego, pero cuando de verdad sorprende y crea adicción es cuando se juega en red. No es lo mismo matar a un monstruo en mitad de un nivel que a tu amigo *pukka* (con lo segundo disfrutas ;)).

Cambia totalmente tu visión del juego y la manera de jugar, los movimientos se aceleran en un 200%. Para jugar en red lo único que necesita es una conexión a internet o a una LAN.

En la página de quake en linux (ver 2.1) se puede encontrar un ejecutable que se llama `qwcl`. Esto es el cliente de QuakeWorld, que no es más que una mejora de la engine de red del quake (solo sirve para jugar en red) con algunos añadidos, como el poder "cambiar" de piel o uniforme.

Se puede jugar en red con el quake normal, pero la mejora con el *qw* es muy notable, de hecho, es muy raro encontrar servers de quake normal en internet, lo normal es que sean de quakeworld.

Una herramienta muy útil para jugar en internet es el `qspy` o el `quickspy` (para consola) que consiste en un programa que localiza servers de quake world y nos dice cuantas personas hay, el ping, etc.

Hace poco ha salido el `xqf` que es un magnífico interface para el `qspy` hecho con gtk (para las X), se puede bajar de `www.botik.ru/ roma/quake`

Para instalar el qw lo único que hay que hacer es descomprimir el archivo `qwcl` en el directorio `/usr/local/games/quake` y poner `./qwcl +connect IP` siendo IP la dirección en donde queremos jugar (ver 7).

Lo mejor es tener la última versión del qw (actualmente la 2.30) y la anterior ya que algunos server no se han actualizado aún.

Si se tiene una tarjeta aceleradora necesitará el `glqwcl`. Para indicarle la resolución de pantalla se hace como en el `glquake` mediante los parámetros `-width ANCHO -height ALTO`.

El parámetro `rate XXXX` controla el numero de *bps* que recibimos del servidor además del número de *fps*. Para un modem de 33.6 el parámetro óptimo es de 3000, para un 28.8 de 2500.

3.3 Añadidos al quake.

Uno de los principales factores que han hecho al quake un juego "de culto" es ademas de su excelente modo de red el que se le puedan añadir variaciones fácilmente.

Puede añadirle nuevas armas, nuevos montruos, nuevos niveles, incluso nuevos "juegos". El más popular es el `Capture the Flag`, que divide a todos los jugadores en dos equipos (es para jugar en red) y le da a cada uno una bandera y una base, la misión de cada equipo es capturar la bandera del otro y llevarla a su base, siempre y cuando no lo capturen la suya. La mayoría de los patches o modificaciones se instalan igual. Supongamos el ctf. Los pasos para instalarlo son.

1. Se crea el directorio `/usr/local/games/quake/ctf`
2. Se descomprimen los ficheros del ctf en ese directorio.
3. Desde el directorio quake se pone `./squake -game ctf`.

Sin duda alguna la mejor variación que se ha hecho del quake son los bots, osea robots dotados de inteligencia artificial para simular el juego en red con *humanos*. Los mejores son el omicron (por lo menos es el que más me gusta a mi ;)) y el reaperbot (una lista completa de los bots, con puntuaciones a cada uno y comentarios se puede encontrar en `www.ionet.net/~mimpchnk`)

Otra variación muy interesante es el `Team Fortress` que se puede encontrar en `www.planetfortress.com/teamfortress>`. Es básicamente el `ctf` solo que podemos elegir distintos *caracteres* cada uno con funciones y armas diferentes. En este mod es donde menos importa las diferencias de ping y donde más se juega en equipo.

El `Rocket Arena` es otra variación del quake, el argunmento es muy simple, *dos entran, uno sale*, osea, son combates uno contra uno en mapas especiales, los dos jugadores salen con todas las armas y armaduras y el que mate al otro se queda y sigue jugando contra el siguiente contrincante. Tambien se pueden configurar los servidores para poder jugar por equipos. Se puede obtener de http://www.planetquake.com/servers/arena/.

4 Quake II

Para jugar al quake2 lo único que hace falta es el cd y el ejecutable, osea que aunque tu disco duro esté lleno con un poco más de 800k que es lo que ocupa el ejecutable puedes jugar. Lo único que hay que hacer es lo siguiente:

- Lo primero, creamos un directorio `/usr/local/games/quake2` (por ejemplo) y dentro de este un directorio `baseq2`.
- Copiamos el fichero `/mnt/cdrom/Install/Data/baseq2/pak0.pak` a este directorio. Si se quiere jugar sin copiarlo se hace un enlace simbólico con

  ```
  ln -s /mnt/cdrom/Install/Data/baseq2/pak0.pak /usr/local/games/quake2/baseq2
  ```

  Atención que el *ficherito* sólo ocupa 160 megas ;).
- Luego creamos el directorio `players` dentro del `baseq2` y los directorios `male` y `female` dentro de este (osea que queda algo así:

  ```
  quake2---|-baseq2---|-players---|-male
                                  |-female
  ```

- Compiamos todos los ficheros que hay en `/mnt/cdrom/Install/Data/baseq2/players/male` en `male` y los del `female` en `female`.
- Descomprimimos el `quake2-3.20-i386-unknown-linux2.0.tar.gz` en el directorio `quake2`.

Y listo, ya podemos jugar.

Si no tiene la librerías `MesaGL` instaladas tendrá que copiar la que viene con el ejecutable a `/usr/local/lib` (por ejemplo) y poner `ldconfig` como root (tiene que ser la version 2.6beta3 como mínimo) e instalar las librerias `glide` si va a usar una 3dfx (ver 2.2.2).

Si se va a jugar en la svga el fichero `quake2` tiene que estar `setuid root` (con el ejecutable vienen un script para poner bien los permisos), simplemente ponemos `./fixperms.sh` como root en el directorio `quake2`).

Para arrancarlo usamos:

```
        ./quake2 +set vid_ref soft (para la svga)
        ./quake2 +set vid_ref softx (para las X)
        ./quake2 +set vid_ref gl (para jugar con 3dfx)
        ./quake2 +set vid_ref xgl (soporte para OpenGL,solo funciona en las X)
        ./quake2.3dfxgl (para usar el mini-port del OpenGL, solo en sistemas
con glibc, es decir RedHat 5.X o debian 2.0)
```

Eso solo se hace la primera vez, luego con poner `./quake2` sirve.

4.1 Juego en Red

Para jugar en red arrancamos el quake2, vamos al `menu multiplayer` y le damos a `join network game`.

Hay dos opciones, `server list`, `update server list` y luego una lista. Le damos a `server list` y añadimos las direcciones de servers que queramos , luego le damos a `update server list` y nos saldrán los servers con la información en la lista, colocado por ping, de menor a mayor.

Tambien puedemos jugar poniendo `+connect IP` desde la consola del quake2 (que sale dandole a la tecla ?). Previamente habremos configurado nuestro jugador en `player config` (el nombre, el modelo, la *skin*, el tipo de conexión y las opciones de descarga que se usan para bajarse *skins*, mapas, sonidos y demás cosas de los servidores).

La mayoria de los servers usan lo mapas de deathmatch (q2dm*). En el quake venían en el cd, pero en el quake2 salieron después con lo que si queremos jugar en internet estamos obligados a bajárnoslos.

Los mapas vienen en la versión 3.13, ocupan unos 10 megas, ver 2.1, que además del ejecutable traen un `pak1.pak` con los mapas de deathmatch).

Se instalan descomprimiendo el archivo en el directorio quake2, pero hay que recordar que tenemos que instalar luego la versión 3.20 que es la que usan los servidores (esta es la última versión que ha salido hasta la fecha, es recomendable tener siempre la última versión y algunas veces necesario :)).

4.2 Principales Problemas

Aparte de los del quake uno (que tambien pasan aquí) un problema del quake 2 es:

- `Cuando juego como usuario y me conecto a un server el programa termina ....` Este es un problema de permisos, con una mala solución ... cuando juega en servers con modificaciones (como el *ctf2* o el *Lithium II*) al conectar se reinicia el sistema de video, y si no esta como root el quake2 "suelta" la id del root después de iniciar el sonido y el vídeo y se produce el problema de I/O perm. La única manera de que esto no pase es jugando como root o usando `sudo`.

Nota: para jugar al quake2 en red NO necesita tener el Cd del quake, sólo tiene que tenerlo instalado (por supuesto, si está jugando desde el cd tendrá que meterlo ;)).

4.3 Principales modificaciones.

Al igual que para quake, para quake 2 hay muchas modificaciones, la más extendida es el *Lithium II*, que añade nuevas funcionalidades al quake (como runas que te dan distintos poderes o el "hook"), de hecho la mayor parte de los servidores usan el *Lithium II*, pero para jugar en ellos no hace falta ningun fichero extra.

Si queremos usar bots como en el quake los únicos que conozco que funcionen en linux son los eraser (ver la url que hay en 3.3) pero consumen muchos recursos (aún así se pueden usar).

Otra modificación que está sustituyendo al ctf2 es el lmctf que es una variación de este con mapas más grandes.

El equivalente al `Team Fortress` del quake2 se llama `Weapons Factory`, que se puede obtener de http://www.captured.com/weaponsfactory/.

El `Rocket Arena 2` tambien está disponible y se puede bajar de la misma página que el `Rocket Arena` para quake (ver 3.3.

5 Cómo montar un server dedicado.

Si tiene una conexión permanente a Internet, ¡¡¡contribuya a la causa y monte un server de quake !!!

5.1 Quake

Para montar un server de quake dedicado solo necesitamos los pak's y por supuesto el ejecutable (que se llama `unixded-1.0-i386-unknown-linux2.0.tar.gz`). Lo ejecutamos (se recomienda usar el `screen` y ponerlo en una consola *detached*) y listo.

El `screen` lo puede encontrar en `prep.ai.mit.edu/pub/gnu` Lo único que sale al arrancar un server dedicado es una consola de texto.

5.2 QuakeWorld.

Para montar un server de quake world necesitamos el fichero `qwsv2.30-i386-unknown-linux2.0.tar.gz`¿.

Una modificación muy interesante para montar un servidor es el `Kombat Teams`, que ofrece una serie de posibilidades a los usuarios como poder cambiar el mapa, fijar el tiempo, el límite de frags, etc. Se puede bajar de http://www.demolition.org/k/.

5.3 Quake 2

Aquí no necesitamos nada nuevo, solamente poner `./quake2 +set dedicated 1 +exec server.cfg`. En `server.cfg` ponemos los mapas que queremos cargar y demás cosas, recordar que si no ponemos la linea *map q2dm1*, (por ejemplo) no se cargará ningun mapa. Igualmente se recomienda se usar el `screen`.

6 Problemas con las tarjetas aceleradoras.

Hay una página con una faq muy completa sobre las 3dfx en www.voodooextreme.com/3Fingers. Tambien hay un programa llamado `GlideControl` disponible en http://www.ling.ed.ac.uk/ oliphant/gc muy útil para configurar los parámetros de las 3dfx. Los problemas más comunes suelen ser:

- Los usuarios no pueden usar la tarjeta aceleradora: Esto se puede solucionar instalando el *Device3Dfx*, que es un paquete que nos crea un dispositivo `/dev/3dfx` y mediante la inserción de un módulo en el kernel (`3dfx`) permite que los usuarios puedan acceder a la tarjeta aceleradora. Asegurese si lo instala de que los permisos de `/dev/3dfx` están correctamente puestos (ejecute `chmod 666 /dev/3dfx`). Este paquete se puede encontrar en http://glide.xxedgexx.com/3DfxRPMS.html.
- Cuando ejecuto el glquake me da un *core dump*:. El quake2 y el glqwcl están disponibles para glibc, sin embargo el glquake no, esto crea bastantes problemas a la hora de hacerlo funcionar. Si tiene una voodoo2 en principio no debería tener problemas porque el propio paquete del Glide_V2 trae además de las versiones de las librerías para glibc las de libc5 (que se instalan en `/usr/i486-linux-libc5/lib`), sin embargo el paquete de la voodoo 1 no las trae, con lo que puede que tenga que bajarse la versión para libc5, extraer las librerías (si está en rpm con el `mc` es muy sencillo hacerlo, sino tendrá que usar el `rpm2cpio` o el `rpmget` ya que si lo instala directamente con el `rpm` sobrescribirá las de la versión para glibc) e instalarlas (en `/usr/i486-linux-libc5/lib`). Además asegurese de que en el fichero `/etc/ld.so.conf` aparece el directorio `/usr/i486-linux-libc5/lib` y de que tiene instalado el paquete de compatibilidad con la libc5 (libc-5.3.X-Y en redhat).

  ```
  ¡asegurese si se baja este paquete de que lo est\'{a} cogiendo de una versi\'{o}n de
  RedHat 5.X o del directorio /libc6 en caso de que lo est\'{e} cogiendo del contrib
  de RedHat, sino le sobreescribir\'{a} las antiguas librer\'{\i}as y no funcionar\'{a} el
  sistema!
  ```

Otra posible causa de error es que falte alguna librería en la ruta de búsqueda del ejecutable, esto se puede comprobar mediante el comando `ldd fichero`. Si por ejemplo al hacer `ldd glquake` le sale alguna librería seguida de la linea *Not Found* ahí estará el problema, asegurese de que tiene esa librería instalada y que el directorio en el que está dicha librería aparece en el fichero `/etc/ld.so.conf`, si no lo está añada el directorio y ejecute luego como root `ldconfig`.

- Cuando ejecuto el glquake me da el error `glquake: can't resolve symbol '__register_frame_info'`. El glquake necesita las `MesaGL-2.6`, si no las tenía (vienen con el ejecutable del quake2) consulte la dirección de las librerias en 2.1, instalelas y ejecute ldconfig.
- El glquake/glqwcl/quake2 se ve muy oscuro. Ajuste el contraste de la tarjeta, esto se hace mediante la variable de entorno `SST_GAMMA` en una voodoo 1 y `SSTV2_GAMMA` en las voodoo 2, por ejemplo poniendo :

```
[unscared][~]$ export SSTV2_GAMMA=1.70
```

Otra opción es bajarse el programa `GlideControl` mencionado anteriormente con el que puede cambiar la *gamma* de la tarjeta además de otros parámetros.

- Al jugar en linux obtengo menos fps que en windows: Bueno, las Mesa no están todavía lo suficientemente optimizadas, y en procesadores *lentos* no dan un buen rendimiento. Una opción es probar el port de opengl que han incluido los autores del quake como opción a las librerías Mesa (se llama `lib3dfxgl.so`). Para usarlo creese un fichero llamado glquake.3dfxgl y meta en él lo siguiente:

```
#!/bin/sh
LD_PRELOAD=./lib3dfxgl.so ./glquake $*
```

Para el quake world puede usar el `glqwcl.3dfxgl` que viene con el `glqwcl` y en el quake2 puede especificar el miniport en las opciones de video. Otra opción es asegurarse de que la `libMesaGL.so.2` es un enlace a la `libMesaGL.so.3`. La versión 3 está bastante más optimizada que la 2.6 y el rendimiento aumenta bastante, el problema es que el glquake necesita la versión 2.6. Una solución *chapucera* es hacer lo siguiente; lo primero vemos donde están las librerías y los ejecutables:

```
[root@unsCAred][/usr/local/games/quake2]# ls -la libMesaGL.so*
lrwxrwxrwx  1 root  root     14 ene 18 19:30 libMesaGL.so -> libMesaGL.so.2
lrwxrwxrwx  1 root  root     16 ene 18 19:30 libMesaGL.so.2 -> libMesaGL.so.2.6
-rwxr-xr-x  1 root  root 897135 dic 26 14:43 libMesaGL.so.2.6
[root@unsCAred][~]# ls -la /usr/lib/libMesaGL.so.2
lrwxrwxrwx  1 root  root     12 ene 13 20:01 /usr/lib/libMesaGL.so.2 -> libMesaGL.so
[root@unsCAred][~]# ls /usr/local/games/quake/glquake*
/usr/local/games/quake/glquake         /usr/local/games/quake/glquake.mesa
/usr/local/games/quake/glquake.3dfxgl
```

ahora hacemos un script que borre el enlace de la libMesaGl.so.2,que añada en el `path` de búsqueda el directorio /usr/local/games/quake2, que ejecute el glquake y que deje todo como estaba. Para usar las MesaGL:

```
[root@unsCAred][~]# cat /usr/local/games/quake/glquake.mesa
#!/bin/sh
# Movemos la libreria para que no la carge:
mv /usr/lib/libMesaGL.so.2 /usr/lib/libMesaGL.so.2.tmp
# A\~{n}adimos el directorio del quake2 a la ruta de b\'{u}squeda
LD_LIBRARY_PATH_OLD = $LD_LIBRARY_PATH
LD_LIBRARY_PATH=/usr/local/games/quake2
# Ejecutamos el glquake
./glquake $*
# Dejamos todo como estaba.
mv /usr/lib/libMesaGL.so.2.tmp /usr/lib/libMesaGL.so.2
LD_LIBRARY_PATH= $LS_LIBRARY_PATH_OLD
```

Para el miniport de 3dfx:

```
[root@unsCAred][~]# cat /usr/local/games/quake/glquake.3dfxgl
#!/bin/sh

# This script preloads the lib3dfxgl.so library which overrides the functions
# present in libMesaGL.so.  This allows glquake to run using 3DFX's miniport
# and without Mesa 3-D.  Glide and X11 libraries have to be installed tho,
# even tho GLQuake doesn't use X11, it's linked against it because Mesa 3-D
# requires them.
mv /usr/lib/libMesaGL.so.2 /usr/lib/libMesaGL.so.2.tmp

LD_LIBRARY_PATH_OLD= $LS_LIBRARY_PATH
LD_LIBRARY_PATH=/usr/local/games/quake2
LD_PRELOAD=./lib3dfxgl.so ./glquake $*
mv /usr/lib/libMesaGL.so.2.tmp /usr/lib/libMesaGL.so.2
LD_LIBRARY_PATH= $LS_LIBRARY_PATH_OLD
```

El problema de hacer esto es que el script se tiene que ejecutar como root. Una opción para hacer lo mismo pero sin necesidad de ser root es eliminar la `libMesaGL.so.2` de la ruta de búsqueda del ldconfig y ponerla en otro directorio (por ejemplo en `/usr/local/lib/MesaGL`). Para que nos funcionen los programas que necesitan las Mesa lo único que tenemos que hacer es añadir la linea `export LD_LIBRARY_PATH=/usr/local/lib/MesaGL` en el fichero `~/.bashrc` si usamos el *bash* como shell o la linea `setenv LD_LIBRARY_PATH /usr/local/lib/MesaGL` en el .cshrc/.tcsh si usamos *csh/tcsh*. Haciendo esto tendríamos lo siguiente:

```
[root@unsCAred][~]# cd /usr/local/lib/MesaGL/
[root@unsCAred][/usr/local/lib/MesaGL]# ls -la libMesaGL.so.2
lrwxrwxrwx   1 root  root 23 ene 22 21:19 libMesaGL.so.2 -> /usr/lib/libMesaGL.so.3
```

Ahora ya podemos eliminar las lineas que llevan un *mv* de los scripts `glquake.mesa` y `glquake.3dfx` con lo que ya no es necesario ser root para que funcione.

- No me funciona el ratón: Las `3dfx` y el `gpm` no se llevan bien , en el README viene bastante claro que para que funcione el ratón con las tarjetas aceleradoras hay que matár el gpm antes. Para hacerlo solo que ejecutar un `killall gpm` como root (o un `gpm -k`¿. Tambien se puede hacer mediante la orden `/etc/rc.d/init.d/gpm stop` y reiniciarlo con `/etc/rc.d/inid.d/gpm start` (esto para redhat, en otras distribuciones puede variar ligeramente el directorio `stat.d`); Este sistema es el más recomendable.

7 Servidores españoles de quake.

He aquí una lista de algunos servidores de quake en España.

- QuakeWorld:
  - `GODS` 147.83.7.151
  - `NPM` 194.74.94.4:27500
  - `IBK` 161.72.161.72:27500
  - `Spain` 194.179.72.11
  - `FragZone Stockholm` 194.14.204.119:27500
  - `Axes4All` 194.109.6.220:27500
  - `quake.arrakis.es` 195.5.65.162:27500

  - [LPB] 147.96.20.220:27500

- Quake 2:
  - NPM 194.74.94.9:27910 Deathmatch
  - 194.74.94.9:27920 Capture the flag
  - MaTaDoR-Q2 158.109.8.88:27910 Deathmatch
  - NOKLAN-Q2 194.224.96.59:27910 Deathmatch
  - VAMP-Q2 158.53.204.155:27910 Capture the flag
  - ARRAKIS 195.5.65.162:27910 Deathmatch
  - 195.5.65.165:29000 Capture the flag
  - [QU]-Q2 194.179.71.92:27910 Deathmatch
  - quake.ctv.es 195.57.143.92:27910 Deathmatch

8 Agradecimientos:

A todos los habituales del canal #linux del irc hispano por soportarme, y en especial a aquellos que me han enseñado las maravillas del quake (pukka, wait_man, Al y alguno que me dejo ...), además de a todos los que contribuyen a la expansión del linux.

A Graffic por el 3Dfx-COMOR del que he sacado casi toda la información sobre las 3dfx.

A Satch Sky-Man y Devan por las correcciones y añadidos.

A *Id Software* por hacer el quake, y lo más importante, por portarlo a Linux.

9 Anexo: El INSFLUG

El *INSFLUG* forma parte del grupo internacional *Linux Documentation Project*, encargándose de las traducciones al castellano de los Howtos (Comos), así como la producción de documentos originales en aquellos casos en los que no existe análogo en inglés.

En el **INSFLUG** se orienta preferentemente a la traducción de documentos breves, como los *COMOs* y *PUFs* (**P**reguntas de **U**so **F**recuente, las *FAQs*. :)), etc.

Diríjase a la sede del INSFLUG para más información al respecto.

En la sede del INSFLUG encontrará siempre las **últimas** versiones de las traducciones: www.insflug.org. Asegúrese de comprobar cuál es la última versión disponible en el Insflug antes de bajar un documento de un servidor réplica.

Se proporciona también una lista de los servidores réplica (*mirror*) del Insflug más cercanos a Vd., e información relativa a otros recursos en castellano.

Francisco José Montilla, pacopepe@insflug.org.