Linux Quake HOWTO

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Linux Quake HOWTO

Bob Zimbinski (bobz@mr.net), Brett A. Thomas (quark@baz.com), and Mike Hallock (mikeh@medina.net)

v1.0.1.18, 18 December 1998

This document explains how to install, run and troubleshoot Quake, QuakeWorld and Quake II on an Intel Linux system.

1. Introduction

2. Quake/Quakeworld

3. Quake II

4. Related Software

5. Troubleshooting/FAQs

6. Tips & Tricks

7. Administrivia

---

1. Introduction

Quake, QuakeWorld and Quake II are tremendously popular 3D action games developed by id Software.

If you're not familiar with the Quake games, there are better places than this HOWTO to learn about the basics. See section Other Sources of Information below for a list of some of these better places.

This document assumes you have Linux for the Intel platform up and running, and in some cases the X Window System as well. X is not required to run these games, but it's a nice way to test a basic installation. If you are not running X, you may safely skip over any references to it.

Sections of this document that were updated in the last revision have a ** after the section heading. Sections updated in the revision prior to the last are marked with a ++.

1.1 Feedback,Comments, Corrections

This document certainly does **not** contain everything there is to know about Linux Quake. With your help, though, we can bring it closer to that ideal. We want this HOWTO to be as complete and accurate as possible, so if you notice mistakes or omissions, please bring them to our attention.

Questions, comments, or corrections should be sent to Bob Zimbinski (bobz@mr.net) or Mike Hallock (mikeh@medina.net). Constructive criticism is welcome. Flames are not.

1.2 Other Formats of This Document

This document is available in a variety of different formats. Far more than is necessary, actually. You can find them at http://www.linuxquake.com/howto/other−formats.

- ASCII text
- ASCII text with backspace−overstrikes , suitable for viewing with **man(1)**
- HTML all zipped up and ready to be dropped into your web site
- Postscript
- PRC for viewing on your Palm Pilot with Aportis Doc
- SGML
- LaTeX
- All of the above are also available in a single convenient package

1.3 Acknowledgments

The original version of this document was written by Brett A.Thomas (quark@baz.com) and Mike Hallock (mikeh@medina.net). Bob Zimbinski (bobz@mr.net) rewrote and expanded the original document prior to its initial public release.

Special thanks to the following people for bringing us Quake for Linux:

- John Carmack and the rest of id Software for these great games.
- Dave 'Zoid' Kirsch (zoid@idsoftware.com) for the Linux ports.
- Dave Taylor (ddt@crack.com) for starting this whole wacky Linux port business.
- Daryll Strauss (daryll@harlot.rb.ca.us) for the Linux glide ports.
- Brian Paul (brianp@elastic.avid.com) for the Mesa graphics library.
- David Bucciarelli (tech.hmw@plus.it) for the Mesa/glide driver.

Thanks to the following people for contributions to this Howto:

- Mike Brunson (brunson@l3.net) for the vid_restart tip.
- Joey Hess (joey@kite.ml.org) for info on running svga & gl games from X

- Joe S. (jszabo@eden.rutgers.edu) for a steaming pile of suggestions
- Brad Lambert (bradl@dial.pipex.com) for the `−noudp` reminder.
- agx (gguenthe@iris.rz.uni−konstanz.de) for Quake installation info and pointing out QIPX.
- Derrik Pates (dmp8309@silver.sdsmt.edu) for making me think about security.
- Michael Dwyer (michael_dwyer@mwiworks.com) for the "OS considerations" ideas.
- Derek Simkowiak (dereks@kd−dev.com) for the Quake I CD installation procedure.
- sunstorm (sunstorm@glasscity.net) Quake Mission Pack 2 information.
- Neil Marshall (marshall@pssnet.com) for screen info.
- Serberus (serberus@clarion.co.uk) for Vispatch pointer.
- Andrew Chase (fixy@fixy.org) for the GLQuake "POSSIBLE SCAN CODE ERROR" fix.

1.4 Other Sources of Information

Linux−Specific Quake Information

- LQ:Linux Quake & Utilites http://www.linuxquake.com/.
- Linux Quake Page http://captured.com/threewave/linux/
- QuakeWorld.net http://www.quakeworld.net

General Quake Information

- id Software http://www.idsoftware.com
- PlanetQuake http://www.planetquake.com
- QuakeWorld Central http://qwcentral.stomped.com
- 3Dfx's GL Quake FAQ (somewhat out of date) http://www.3dfx.com/game_dev/quake_faq.html
- Farenheit 176 Console Command Listing http://www.planetquake.com/f176
- rec.games.computer.quake.* newsgroups

Linux Gaming Information ++

- The Linux Game Tome http://www.happypenguin.org/
- LinuxGames.com http://www.linuxgames.com/
- LinuxGames.org http://www.linuxgames.org/

---

2. Quake/Quakeworld

To install Quake on your Linux system, you'll need some flavor of the official Quake distribution from id. This will be either the retail DOS/Windows CD−ROM that you bought at your favorite software store, or the shareware version you downloaded from the net (see below for details on acquiring the shareware version). Alternatively, if you've already got Quake installed on a DOS/Windows machine, you can use the relevant files from that installation.

2.1 Minimum Requirements

You will need, as a bare minimum, the following:

- A Pentium 90 or better (133 recommended) computer

- 16 MB RAM (24 recommended)
- The Quake CD−ROM **or** the shareware version (quake106.zip)
- Linux kernel version 2.0.24 or later
- libc 5.2.18 or later
- One of the following:
    - X11R5 or later (for xquake)
    - SVGAlib 1.2.0 or later (for squake and glquake)
- 30−80 megabytes free disk space (depends on how you install)
- Access to the root account of the machine you're installing on

Optional:

- A supported soundcard
- A 3Dfx VooDoo Graphics or VooDoo2 3D graphics accelerator card.
- Mesa 2.6 or later (for glquake)

2.2 Installing Quake

Download the Necessary Files

All the necessary files for Linux Quake are available at id Software's ftp site, ftp.idsoftware.com. This site can be quite busy at times,so you may want to use one of these mirror sites instead:

- ftp.cdrom.com/pub/idgames/idstuff (California, USA)
- ftp.gamesnet.net/idsoftware (California, USA)
- ftp.linuxquake.com/lqstuff (Ohio, USA)
- ftp.stomped.com/pub/mirror/idstuff (Minnesota, USA)
- mirrors.telepac.pt/pub/idgames (Lisbon, Portugal)
- download.netvision.net.il/pub/mirrors/idsoftware (Haifa, Israel)

The Quake files mentioned in in this section are:

- Shareware Quake for Windows distribution ftp://ftp.idsoftware.com/idstuff/quake/quake106.zip
- X11 Quake binary ftp://ftp.idsoftware.com/idstuff/unsup/unix/quake.x11−1.0−i386−unknown−linux2.0.tar.gz
- SVGAlib Quake binary ftp://ftp.idsoftware.com/idstuff/unsup/squake−1.1−i386−unknown−linux2.0.tar.gz
- OpenGL/Mesa Quake binary ftp://ftp.idsoftware.com/idstuff/unsup/unix/glquake−0.97−i386−unknown−linux2.0.tar.gz
- Linux QuakeWorld clients (X11, SVGAlib and GL clients are all included in each package)
    - libc5 tar.gz package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwcl−2.30−i386−unknown−linux2.0.tar.gz
    - glibc tar.gz package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwcl−2.30−glibc−i386−unknown−linux2.0.tar.gz
    - libc5 rpm package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwcl−2.30−1.i386.rpm
    - glibc rpm package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwcl−2.30−glibc−1.i386.rpm
- Linux QuakeWorld server
    - libc5 tar.gz package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwsv−2.30−i386−unknown−linux2.0.tar.gz

    - ♦ glibc tar.gz package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwsv−2.30−glibc−i386−unknown−linux2.0.tar.gz
    - ♦ libc5 rpm package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwsv−2.30−1.i386.rpm
    - ♦ glibc rpm package ftp://ftp.idsoftware.com/idstuff/quakeworld/unix/qwsv−2.30−glibc−1.i386.rpm
- Capture the Flag client package ftp://ftp.cdrom.com/pub/quake/planetquake/threewave/ctf/client/3wctfc.zip

Other software mentioned:

- lha archive utility ftp://sunsite.unc.edu/pub/Linux/utils/compress/lha−1.00.tar.Z.
- SVGAlib graphics library http://sunsite.unc.edu/pub/Linux/libs/graphics/svgalib−1.3.0.tar.gz
- SVGAlib libc5 binary http://sunsite.unc.edu/pub/Linux/libs/graphics/svgalib−1.3.0.libc5.bin.tar.gz
- Glide runtime libraries http://glide.xxedgexx.com/3DfxRPMS.html

Create the Installation Directory ++

The first thing you'll need to do is decide where you want to install Quake. The "standard" location is `/usr/local/games/quake`. This is where the `.rpm` packages put Quake. If you choose to install somewhere else, please substitute the appropriate path wherever `/usr/local/games/quake` is mentioned.

Note to Redhat users: If you plan on installing QuakeWorld from the `rpm` packages, you should probably install Quake in `/usr/local/games/quake`, since the `rpm`s install to this directory by default.

So go ahead and create the directory you'll install Quake in, and cd to it. The rest of these instructions will assume that this is your current directory.

```
mkdir /usr/local/games/quake
cd /usr/local/games/quake
```

Installing From a Quake CD

If you're installing from a Quake CD−ROM, read on. Otherwise you have permission to skip this section.

There are at least two versions of the Quake CD in circulation. I've got one from the early days that has Quake version 1.01 on it. Iv'e seen other CDs that contain version 1.06. You have 1.01 if you see files on your CD called `quake101.1` and `quake101.2`. If instead you see a file called `resource.1`, you have a newer CD. Mount your Quake CD now and determine which version you've got. In the example below, replace `/dev/cdrom` and `/mnt/cdrom` with the device file and mount point appropriate for your sysem:

```
mount −t iso9660 /dev/cdrom /mnt/cdrom
ls /mnt/cdrom
```

- If you have a `resource.1` file on your CD, you can skip ahead to the next bullet. For a version 1.01 CD, you'll need to download the Quake shareware package to update the `.pak` files after the installation. The last bullet in this section explains this.
    - ♦ Concatenate the two resource files from your CD to a single file on your hard disk:

      ```
      cat /mnt/cdrom/quake101.1 /mnt/cdrom/quake101.2 > resource.1
      ```

    - ♦ Now continue on to the next paragraph, but when I refer to `/mnt/cdrom/resource.1`, you should use `/usr/local/games/quake/resource.1` instead.

- Now it's time to extract the Quake files. The `resource.1` file on your CD is really an lha archive (lha is a file compression and archiving format like zip or tar). We'll use the `lha(1)` command to extract it. If lha is not already installed on your system, you can get it from ftp://sunsite.unc.edu/pub/Linux/utils/compress/lha−1.00.tar.Z.

  ```
  lha e /mnt/cdrom/resource.1
  ```

  When lha is done, your Quake directory will contain a bunch of new files. A directory called `id1/` will also be created. The files in this directory are the only ones that are important for Linux Quake, so you can safely remove everything else. If you're totally new to Quake, or even if you're not, you may want to hang on to the `*.txt` files. On my system, I throw all the readmes that accumulate into a `doc/` directory. So:

  ```
  cd /usr/local/games/quake
  mkdir doc
  mv *.txt doc
  rm −f *
  ```
- If you installed from a version 1.01 CD, now you need to overwrite your `id1/pak0.pak` file with the one from the Quake shareware version. Install the shareware version as described in [Shareware Version Install](), only install it in some temporary directory so you don't overwrite your real Quake files. When you've extracted all the shareware files, copy the `id1/pak0.pak` file from the temporary shareware directory to your `/usr/local/games/quake/id1` directory. After that, you can erase the temporary shareware files.

That's it for installing from the CD. You can jump ahead to section " [Installing the Linux Binaries]()" now.

DOS/Windows to Linux Install

If you have Quake installed under Windows or DOS on a different machine, you can transfer the files in `quake\id1\` to your Linux system via FTP or some other mechanism. Keep in mind that the filenames on your Linux system must be in lower case for Quake to find them, so you may have to rename them after the transfer. Also note that it may be necessary to delete your DOS/Win installation after you do this to remain in compliance with the terms of id's software license. It's not my fault if you do something illegal.

If your DOS/Win and Linux systems are on the same machine, you have two options: copy the files from your DOS/Windows partition to your Linux partition, or link to the necessary files from Linux. Both options will work equally well. You just save around 50 megabytes of disk space when you link instead of copy.

Whatever you choose to do, start by cd'ing to your Quake directory and creating a new directory below it called `id1`:

```
cd /usr/local/games/quake
mkdir id1
```

- If you want to copy the files from your DOS/Windows partition, do something like this:

  ```
  cp /win95/games/quake/id1/*.pak id1
  ```
- To create links to your DOS/Windows Quake files instead, do this:

```
                cd id1
                ln −s /win95/games/quake/id1/*.pak .
```

Replace `/win95/games/quake` with the correct path to your DOS/Windows partition and Quake installation directory.

The Quake data files are now installed. Move ahead to " Installing the Linux Binaries".

Shareware Version Install

The single−episode shareware version of Quake is freely available for download from id's ftp site. It has all the features of the full version, with a couple of major limitations: You can't play QuakeWorld with it, and you can't play custom or modified levels.

Installing the shareware version of Quake isn't really much different than installing from the CD.

See section Download the Necessary Files for the location of the shareware distribution. Download it and extract it to your Quake directory:

```
cd /usr/local/games/quake
unzip −L /wherever/you/put/it/quake106.zip
```

Now you've got (among others) a file called `resource.1` that's really an lha archive (lha is a file compression and archiving format like zip or tar). We'll use the **lha(1)** command to extract it. If lha is not already installed on your system, you can get it from ftp://sunsite.unc.edu/pub/Linux/utils/compress/lha−1.00.tar.Z.

```
lha e resource.1
```

When lha is done, your Quake directory will contain a bunch of new files. A directory called `id1/` will also be created. The files in this directory are the only ones that are important for Linux Quake, so you can safely remove everything else. If you're totally new to Quake, or even if you're not, you may want to hang on to the `*.txt` files. On my system, I throw all the readmes that accumulate into a `doc/` subdirectory. So:

```
cd /usr/local/games/quake
mkdir doc
mv *.txt doc
rm −f *
```

Now you're ready to install the Linux binaries.

2.3 Adding the Linux Binaries

Decide which of the three flavors of Quake you'd like to install:

- X11 Quake allows you to run Quake in a window on your X desktop. It's the least exciting client, but it's a great, safe way to test your installation.
- Squake is the SVGAlib Quake client. It runs full screen on your console.

- GLQuake is the OpenGL Quake client, the One True Way to play Quake if you have a 3Dfx accellerator card.

Download the packages you want (see section Download the Necessary Files) and extract them to your Quake directory like so:

```
cd /usr/local/games/quake
tar −xzf XXXX−i386−unknown−linux2.0.tar.gz
```

2.4 Setting Permissions

Quake and QuakeWorld servers can be run by any user. The Quake clients, however, need access to your sound and graphics cards, which requires privileges that normal users don't have. One (bad) way to deal with this is to always run Quake as root. Responsible system administrators will cringe at this filthy suggestion. Making the Quake binaries setuid root is a more acceptable solution. Quake can then be run by regular users and still have the privileges it needs to access the sound and graphics devices. Setuid presents a security risk, though. A clever user could exploit a bug or security hole in Quake to gain root access to your system. Of course, if you don't run a multi−user system, this is may not be a big concern.

`squake` is the only Quake client that *must* be run with root permissions. With a little work, you can run the X and GL clients without setuid. Running X and GL games without setuid in the Tips and Tricks section tells how to make this work.

If you plan to run `squake`, make it setuid root with the following commands:

```
chown root squake
chmod 4755 squake
```

If you've decided it's ok to run `quake.x11` and `glquake` setuid root on your system, you can repeat the above commands for these binaries as well.

2.5 X11 Quake

If you installed the X11 client, now's a good time to try it out. You may need to do further configuration for `glquake` and `squake`, but at this point `quake.x11` should be ready to go.

```
cd /usr/local/games/quake
./quake.x11
```

If all is well, a small Quake window should appear with the first demo running in it. You should hear sound effects and possibly music, if your CD is in the drive. If any of this fails to occur, please see the Troubleshooting section for help.

2.6 SVGAlib Quake

Both squake and glquake require SVGAlib to run (glquake uses SVGAlib to process keyboard and mouse input, in case you're wondering). SVGAlib comes with most modern Linux distributions, and **must** be properly configured before squake or glquake will run correctly.

libvga.config is SVGAlib's configuration file. On most systems you'll find it in either /etc or /etc/vga. Make sure the mouse, monitor, and video card settings in this file are correct for your system. See the SVGAlib documentation for more details.

If you don't already have SVGAlib on your system, download it from the location mentioned in the files section above.

If you have a RedHat 5.x or other glibc−based Linux distribution, see Glibc, RedHat 5.x, Debian 2 considerations in the Troubleshooting/FAQs section for important information about compiling libraries for use with Quake. A precompiled libc5 SVGAlib binary is available at http://sunsite.unc.edu/pub/Linux/libs/graphics/svgalib−1.3.0.libc5.bin.tar.gz for those who don't want to deal with the hassle of compiling for libc5.

You should run squake from a virtual console. It won't run from X unless you're root when you start it. And running a game as the root user is something that should be avoided. So if you're in X, do a CTRL+ALT+F1, login and then:

```
            cd /usr/local/games/quake
            ./squake
```

Running SVGA and GL games from X in the Tips & Tricks section below explains how to launch SVGA and GL Quake from X without manually switching to a virtual console.

2.7 GLQuake

Hardware−accelerated OpenGL Quake is Quake the way God intended it to be. There is no substitute, and once you've experienced it there's no going back.

To run glquake, you need a 3D card with the Voodoo, Voodoo2 or Voodoo Rush graphics chipset on it. There are specific issues to be dealt with if you have a Voodoo Rush card, and I won't go into them now because frankly, I wouldn't know what I was talking about. A future version of this HOWTO will cover Rush issues (If somebody wants to write about Voodoo Rush issues, I'll gladly include it here).

The SVGAlib, Glide, and Mesa libraries must all be installed and configured properly on your system for glquake to work. The following sections will very briefly cover what you need to do to get them going.

Bernd Kreimeier's (bk@gamers.org) Linux 3Dfx HOWTO (http://www.gamers.org/dEngine/xf3D/howto/3Dfx−HOWTO.html) is good source for further information.

The 3dfx.glide.linux newsgroup on the 3dfx news server (news.3dfx.com) is another good source of information about the intersection of Linux, glide, Mesa and Quake.

SVGAlib

glquake uses SVGAlib to get input from the mouse and keyboard, so you'll need to configure it as outlined in section SVGAlib Quake.

Glide ++

Glide is a library that provides an API for programming 3Dfx based cards. If you want the Mesa graphics library to use your 3Dfx card, you've gotta have it.

Do **NOT** use the Linux Glide library distributed at 3Dfx's web site. It's more than a year out of date. You're bound to have problems if you try to use it. The latest version of glide can always be found at http://glide.xxedgexx.com/3DfxRPMS.html. Select the package(s) appropriate for your system, and install according to the instructions on the web page.

Note that unless you download the 3Dfx device driver package in addition to the Glide library, you will only be able to run Glide applications (like GLQuake) as root. Install the `/dev/3dfx` module and you can play GLQuake as a regular user. PentiumPro/Pentium II users have an additional incentive for downloading this driver: it can dramatically increase your framerate. The driver enables support for MTRRs, a memory−caching feature of these CPUs. See http://glide.xxedgexx.com/MTRR.html for a better explanation of this feature.

Once you have glide installed, try out the test program that comes with it. Remember this program: it's a good way to reset your display if you ever have a glide application (like GLQuake) crash and leave your screen switched off. **NOTE: run this test from a VC, *not* X!** It's possible for the test app to lose mouse and keyboard focus in X, and then you'll have no way of shutting it down.

```
            /usr/local/glide/bin/test3Dfx
```

Your screen should turn blue and prompt you to hit any key. After you press a key you should be returned to the prompt. 3dfx.glide.linux on 3dfx's news server (news.3dfx.com) is a great source of information for Linux glide−specific problems.

Mesa

Once glide's installed, you need to install Mesa, a free OpenGL implementation by Brian Paul (brianp@elastic.avid.com). Luckily, you won't have to look far, because Mesa 2.6 is included with the QLQuake & QuakeWorld binaries. All you have to do is move it to the right place:

```
            cd /usr/local/games/quake
            cp libMesaGL.so.2.6 /usr/local/lib
            ldconfig
```

The RedHat distribution comes with a (IMO) broken `ld.so` configuration. `/usr/local/lib` is not part of `ld.so`'s search path by default, so anything you install there won't get used. You can remedy the situation by adding the line `/usr/local/lib` to your `/etc/ld.so.conf` file, or including `/usr/local/lib` in your `$LD_LIBRARY_PATH`. Alternatively, you could install all new libraries someplace like `/lib`, but this approach seriously offends my tidy nature.

If you want to upgrade Mesa to a more recent version (Mesa 3.0 is the most recent version as of this writing), you can download the latest from ftp://iris.ssec.wisc.edu/pub/Mesa If you have a RedHat 5.x or other glibc−based Linux distribution, see Glibc, RedHat 5.x, Debian 2 considerations in the Troubleshooting/FAQs section for important information about compiling libraries for Quake.

After you've built it according to the instructions, you will have to do two things:

- Remove your old Mesa installation. If you previously installed a `libMesaGL.so.2.6` as described above, you must remove it or Quake may not use the new version.

  ```
  cd /usr/local/lib/
  rm −f libMesaGL.so.2*
  ```
- If the new Mesa has a major version number that's greater than 2, you need to create a link to it with the name `libMesaGL.so.2`:

  ```
  cd /usr/local/lib/
  ln −s /wherever/you/installed/it/libMesaGL.so.3.0 libMesaGL.so.2
  ldconfig
  ```

Now switch to a VC (CTRL+ALT+F1) and start glquake.

```
cd /usr/local/games/quake
./glquake
```

2.8 Linux−Specific Command Line Options

This section covers command line options that are specific to the Linux versions of Quake. There are plenty of other Quake options, but they're beyond the scope of this HOWTO. Check out some of the sites listed in section General Quake Information for this kind of information.

−mem num

Specify memory in megabytes to allocate (default is 8MB, which should be fine for most needs).

−nostdout

Don't do any output to stdout. Use this if you don't want all the console output dumped to your terminal.

−mdev device

Mouse device, default is `/dev/mouse`

−mrate speed

Mouse baud rate, default is 1200

−cddev device

CD device, default is `/dev/cdrom`

−mode num

Use indicated video mode (squake only)

−nokdb

Don't initialize keyboard

−sndbits 8 or 16

Set sound bit sample size. Default is 16 if supported.

−sndspeed speed

Set sound speed. Usual values are 8000, 11025, 22051 and 44100. Default is 11025.

−sndmono

Set mono sound

−sndstereo

Set stereo sound (default if supported)

2.9 QuakeWorld

http://www.quakeworld.net says it better than I could:

> QuakeWorld is an Internet multi−player specific version of Quake. While the original version of Quake can be played over the Internet, modem users − the majority of players, had less than satisfactory play. Symptoms like excessive lag − actions actually happening much later than you did them; packet loss − the game would freeze and resume several seconds later; and various other difficulties plagued users. After realizing how many people played Quake on the internet, and how many wanted to, but couldn't due to the play being unsatisfactory, John Carmack of id Software decided to create a version of Quake that was optimized for the average modem Internet player. This Internet specific version does only 1 thing, play deathmatch games over a TCP/IP network such as the Internet. It has no support for solo play, and you can't do anything with out connecting to a special server.

You need the full, registered or retail version of Quake to play QuakeWorld, and a Linux QuakeWorld client. QuakeWorld clients come in the same flavors (X11, SVGAlib and Mesa) as normal Quake, but they're all bundled together in one package, so you only need to download one file. However, you've got four packages to choose from:

- a libc5 tar.gz package
- a glibc tar.gz package
- a libc5 rpm package
- a glibc rpm package

Install just one of these packages. Each contains the same files, they're just linked against different libraries. Redhat 5.x users should choose the the glibc rpm package. Users of glibc based systems without rpm support should use the glibc tar package. The libc5 rpm is for Redhat distributions prior to 5.0 and other distributions that use the rpm package format. The libc5 tar.gz package is for Slackware and everyone else.

See the Download the Necessary Files section for the location of the Linux Quakeworld files.

The prerequisites and configuration for these binaries are the same as for Quake, so refer to the previous sections for help on setting up SVGAlib or glide/Mesa.

Installing the RPM packages

Installation of the rpm packages should be as simple as:

```
su root
rpm −Uvh qwcl−xxxxx.i386.rpm
```

`qwcl`, `glqwcl` and `glqwcl.glx` will be installed setuid root so that they can access the graphics devices on your system. The X and GL clients can be run without root privileges if you follow the instructions in Running X and GL games without setuid below.

Rpm may complain that it can't find `libglide2x.so`. The Glide library is only necessary if you have a 3Dfx card and want to run QuakeWorld in GL mode (`glqwcl`). If you don't plan to use the GL mode, you can override the glide dependency with the `−−nodeps` option:

```
su root
rpm −Uvh qwcl−xxxxx.i386.rpm −−nodeps
```

Installing the tar.gz packages

To install, just untar the file in your Quake directory. Do it as root so the proper file permissions get set:

```
cd /usr/local/games/quake
su root
tar −xzf qwcl2.21−i386−unknown−linux2.0.tar.gz
```

`qwcl`, `glqwcl` and `glqwcl.glx` will be installed setuid root so that they can access the graphics devices on your system. The GL and X clients can be run without root privileges if you follow the instructions in Running X and GL games without setuid below.

Running QuakeWorld

Once QuakeWorld is installed alongside your Quake files, you can start it up like:

```
./qwcl +connect some.server.address
```

See section Related Software for info about some nifty front ends for QuakeWorld that make finding servers easy.

- **lib3dfxgl.so**

  With Quakeworld version 2.30, an alternative to the Mesa library is available. `lib3dfxgl.so` is a mini−GL driver optimized for Quake that provides slightly better framerates than Mesa. This is a port of a driver that 3Dfx developed for Quake under Windows, and apparently not all of its features work properly yet. So hopefully we can expect its performance to improve with time.

Like Mesa, `lib3dfxgl.so` requires Glide in order to access your 3Dfx card. The Quakeworld packages come with a script, `glqwcl.3dfxgl` for running Quakeworld with this library on glibc systems. The next paragraph explains how to run Quakeworld with `lib3dfxgl.so` on a libc5 system. On a glibc system in order for this script to work,the `glqwcl` executable **must not be setuid**, nor should you run it as root. `glqwcl` will silently load Mesa rather than `lib3dfxgl.so` if it runs with root permissions. This non−root requirement implies that you have the `/dev/3dfx` driver installed.

On a libc5 system, you need to create a symbolic link to `lib3dfxgl.so` called `libMesaGL.so.2` like so:

```
        cd /usr/local/games/quake
        ln −sf lib3dfxgl.so libMesaGL.so.2
```

Then start Quakeworld from a script that tells `$LD_LIBRARY_PATH` to look in the current directory:

```
    #!/bin/sh

    LD_LIBRARY_PATH=".:$LD_LIBRARY_PATH" ./glqwcl $*
```

You can tell which driver is being loaded by looking for output like the following in your console as Quakeworld starts:

```
        GL_VENDOR: 3Dfx Interactive Inc.
        GL_RENDERER: 3Dfx Interactive Voodoo^2(tm)
        GL_VERSION: 1.1
        GL_EXTENSIONS: 3DFX_set_global_palette WGL_EXT_swap_control GL_EXT_paletted_tex
```

If `GL_VENDOR` says `Brian Paul` rather than `3Dfx Interactive Inc.`, that means Mesa is still being used rather than the miniport dirver.

- **glqwcl.glx** ++

  `glqwcl.glx` is linked against standard OpenGL libraries instead of Mesa. This allows Quakeworld to run on other 3D hardware that is supported by other OpenGL implementations. At this time, I dont' know of any OpenGL implementations that support hardware other than 3Dfx, but this renderer ensures that when they appear, we'll be able to play Quakeworld with them.

  Use of the GLX interface also removes GL Quake II's dependency on SVGAlib for keyboard and mouse input.

  This is a GLX application, and as such, must be run from X.

  You can use this client with Mesa/3Dfx if you install Mesa and Glide as explained in the previous section, then set the `$MESA_GLX_FX` environment variable to "fullscreen" before you run `quake2`:

```
        export MESA_GLX_FX=fullscreen
```

```
./glqwcl.glx +_windowed_mouse 1
```

Why the `+_windowed_mouse 1` option? Remember that this is an X application which happens to use your 3Dfx card. Even though the display takes up your entire screen, Quakeworld is stil running in a window. This means that if you're not *very* careful, you could move the mouse pointer outside the Quakeworld window, and Quakeworld will suddenly stop responding to mouse and keyboard input. `+_windowed_mouse 1` avoids this problem by telling `glqwcl.glx` to grab the mouse and not let it move outside its window.

2.10 Servers

Most, if not all, existing information about running a DOS/Windows QW server is equally applicable to running a Linux server.

To start a QuakeWorld server, simply do:

```
./qwsv
```

The official QuakeWorld server manual lives at http://qwcentral.stomped.com.

2.11 Mods & Addons

One of the very cool things about the Quake games is that the authors made them easily extensible. End users can create their own levels, add new weapons or monsters, or even completely change the rules of the game.

Capture the Flag

This is my favorite variation of both Quake and Quake 2. Instead of just running around and killing everyone you meet (which definitely has its merits, don't get me wrong!), CTF is team−based and more strategic. Dave 'Zoid' Kirsch, also the maintainer of the Linux Quake ports, created this mod.

Everything you need to know about CTF can be found at http://captured.com/threewave/ You need ftp://ftp.cdrom.com/pub/quake/planetquake/threewave/ctf/client/3wctfc.zip to play. To install the client piece of CTF, simply create a directory called `ctf` in your Quake directory and unzip the `3wctfc.zip` file there. The accompanying `readme.txt` file is chock full of good information.

```
cd /usr/local/games/quake
mkdir ctf
cd ctf
unzip −L /wherever/you/put/it/3wctfc.zip
```

For information on running a CTF server, see the web pages mentioned above.

Mission Packs

Activision released two add−on packs of extra levels for Quake, **Scourge of Armagon** and **Dissolution of Eternity**.

- **Mission Pack 1: The Scourge of Armagon** Assuming your CD is mounted on `/mnt/cdrom` and Quake is installed in `/usr/local/games/quake`:

  ```
              cd /usr/local/games/quake
              mkdir hipnotic
              cp /mnt/cdrom/hipnotic/pak0.pak hipnotic
              cp /mnt/cdrom/hipnotic/config.cfg hipnotic
  ```

  Play the mission pack like this:

  ```
              cd /usr/local/games/quake
              ./quake.x11 −game hipnotic
  ```
- **Mission Pack 2: Dissolution of Eternity** Installing the second mission pack is pretty much the same procedure as installing the first. Follow the directions for Mission Pack 1, but replace the word *hipnotic* with the word *rogue*, and skip the `config.cfg` step, as this file isn't included on the Mission Pack 2 CD.

Quake Tools

Anybody care to contribute some info about qcc, bsp and all that?

3. Quake II

To install Quake II on your Linux system, you'll need some flavor of the official Quake II distribution from id. This will be either the retail Windows CD−ROM that you bought at your favorite software store, or the demo version you downloaded from the net. See Download the Necessary Files for details on acquiring the demo version. Alternatively, if you've already got Quake installed on a Windows machine, you can use the relevant files from that installation.

3.1 Prerequisites

You will need, as a bare minimum, the following:

- A Pentium 90 or better (133 recommended) computer
- 16 MB RAM (24 recommended)
- The Quake 2 CD−ROM **or** the demo version (q2−314−demo−x86.exe)
- Linux kernel version 2.0.24 or later
- libc 5.2.18 or later
- One of the following:
  - ♦ X11 server that supports the MITSM shared memory extension. 8 and 16 bit displays are supported. (for X renderer)
  - ♦ SVGAlib 1.2.10 or later (for SVGA and GL renderer)
- 25−400 megabytes free disk space (depending on how you install)
- Access to the root account of the machine you're installing on

Optional:

- A supported soundcard
- A 3Dfx Voodoo Graphics or Voodoo2 or Voodoo Rush 3D graphics accelerator card.

- 3Dfx glide libraries installed (for GL renderer)
- Mesa 2.6 or later (for GL renderer)

3.2 Installing Quake II

Download the Necessary Files++

All the necessary files for Linux Quake II are available at id Software's ftp site, ftp.idsoftware.com. This site can be quite busy at times,so you may want to use one of these mirror sites instead:

- ftp.cdrom.com/pub/idgames/idstuff (California, USA)
- ftp.gamesnet.net/idsoftware (California, USA)
- ftp.stomped.com/pub/mirror/idstuff (Minnesota, USA)
- mirrors.telepac.pt/pub/idgames (Lisbon, Portugal)
- download.netvision.net.il/pub/mirrors/idsoftware (Haifa, Israel)

The Quake II files mentioned in in this section are:

- Quake II Linux Binaries
    - libc5 tar.gz package ftp.idsoftware.com/idstuff/quake2/unix/quake2−3.20−i386−unknown−linux2.0.tar.gz
    - glibc tar.gz package ftp.idsoftware.com/idstuff/quake2/unix/quake2−3.20−glibc−i386−unknown−linux2.0.tar.gz
    - libc5 rpm package ftp.idsoftware.com/idstuff/quake2/unix/quake2−3.20−6.i386.rpm
    - glibc rpm package ftp.idsoftware.com/idstuff/quake2/unix/quake2−3.20−glibc−6.i386.rpm
- Quake II Demo Version for Windows ftp://ftp.idsoftware.com/idstuff/quake2/q2−314−demo−x86.exe
- Quake II Game Source ftp://ftp.idsoftware.com/idstuff/quake2/source/q2src320.shar.Z
- Xatrix (Mission Pack 1) Game Source ftp://ftp.idsoftware.com/idstuff/quake2/source/xatrixsrc320.shar.Z
- Rogue (Mission Pack 2) Game Source ftp://ftp.idsoftware.com/idstuff/quake2/source/roguesrc320.shar.Z
- Quake II Capture the Flag ftp://ftp.idsoftware.com/idstuff/quake2/ctf/q2ctf102.zip
- Quake II Capture the Flag Game Source ftp://ftp.idsoftware.com/idstuff/quake2/ctf/source/q2ctf−1.02−source.shar.Z

Other software mentioned:

- SVGAlib graphics library http://sunsite.unc.edu/pub/Linux/libs/graphics/svgalib−1.3.0.tar.gz
- SVGAlib libc5 binary http://sunsite.unc.edu/pub/Linux/libs/graphics/svgalib−1.3.0.libc5.bin.tar.gz
- Glide runtime libraries http://glide.xxedgexx.com/3DfxRPMS.html
- Mesa 3D graphics library http://www.ssec.wisc.edu/~brianp/Mesa.html
- unzip archive utility http://sunsite.unc.edu/pub/Linux/utils/compress/unzip−5.31.tar.gz

Create the Installation Directory ++

The first thing you'll need to do is decide where you want to install Quake II. The "standard" location is `/usr/local/games/quake2`. This is where the `.rpm` packages put Quake II. If you choose to install somewhere else, please substitute the appropriate path wherever `/usr/local/games/quake2` is mentioned.

So go ahead and create the directory you'll install Quake II in, and cd to it. The rest of these instructions will assume that this is your current directory.

```
mkdir /usr/local/games/quake2
cd /usr/local/games/quake2
```

Installing from CD

Place your Quake II CD in your CD ROM drive, and mount it:

```
mount −t iso9660 /dev/cdrom /mnt/cdrom
```

If your CDROM is typically mounted elsewhere, substitute its location for `/mnt/cdrom`. If you're not sure where your CDROM is mounted, please see the documentation for your particular distribution.

- **Full Install** The simplest install method is a "full" install, which involves copying the entire contents of your CD to your hard drive. This requires about 350 MB, and is accomplished by issuing the following commands:

  ```
  cd /usr/local/games/quake2
  cp −r /mnt/cdrom/Install/Data/* .
  ```

  There'll be a bunch of unnecessary Windows files hanging around that you can safely delete:

  ```
  rm −f /usr/local/quake2/*.dll
  rm −f /usr/local/quake2/quake2.exe
  rm −f /usr/local/quake2/baseq2/gamex386.dll
  ```
- **Medium Install** If 450 MB is too much space for you to devote to Quake II, you can skip installing the movie cutscenes and link to them on the CD−ROM instead. This will reduce the space requirements to about 200 MB:

  ```
  cd /usr/local/games/quake2
  mkdir baseq2
  cp /mnt/cdrom/Install/Data/baseq2/pak0.pak baseq2
  cp −r /mnt/cdrom/Install/Data/baseq2/players baseq2
  ln −s /mnt/cdrom/Install/Data/baseq2/video baseq2/video
  ```

  Note that this doesn't mean you have to mount your Quake II CD every time to want to play Quake II. If the game can't load the videos, it just won't display them.

Windows to Linux install

If you have Quake II installed under Windows on a different machine, you can transfer the files in `quake2\baseq2\` to your Linux system via FTP or some other mechanism. Keep in mind that the filenames on your Linux system must be in lower case for Quake II to find them, so you may have to rename them after the transfer. Also note that it may be necessary to delete your Windows installation after you do this to remain in compliance with the terms of id's software license. It's not my fault if you do something illegal.

If your Windows and Linux systems are on the same machine, you have two options: copy the files from your

Windows partition to your Linux partition, or link to the necessary files from Linux. Both options will work equally well. You'll just save a lot of disk space when you link instead of copy. As usual, replace `/win95/games/quake2` in the following examples with the correct path to your Windows partition and Quake II installation.

- If you want to copy the files from your Windows partition, do something like this:

  ```
  cd /usr/local/games/quake2
  cp −r /win95/games/quake2/baseq2 .
  ```

- To create links to your Windows Quake II files instead, do this:

  ```
  cd /usr/local/games/quake2
  ln −s /win95/games/quake2/baseq2 .
  ```

  This second method requires that the Windows partition you're linking to be writeable by users, which may not be appropriate for all systems. By making your Windows partition writeable, you are giving all users the opportunity to destroy your entire Windows installation. If that's ok with you, modify your `/etc/fstab` to mount the Windows partition with the options *umask=002,gid=XXX*, where XXX is the group id number of the "users" group. Look in `/etc/group` for this information. When `fstab` is updated, umount and re−mount the Windows partition and you're done.

You're done installing the Quake II data files. Move ahead to " Installing the Linux Binaries".

Installing the demo version

id Software has a freely available demo version of Quake II at their ftp site. It's a 40 megabyte download. The demo includes all features of the full version, including multiplayer, but it only comes with three levels, so it may be difficult to find a server to play on.

See the section Download the Necessary Files above for the location of the Quake II demo. Download it and place it in your Quake II directory.

The demo distribution is a self−extracting zip file (it's self−extracting in other OS's anyway). You can extract it with the **unzip(1)** command, which should be included in most modern distributions. If you don't have unzip, you can download it from the location listed in the Download the Necessary Files section.

cd to your Quake II directory and extract the archive:

```
cd /usr/local/games/quake2
unzip q2−314−demo−x86.exe
```

Now we've got to delete some things and move some other things around:

```
rm −rf Splash Setup.exe
mv Install/Data/baseq2 .
mv Install/Data/DOCS docs
rm −rf Install
rm −f baseq2/gamex86.dll
```

The Quake II demo is now installed. You just need to add the Linux binaries.

3.3 Adding the Linux Binaries

There are four Linux Quake II packages available for download:

- a libc5 tar.gz package
- a glibc tar.gz package
- a libc5 rpm package
- a glibc rpm package

Install just one of these packages. Each contains the same files, they're just linked against different libraries. Redhat 5.x users should choose the the glibc rpm package. Users of glibc based systems without rpm support should use the glibc tar package. The libc5 rpm is for Redhat distributions prior to 5.0 and other distributions that use the rpm package format. The libc5 tar.gz package is for Slackware and everyone else.

See the Download the Necessary Files section for the location of the Linux Quake II files.

Installing the RPM packages

Installation of the rpm packages should be as simple as:

```
su root
rpm −Uvh quake2−xxxxx.i386.rpm
```

Rpm may complain that it can't find `libglide2x.so`. The Glide library is only necessary if you have a 3Dfx card and want to run Quake II in GL mode. If you don't plan to use the GL mode, you can override the glide dependency with the `−−nodeps` option:

```
su root
rpm −Uvh quake2−xxxxx.i386.rpm −−nodeps
```

Installing the tar.gz packages

To install, just untar the file in your Quake II directory. Do it as root so the proper file permissions get set:

```
cd /usr/local/games/quake
su root
tar −xzf qwcl2.21−i386−unknown−linux2.0.tar.gz
```

3.4 Setting Permissions

If you ran `rpm` or `tar` as root when installing the Quake II package on your system, the file permissions should be properly set already. The `quake2` executable was installed setuid root so that it can access the graphics devices on your system. For security, the `ref_*.so` rendering libraries are owned by root and writeable only by him. If root doesn't own the libraries, or they're world writeable, `quake2` will refuse to run.

If you plan to only run Quake II with the GL or X renderers, your `quake2` doesn't need to be setuid root. See Running X and GL games without setuid in the Tips and Tricks section below for information on running

Quake II without root permissions.

Quake2.conf

For security reasons, there is a quake2.conf file, which tells Quake II where to find the rendering libraries (ref_*.so) it needs. It contains only one line, which should be the path to your Quake II installation. Quake II looks for this file in /etc. If you installed Quake II from an .rpm file, this file was installed for you. If you installed from a .tar package, you need to create it like so:

```
su root
cd /usr/local/games/quake2
pwd > /etc/quake2.conf
chmod 644 /etc/quake2.conf
```

3.5 The X Renderer

Quake II should be ready to run under X now. Give it a try:

```
cd /usr/local/games/quake2
./quake2 +set vid_ref softx
```

If all is well, after a pretty significant pause, a small Quake II window will appear with the first demo running in it. You should hear sound effects and possibly music, if your CD is in the drive. If any of this fails to occur, please see section Troubleshooting for help.

3.6 The SVGAlib Renderer ++

You need SVGAlib installed and configured if you're going to use either the ref_soft or ref_gl renderers. (Quake II uses SVGAlib to process keyboard and mouse input, in case you're wondering why you'd need it for the GL renderer). SVGAlib comes with most modern distributions, and **must** be properly configured before Quake II will run correctly outside of X.

libvga.config is SVGAlib's configuration file. On most systems you'll find it in either /etc or /etc/vga. Make sure the mouse, monitor, and video card settings in this file are correct for your system. See the SVGAlib documentation for more details.

If you don't already have SVGAlib on your system, download it from the location mentioned in the files section above.

You should run Quake II from a virtual console when using the ref_soft or ref_gl renderers. It won't run from X unless you're root when you start it, and that's not advisable. So if you're in X, do a CTRL+ALT+F1, login and then:

```
cd /usr/local/games/quake2
./quake2 +set vid_ref soft
```

Running SVGA and GL games from X in the Tips & Tricks section below explains how to launch SVGA and GL Quake II from X without manually switching to a virtual console.

3.7 The OpenGL Renderer

Hardware−accelerated OpenGL Quake is Quake the way God intended it to be. There is no substitute, and once you've experienced it there's no going back.

To run Quake II in GL mode, you need a 3D card with the Voodoo, Voodoo2 or Voodoo Rush graphics chipset on it. There are specific issues to be dealt with if you have a Voodoo Rush card, and I won't go into them now because frankly, I wouldn't know what I was talking about. A future version of this HOWTO will cover Rush issues (If somebody wants to write about Voodoo Rush issues, I'll gladly include it here).

The SVGAlib, Glide, and Mesa libraries must all be installed and configured properly on your system for `quake2` to work. The following sections will very briefly cover what you need to do to get them going.

Bernd Kreimeier's (bk@gamers.org) Linux 3Dfx HOWTO (http://www.gamers.org/dEngine/xf3D/howto/3Dfx−HOWTO.html) is good source for further information.

The 3dfx.glide.linux newsgroup on the 3dfx news server (news.3dfx.com) is another good source of information about the intersection of Linux, glide, Mesa and Quake.

SVGAlib

Quake II uses SVGAlib to get input from the mouse and keyboard, so you'll need to configure it as outlined in section SVGAlib Renderer.

Glide ++

Glide is a library that provides an API for programming 3Dfx based cards. If you want the Mesa graphics library to use your 3Dfx card, you've gotta have it.

Do **NOT** use the Linux Glide library distributed at 3Dfx's web site. It's more than a year out of date. You're bound to have problems if you try to use it. The latest version of glide can always be found at http://glide.xxedgexx.com/3DfxRPMS.html. Select the package(s) appropriate for your system, and install according to the instructions on the web page.

Note that unless you download the 3Dfx device driver package in addition to the Glide library, you will only be able to run Glide applications (like Quake II) as root. Install the `/dev/3dfx` module and you can play Quake II as a regular user. PentiumPro/Pentium II users have an additional incentive for downloading this driver: it can dramatically increase your framerate. The driver enables support for MTRRs, a memory−caching feature of these CPUs. See http://glide.xxedgexx.com/MTRR.html for a better explanation of this feature.

Once you have glide installed, try out the test program that comes with it. Remember this program: it's a good way to reset your display if you ever have a glide application (like Quake II) crash and leave your screen switched off. **NOTE: run this test from a VC, *not* X!** It's possible for the test app to lose mouse and keyboard focus in X, and then you'll have no way of shutting it down.

```
        /usr/local/glide/bin/test3Dfx
```

Your screen should turn blue and prompt you to hit any key. After you press a key you should be returned to the prompt. 3dfx.glide.linux on 3dfx's news server (news.3dfx.com) is a great source of information for Linux glide−specific problems.

Mesa **

Once glide's installed, you need to install Mesa, a free OpenGL implementation by Brian Paul (brianp@elastic.avid.com). Luckily, you won't have to look far, because Mesa 2.6 is included with the Quake II binaries. All you have to do is move it to the right place:

```
cd /usr/local/games/quake2
cp libMesaGL.so.2.6 /usr/local/lib
ldconfig
```

The RedHat distribution comes with a (IMO) broken `ld.so` configuration. `/usr/local/lib` is not part of `ld.so`'s search path by default, so anything you install there won't get used. You can remedy the situation by adding the line `/usr/local/lib` to your `/etc/ld.so.conf` file, or including `/usr/local/lib` in your `$LD_LIBRARY_PATH`. Alternatively, you could install all new libraries someplace like `/lib`, but this approach seriously offends my tidy nature.

If you want to upgrade Mesa to a more recent version (Mesa 3.0 is the most recent version as of this writing), you can download the latest from ftp://iris.ssec.wisc.edu/pub/Mesa If you have a RedHat 5.x or other glibc−based Linux distribution, see Glibc, RedHat 5.x, Debian 2 considerations in the Troubleshooting/FAQs section for important information about compiling libraries for Quake.

After you've built it according to the instructions, you will have to do two things:

- Remove your old Mesa installation. If you previously installed a `libMesaGL.so.2.6` as described above, you must remove it or Quake II may not use the new version.

  ```
  cd /usr/local/lib/
  rm −f libMesaGL.so.2*
  ```

- If the new Mesa has a major version number that's greater than 2, you need to create a link to it with the name `libMesaGL.so.2`:

  ```
  cd /usr/local/lib/
  ln −s /wherever/you/installed/it/libMesaGL.so.3.0 libMesaGL.so.2
  ldconfig
  ```

lib3dfxgl.so **

With Quake II version 3.19, an alternative to the Mesa library is available. `lib3dfxgl.so` is a mini−GL driver optimized for Quake that provides better framerates than Mesa. This is a port of a driver that 3Dfx developed for Quake under Windows. It's included in the Quake II package, and there's no reason you shouldn't use it.

Like Mesa, `lib3dfxgl.so` requires that the Glide library be installed in order to access your 3Dfx card.

Choosing a GL driver **

With version 3.20, using this driver instead of Mesa is much easier than it was previously. There's a new CVAR, `gl_driver` that you set to indicate which GL driver `quake2` should use. To run with the `lib3dfxgl.so` driver, do:

```
./quake2 +set vid_ref gl +set gl_driver lib3dfxgl.so
```

To run with Mesa, do:

```
./quake2 +set vid_ref gl +set gl_driver libMesaGL.so
```

Note that the `.so` files you refer to must exist in your quake2 directory. A symbolic link is fine if, say, your Mesa library is in `/usr/local/lib` and you don't want to have two copies around.

3.8 The GLX Renderer ++

`ref_glx.so` is linked against standard OpenGL libraries instead of Mesa. This allows Quake II to run on other 3D hardware that is supported by other OpenGL implementations. At this time, I dont' know of any OpenGL implementations that support hardware other than 3Dfx, but this renderer ensures that when they appear, we'll be able to play Quake II with them.

Use of the GLX interface also removes GL Quake II's dependency on SVGAlib for keyboard and mouse input.

This is a GLX application, and as such, must be run from X.

You can use this client with Mesa/3Dfx if you install Mesa and Glide as explained in the previous section, then set the `$MESA_GLX_FX` environment variable to "fullscreen" before you run `quake2`:

```
export MESA_GLX_FX=fullscreen
./quake2 +set vid_ref glx +set _windowed_mouse 1
```

Why the `+set _windowed_mouse 1` option? Remember that this is an X application which happens to use your 3Dfx card. Even though the display takes up your entire screen, Quake II is stil running in a window. This means that if you're not *very* careful, you could move the mouse pointer outside the Quake II window, and Quake II will suddenly stop responding to mouse and keyboard input. `+set _windowed_mouse 1` avoids this problem by telling `quake2` to grab the mouse and not let it move outside its window.

3.9 Linux−Specific Command Line Options

This section will cover command line options that are specific to the Linux version of Quake II. There are plenty of other Quake II options, but they're beyond the scope of this HOWTO. Check out some of the sites listed in section General Quake Information for this kind of information.

These are actually cvars (client variables) that you can set in the Q2 console, but it makes the most sense to set them on the command line. Set them with `+set` on the command line, like:

```
        ./quake2 +set cd_dev /dev/hdc
```

cd_dev device

Name of the CD−ROM device.

nocdaudio value

Disable CD audio if *value* is nonzero

sndbits num

Set sound bit sample size. Default is 16.

sndspeed num

Set sound sample speed. Usual values are 8000, 11025, 22051 and 44100. If set to zero, causes the sound driver to attempt speeds in the following order: 11025, 22051, 44100, 8000.

sndchannels num

Indicates stereo or mono sound. Defaults to 2 (stereo). Use 1 for mono.

nostdout value

Don't do any output to stdout. Use this if you don't want all the console output dumped to your terminal.

3.10 Quake II Servers ++

Linux's strength as an internet server make it a perfect platform for running an internet Quake II server. This section will touch on the basics and Linux−specific aspects of starting up a Quake II server. Try http://www.3dgw.com/hellsgates/serverguide.htm for more detailed Quake II server setup information.

Listen Servers

You can start a Quake II "Listen" server from within the game via the *Multiplayer* menu. This allows you to host a game and participate in it at the same time.

To start a Listen server, start Quake II, bring up the Quake II menu with the ESC key, and select *Multiplayer*. It should be pretty self−explanatory from there.

Dedicated Servers

For a permanent, stand−alone Quake II server that needs to run without constant attention, using the Listen server is impractical. Quake II has a Dedicated server mode that is better suited to this type of use. A dedicated server is started from the command line and uses fewer system resources than a Listen server because it doesn't start the graphical client piece at all.

To start a dedicated server, use the command line option `+set dedicated 1`. You can set additional server parameters either on the command line or in a config file that you `+exec` on the command line. Your config file should reside in the `baseq2` directory.

A few common server options are listed below. To set options on the command line, do `+set fraglimit 30`. Options are set the same way in a config file, only you don't want the `+` before the `set`. Invoke your config file like this: `+exec server.cfg`.

fraglimit

Number of frags required before the map changes

timelimit

Time in minutes that must pass before the map changes

hostname

The name of your Quake II server. This is an arbitrary string and has nothing to do with your DNS hostname.

maxclients

The maximum number of players that can connect to the server at once.

For enough Quake II console and command line information to choke a horse, see Farenheit 176 (http://www.planetquake.com/f176).

Other Sources of Server Information

- The Q2 Server FAQ has a basic step−by−step guide to set up a Q2 Server under Linux: http://www.bluesnews.com/faqs/q2s−faq.html
- Grant Cornelius Reticulus Copernicus Sperry (flubber@xmission.com) has some basic Q2 server config files and startup scripts at http://www.atomicage.com:80/quake/server/server_cfg/.

3.11 Mods & Addons

Quake II modifications like Capture the Flag, Jailbreak, and Lithium II are very popular extensions of the original Quake II game. Some mods reside entirely on the server (Lithium), and some also require changes to your client (CTF). For server only mods, you just connect normally and play. Client−side mods require you to install additional files in your `quake2` directory before you can play.

Client Side Mods

Generally, installation of a client−side mod consists of just downloading the client package and upacking it in your Quake II directory, but you should refer to the mod's documentation for specific details. It may be necessary to download a Linux−specific package in addition to the main (Windows) client package. Also be aware that all mods may not be available for Linux.

Client−side mod packages usually contain a new `gamei386.so` file and one or more `.pak` files. Other new files may be included as well. These new files will be installed in a subdirectory below your Quake II directory. Use `+set game` *`mod−dir`* on the command line to run the mod. Rocket Arena 2, for example, gets installed in a directory called `arena`. To play RA2, your would start your client like so:

```
./quake2 +set game arena
```

Capture the Flag

Since this is by far the most popular variation of multiplayer Quake II, I've included specific instructions for installing this mod. Capture the Flag for Quake II is available from id's ftp site. Download it, then install like so:

```
cd /usr/local/games/quake2
mkdir ctf
cd ctf
unzip −L /wherever/you/put/it/q2ctf102.zip
```

Start Quake II with `+set game ctf` to play CTF.

Server Side Mods

Running a Quake II mod on a server isn't much different than running one on the client side. Generally you'll need to install `gamei386.so` and `server.cfg` files in a new subdirectory and then start your server like

```
./quake2 +set game XXXX +set dedicated 1 +exec server.cfg
```

Where XXXX above is the name of the mod's new subdirectory. The exact procedure will vary from mod to mod, of course. See the mod's documentation for specific details.

Game Source ++

The entire game, with the exception of the engine itself, resides in a shared library, `gamei386.so`. Quake II mods are created by changing the contents of this file. The C source is freely available (section Download the Necessary Files above) for anyone to download and modify.

After you've downloaded the source, here's how to get started with it:

```
cd /usr/local/games/quake2
mkdir mymod
cd mymod
gunzip /wherever/you/put/it/q2src320.shar.Z
sh /wherever/you/put/it/q2src320.shar
```

You'll be presented with a bunch of legalese that you must answer `yes` to, then the game source will be extracted. Building a new `gamei386.so` out of these sources is accomplished with a simple `make`. You can run Quake II with the newly compiled library like so:

```
cd /usr/local/games/quake2
```

```
./quake2 +set game mymod
```

Not too exciting yet, since what you just built is identical to the "stock" `gamei386.so`, but this should be good information for aspiring mod authors.

Mission Packs

- **Mission Pack 1: The Reckoning** The Reckoning requires Quake II version 3.15 or later to run. You'll need at least 95 MB for a minimum installation. Another 90 MB are required if you want to install the video sequences as well. Assuming your CD is mounted on `/mnt/cdrom` and Quake II is installed in `/usr/local/games/quake2`:

  ```
  cd /usr/local/games/quake2
  cp −r /mnt/cdrom/Data/all/* xatrix/
  rm −f xatrix/gamex86.dll
  ```

  If you want to install the video sequences:

  ```
  cp −r /mnt/cdrom/Data/max/xatrix/video xatrix
  ```

  Play The Reckoning like this:

  ```
  cd /usr/local/games/quake2
  ./quake2 +set game xatrix
  ```
- **Mission Pack 2: Ground Zero** It requires Quake II version 3.17 or later to run. You'll need at least 120 MB for a minimum installation. Another 115 MB are required if you want to install the video sequences as well. Assuming your CD is mounted on `/mnt/cdrom` and Quake II is installed in `/usr/local/games/quake2`:

  ```
  cd /usr/local/games/quake2
  cp −r /mnt/cdrom/Data/all/* rogue/
  rm −f rogue/gamex86.dll
  ```

  If you want to install the video sequences:

  ```
  cp −r /mnt/cdrom/Data/max/rogue/video rogue
  ```

  Play Ground Zero like this:

  ```
  cd /usr/local/games/quake2
  ./quake2 +set game rogue
  ```

---

4. Related Software

4.1 QStat

Qstat is a command line based program that returns the status of internet Quake, QuakeWorld, and Quake 2 servers created by Steve Jankowski mailto:steve@activesw.com.

Here's the feature summary from the QStat homepage:

- Supports Windows 95, NT, and most Unixes
- Comes with C source code and a binary for Windows
- Supports old Quake (NetQuake), QuakeWorld, Hexen II, and Quake II servers
- Can display all available statistics, including player info and server rules
- Output templates for automatic HTML generation
- Raw display mode for integration with HTML page generators
- Built−in host name cache
- Sort by ping time, game, or both
- More options than you can wiggle a mouse at

Qstat is a must−have tool if you're planning on doing any net play. A number of front−ends for qstat have been written as well. Some of them are listed later in this section.

You can get the latest version of qstat from the QStat Homepage (http://www.activesw.com/people/steve/qstat.html).

4.2 XQF

XQF is a graphical front−end to QStat that uses the GTK toolkit. This is the best QuakeWorld/Quake2 server browser that currently exists, and Roman Pozlevich (roma@botik.ru), is still cranking out revisions at the rate of about one per month.

If you're familiar with GameSpy for the Windows platform, this is the closest thing to it for Linux.

The XQF homepage is at http://www.linuxgames.com/xqf.

4.3 QuickSpy

QuickSpy is a text−based QuakeWorld server browser. It's another front−end to QStat and it works pretty well. If you don't run X and you don't have Quake II, this is a decent option. Beware though, it's no longer under development.

You can get QuickSpy at http://diana.ecs.soton.ac.uk/~rht96r/quake/quickspy/.

4.4 QPlug for Linux

Qplug is a Netscape plugin which retrieves and displays QuakeWorld and Quake II server information embedded in a web page. A Windows Qplug has been around for some time. The author, Olivier Debon (odebon@club−internet.fr) wrote the Linux version from scratch without ever having seen the Windows version.

QPlug for Linux can be got at http://www.geocities.com/TimesSquare/Labyrinth/5084/qplug.html.

4.5 qkHacklib

David Bucciarelli (tech.hmw@plus.it), author of the 3Dfx driver for Mesa, has written a library called qkHack, which tries to emulate all the SVGAlib/fxMesa functions used by Quake and Quake II. This would remove the need for SVGAlib when running glquake or Quake II with ref_gl. Other features from the qkHacklib README:

- You can dynamically switch between fullscreen rendering and the in window rendering just pressing the TAB key (you must start your X server in 16 bpp mode in order to use this feature)
- You can press Ctrl−C in the shell or kill the Quake process without problems
- you can enable/disable the mouse and keyboard 'grabbing' pressing F11/F12
- you can iconify and pause Quake pressing the F10 (it will not eat more CPU cycles). You can restart everything with a double click in the "****" icon
- you can run Quake with any Mesa driver (for example with the X11 driver but you must recompile the Mesa without the Voodoo driver)
- you can run Quake under any Linux box and get the hardware acclerated output on a SGI box (OK, this is a bit exotic and theoretical as feature but it is an example of how powerful can be an GLX/OpenGL application)

I've tried qkHacklib and it works great for Quake on my system. In Quake II, however, the mouse response becomes *really* slow. Others report complete success, though, so give it a try if it sounds like something you need.

David Bucciarelli's qkHacklib web page is at http://www−hmw.caribel.pisa.it/fxmesa/fxqkhack.html.

4.6 GiMd2Viewer

GiMd2Viewer is a Quake 2 model viewer written for Gtk and OpenGL by Lionel Ulmer (bbrox@mygale.org). It loads models and textures from either plain files or .PAK files. It will also animate the models (with frame interpolation).

This program is still under developement and I haven't tried it yet, but it sounds pretty nifty. Check it out at http://www.mygale.org/~bbrox/GiMd2Viewer/.

4.7 QIPX

QIPX is a set of programs that allow Linux Quake clients (using TCP/IP) to connect with DOS Quake clients (using IPX). I guess this is useful if you're playing netquake on a LAN. QIPX is available at http://www.geocities.com/SiliconValley/Park/6083/qipx.html.

4.8 Ice

Ice is a Quake map editor for UN*X created by C.J. Beyer and John Watson. I haven't used this program, nor do I know what its development status is. The Ice homepage is at http://styx.phy.vanderbilt.edu/~ice/.

4.9 Q2getty

Q2getty is a hack of mingetty by Mike Gleason (mgleason@ncftp.com) that allows you to automatically run and respawn a program (like a Quake server) on a virtual console. This program is available in the files section at http://www.ncftpd.com/unixstuff/q2getty.html.

4.10 rcon

Rcon is a pair of tools that allow remote administration of a Quake II server using the RCON protocol. Michael Dwyer (michael_dwyer@mwiworks.com) is the author. Rcon 1.1 is available at http://sunsite.unc.edu/pub/Linux/games/quake/rcon−1.1.tar.gz.

4.11 qlog

Qlog is a GPL'ed QuakeWorld/Quake II server log parser that generates comprehensive player statistics. Craig Knudsen (cknudsen@radix.net) is the author. The qlog homepage is http://www.radix.net/~cknudsen/qlog/.

4.12 Cheapo

Cheapo is a proxy that can be used to route QuakeWorld network traffic. Additionally, the proxy can modify the data and has features for enhancing gameplay. You can connect to cheapo as if it were a Quakeworld server, and then give it commands that forward you to a real server. The proxy can also be run on a firewall machine incapable of handling Quake traffic, so that machines inside the firewall can be used for playing. The Cheapo homepage is at http://www.saunalahti.fi/~softech/.

4.13 qgraph

QGraph (Quake Graph) is a utility to help people in managing Quake's DeathMathes, turnments and Quakeworld games. QGraph is a program who connects (via Lan or the Internet) to a Quake, QuakeII, Quakeworld and Hexen2 Servers and shows you realtime data about the game running on that server. The QGraph homepage is at http://www.frag.com/qgraph.

4.14 Vispatch ++

Vispatch is a utility that patches your Quake 1/QuakeWorld levels to support transparent water. This Windows utility has been around almost since the dawn of GLQuake, but just recently (to my knowledge) a Linux version has surfaced. The WaterVis homepage is at http://www.sod.net/vis. You can download the Linux version of the Vispatch utility at http://www.sod.net/vis/files/unixvis.zip.

4.15 GStat ++

GStat is a server query tool by Aaron Levinson based on QStat, but with some unique extra features. From the GStat homepage: GStat (GameStat) is a utility for querying Quake, Quakeworld, Quake II and Hexen II servers. It has three main functions:

1.Displays statistics for Quake−based servers 2.Functions as a remote server administration tool
3.Console−based server browser for finding a server to play on

Find out more at http://www.ews.uiuc.edu/~alevinsn/gstat.html.

4.16 QuakeLaunch ++

QuakeLaunch is a nice, simple Quake server browser that you can run in the console. It's based on qstat by Steve Jankowski. If you can't or don't want to use an X−based server browser like XQF, this is a good alternative. The QuakeLaunch homepage is at http://www.linuxgames.com/qlaunch.

4.17 QPLog ++

The Quake Player Log is a master/client Perl program that keeps track of player statistics. The object of this program is similar to what the original QuakeWorld attempted. Server logs are sent to the central QPLog master, which keeps track of player statistics and ranks players according to a number of factors. http://junior.ece.ucdavis.edu/qplog/

5. Troubleshooting/FAQs

5.1 General

OS difference considerations

- **Case sensitivity** − In DOS and Windows, case is not important. `BASE1.TXT` is the same as `base1.txt` is the same as `Base1.Txt`. Under Linux and other unices, case IS significant. `MOTD.TXT` and `motd.txt` are different files. This can cause problems with player models and skin files if they're installed with upper− or mixed−case filenames. `players/male/santa.PCX` needs to be renamed to `santa.pcx` in order for the Santa skin to be visible in Linux. The `fixskins.sh` script included with quakeworld will convert all filenames in a directory to lowercase. It's reproduced below for your convenience:

```
#!/bin/sh
for x in *; do
   y=`echo $x | tr '[A-Z]' '[a-z]'`
   if [ $x != $y ]; then
      mv $x $y
   fi
done
```

- **Path delimiters** − DOS and Windows use the backslash "`\`" character to separate file pathname elements. In Unix, the backslash is an escape character. Quake and Quake2 for Windows recognize both "`\`" and "`/`" as valid path delimiters, so if you use file pathnames in your config files (or your mod code, or anywhere else, for that matter), be sure you're using "`/`" and not "`\`".
- **End of line characters** − Under DOS/Windows, each line of a text file ends with a carriage return character (CR) and a linefeed character (LF). Unix textfiles just have a linefeed at the end of each line. Using DOS/Win formatted textfiles in Unix can cause all kinds of mysterious Quake problems. For example, the incorrectly formatted `quake2.conf` file from the Quake2 3.17 package generated the error "`LoadLibrary("ref_XXX.so") failed: No such file or directory`".

LMCTF−TE reports a floating point exception. If you've got an problem you can't explain, try removing the CRs from your text files:

```
mv file.txt file.bak; tr −d '\r' < file.bak > file.txt
```

Glibc, RedHat 5.x, Debian 2 considerations (for Quake 1 only)++

[The following applies to the Quake I binaries (`squake`, `glquake`, and `quake.x11`) only. As of versions 2.30 and 3.19 respectively, QuakeWorld and Quake II are available in both libc5 and glibc versions.]

The Quake executables were compiled with libc5. Newer Linux distributions like RedHat 5.1 and Debian 2.0 use the incompatible libc6 (or glibc) as their default C library. If you're running Quake on a glibc system, there are a few things to watch out for:

- Both RedHat 5 and Debian 2 have libc5 compatibility packages that allow you to run libc5−based applications. Make sure you have these packages installed. Both distributions put the libc5−based libraries in `/usr/i486−linux−libc5/lib`.
- Make sure Quake uses the correct libraries. Create a script like the one below that points `$LD_LIBRARY_PATH` to your compatibility libraries directory before it runs Quake.

```
#!/bin/sh
export LD_LIBRARY_PATH=/usr/i486−linux−libc5/lib
./quake2 +set vid_ref gl $*
```

- If you are going to compile a library like SVGAlib or Mesa for use with Quake, it must be compiled with libc5 and friends. Simply building a new library according its installation instructions will result in a library that is linked against your default library, glibc. You must ensure your new library is linked only with libc5 and libc5−based libraries so it will be compatible with Quake. See your distribution's documentation for information on linking to the non−default libraries.

My mouse doesn't work or seems to respond randomly. **

- Is gpm running? gpm is a program that enables you to cut and paste with the mouse in virtual consoles. Many distributions enable it by default. It may be interfering with Quake. Check if it's running with the following command:

  ```
  ps aux | grep gpm
  ```

  If you get output like

  ```
  root      6115  0.0  0.4   832   316  ?  S   17:54   0:00 gpm −t PS/2
  ```

  then gpm is running and interfering with Quake. `gpm −k` (as root) ought to stop gpm. If it doesn't (`gpm −k` doesn't always work on my system), kill gpm with the command `killall gpm`. If you never use gpm, you may want to stop it from running at startup. See the documentation for your distribution for information on how to do this.

- Is your mouse defined properly in `libvga.config`? This file usually lives in `/etc` or `/etc/vga`. Open it up and look for a line like

```
mouse Microsoft
```

On my system, this is the first option in the file. Make sure the mouse type is appropriate for your hardware.

- Do you have RW permissions for your mouse device? The permissions of your mouse device may only allow you read access to it. Quake wants to open it read−write, so you'll need to either do a `chmod 666` to it, or run as a user or group that's allowed to read and write to it. The actual device file in question will vary according what type of mouse you have on your system. Most of the time, `/dev/mouse` is a symbolic link to your actual mouse device file, so doing an `ls −l /dev/mouse` should allow you to find which device file to modify.

My Microsoft Intellimouse or Logitech MouseMan+ isn't working correctly.

SVGAlib, which handles mouse input for SVGA and GL Quake/QW/Q2, didn't directly support the Intellimouse until version 1.3.0. If you have a version of SVGAlib prior to 1.3.0, you should upgrade, then use mouse type `IntelliMouse` (for serial mice) or `IMPS2` (for PS/2 mice) in your `libvga.config` file.

My mouse is "laggy" and seems much slower than under Windows.

- For many people, just cranking up the value of `sensitivity` in the game console cures the problem. Setting `sensitivity` by hand in the console or in a `.cfg` file allows you to increase the mouse sensitivity more than the slider in the Options menu allows. `sensitivity 15`, for example.
- From Zoid's 1/7/98 .plan update:

  > If you are experience video 'lag' in the GL renderer (the frame rate feels like it's lagging behind your mouse movement) type "gl_finish 1" in the console. This forces update on a per frame basis.
- The latest version of SVGAlib (1.3.0) provides a slew of parameters in `libvga.config` that you can use to customize the behavior of your mouse. With the proper settings it should be possible to make your mouse feel any way you want. On my system, just changing `mouse_accel_type` to `normal` (default is `power`) gave me the results I wanted. I haven't messed with the other settings, and I don't pretend to have a clue about what they all do.

I have a Voodoo2, and when I try to run with the gl renderer it reports that I don't have a Voodoo card installed.

There are different versions of Glide for Voodoo and Voodoo 2 cards. Be sure you downloaded the correct one for your system.

When I'm playing any of the Quake games under SVGAlib or GL and press CTRL−C, the game exits and sometimes leaves my console in an unusable state.

Sometimes when Quake/Quake II exits abnormally, it leaves my console unusable.

Yes. This bites. SVGAlib catches the CTRL−C and decides what to do with it instead of allowing Quake to handle it. I know of no way around this short of hacking SVGAlib.

If you run your Quake games from a script that resets the keyboard and terminal like the one below, you'll run less chance of ending up with a hosed terminal if this does happen, though.

```
#!/bin/sh
./quake2 $*
kbd_mode −a
reset
```

squake/quake2 fails to start and says "`svgalib: cannot get I/O permissions`"

The Quake executables must run as root when using the SVGAlib renderer, so you must either run them as root or make them setuid root. See the installation instructions in this document for details.

Sometimes after playing one of the Quake games in X, key repeat doesn't work any more.

For some reason, the X11 versions of Quake disable key repeat while they're running. If the program exits abnormally for some reason, key repeat never get turned back on. Do

```
xset r on
```

to reenable it.

Quake/Quake II says "/dev/dsp : device not configured"

Your sound hardware is not properly configured. You may simply need to do a `insmod sound`, or it may be necessary to rebuild your kernel. RedHat users may need to invoke the **sndconfig(8)** utility. See the documentation for your Linux distribution and/or the Linux Sound HOWTO for information on configuring your system's sound hardware.

GL Quake/Quake II run slower in Linux than in Windows. ++

The Windows 3Dfx GL miniport is heavily optimized for the things Quake II does. Mesa on the other hand, is more general and less optimized As a result, Linux Quake II runs slower than under Windows. This isn't a limitation of Linux, but a limitation of the current drivers.

With the most recent releases of QuakeWorld and Quake II, the 3Dfx miniport mentioned above is available for Linux. Also, in Quake II version 3.20, the OpenGL function handling code was completely rewritten, resulting in significant speedups.

Additionally, for Pentium Pro and Pentium II users, there are some tweaks than can be done with memory buffering − the latest `/dev/3dfx` device driver has support for automatically setting this up for you. Enabling MTRRs can result in *significant* (10 fps on my system) GL Quake speedups. See http://glide.xxedgexx.com/MTRR.html for some more detailed information about this.

On a PPro/PII system with a Voodoo2, performance under Linux is now at least as fast as under Windows.

How can I start a server and log off, then come back to it later?

screen(1) is a great utility for this sort of thing. It allows you to create many virtual screens in one tty and switch between them. Screen comes with most distributions. You can download it from ftp://prep.ai.mit.edu/pub/gnu or any GNU mirror.

Start screen by typing the command `screen`, then create a new screen window by pressing CTRL−A CTRL−C. You won't see much as you do these things, but be assured, something is happening.

Start a Quakeworld server:

```
/usr/local/games/quake/qwsv
```

Now open a new screen window with CTRL−A CTRL−C and start up a Quake II server:

```
/usr/local/games/quake2/quake2 +set dedicated 1
```

You can switch back and forth between your servers by pressing CTRL−A CTRL−N.

Press CTRL−A CTRL−D to detach from the screen program. Screen and your servers are still running, but they're no longer visible in your terminal window. You can logoff now and your processes will continue to run normally.

Use `screen −r` to re−attach to your previous screen process and access your servers again.

That's all there is to it. See the **screen(1)** man page for more detailed information.

How can I keep the mouse from leaving the window in X? ++

From the Quake II `readme.linux` file (this applies to all version of Quake):

> By default, the mouse will not be 'tied' to the Quake2 window. To cause Quake2 to grab the mouse, select 'Windowed Mouse' from the video menu, or type '_windowed_mouse 0' at the console. Do the reverse to release it. You can bind keys to grab and release the mouse in the console, like so: bind i "_windowed_mouse 1" bind o "_windowed_mouse 0" Then "i" will grab the mouse and "o" will release it.

When I run Quake/QuakeWorld/Quake II under X, the image only takes up half the window and/or the colors are all funny. ++

The X clients only support color depths of 8 and 16 bits per pixel. Your X server is probably running at 24 or 32 bpp. Change the default color depth to 16 bpp and this problem will go away.

Where can I get the source code for Quake/QuakeWorld/Quake II so I can compile it for PPC/Alpha/Gameboy/etc? ++

The source code for the Quake engines is the proprietary property of id Software. While there's a good chance that id will release the source to Quake eventually like they did for Wolfenstein 3−D and Doom, they have not done so yet. I don't have the source to Quake no matter how nicely you ask me for it.

5.2 Quake/QuakeWorld

Quake dies at startup with a segmentation fault.

This usually means your network setup isn't right. Try starting quake with the `−noudp` option and see if the error goes away. If that fixes it, check your `/etc/hosts` file and verify there's an entry for your machine in it. Use 127.0.0.1 for your IP address if you have a dialup account that gives you a different address each time you connect.

What's the difference between `glqwcl`, `glqwcl.3dfxgl`, and `glqwcl.glx`?

- **`glqwcl`** is the standard GL Quakeworld client you've seen in previous versions. It's linked against libMesaGL.so.2.
- **`glqwcl.3dfxgl`** is a script that runs `glqwcl` after preloading the 3Dfx GL miniport library, `lib3dfxgl.so`. Preloading the minport library causes its GL functions to get used instead of Mesa's. Since the GL minport is optimized for Quake, this is a good thing.
- **`glqwcl.glx`** is linked against standard OpenGL libriaries instead of Mesa. This will allow glquake to run on other 3D hardware that is supported by some other OpenGL implementation. This is an X application and so must be run from X.

When I run `glqwcl.glx` fullscreen from X, I can't use my mouse or keyboard.

Run `glqwcl.glx` with the `+_windowed_mouse 1` option. GLX Quakeworld is running in a window, even though it appears to take up your whole screen. If you move the mouse while the window manager is in focus−follows−mouse mode, you're likely to move the pointer outside this window, and then Quake will stop responding to mouse and keyboard input. `+_windowed_mouse 1` makes Quakeworld grab the mouse exclusively.

5.3 Quake II

When I try to run Quake II with the GL renderer, it fails and says "LoadLibrary("ref_gl.so") failed: Unable to resolve symbol"

If immediately prior to the "Unable to resolve symbol" line, you have messages like "`can't resolve symbol 'fxMesaCreateContext'`", your Mesa library doesn't have glide support compiled in. See section The GL renderer in the Quake II installation section for information on installing Mesa and glide.

Quake II fails with the message `LoadLibrary("ref_XXX.so") failed: No such file or directory`

- `/etc/quake2.conf` doesn't have the correct path to your Quake II directory in it. This file should contain one line that is the directory Quake II lives in.
- If `/etc/quake2.conf` *does* contain the correct path, try removing the file and re−creating it by hand. Some versions of Quake II for Linux included an incorrectly formatted `quake2.conf` file.
- Do you have SVGAlib installed? Check `/lib`, `/usr/lib` and `/usr/local/lib` for a file called `libvga.so.1.X.X`, where the X's are some numbers. If nothing turns up, you need to get and install SVGAlib to run Quake II outside of X.
- If the renderer in question is `ref_gl.so`, Mesa may not be properly installed. Did you copy `libMesaGL.so.2.6` to a library directory like the installation instruactions told you to?
- If the renderer in question is `ref_gl.so`, did you install the glide libraries?

When I update the brightness while using the GL renderer, and hit "apply," nothing happens!

Type `vid_restart` in the console to make the changes take affect.

Note about the 3.17 distribution

As of this writing, the most recent Quake II version is 3.20. If for some reason, you're running version 3.17 instead, the following information may be helpful to you.

Two text files (`quake2.conf` and `fixperms.sh`) in the 3.17 distribution were inadvertently saved in MS−DOS CR/LF text−file format instead of the unix LF format. This means there's an extra carriage return character at the end of each line in these files and they're not going to behave right until you fix them.

We'll run them through **tr(1)** to strip out the CR's.

```
for i in fixperms.sh quake2.conf
do
mv $i $i.bak
tr −d '\r' < $i.bak > $i
done
```

When I run Quake II with `+set vid_ref glx` fullscreen from X, I can't use my mouse or keyboard.

Run GLX `quake2` with the `+set _windowed_mouse 1` option. GLX Quake2 is running in a window, even though it appears to take up your whole screen. If you move the mouse while the WM is in focus−follows−mouse mode, you're likely to move the pointer outside this window, and then Quake II will stop responding to mouse and keyboard input. `+set _windowed_mouse 1` makes Quake II grab the mouse exclusively.

Why can't I change to some of the SVGA modes that are in the Quake II Video menu?

SVGAlib probably doesn't know how to create the modes on your card. When Quake II starts up with the SVGA renderer (`ref_soft.so`), it prints a list of all the modes that SVGAlib tells it are available:

```
------- Loading ref_soft.so -------
Using RIVA 128 driver, 4096KB.
mode 320: 200 1075253220
mode 320: 240 1075253220
mode 320: 400 1075253220
mode 360: 480 1075253220
mode 640: 480 1075253220
mode 800: 600 1075253220
mode 1024: 768 1075253220
mode 1280: 1024 1075253220
```

These are the only modes you will be able to successfully switch to from the Video menu. If say, 512x384 isn't on the list, selecting it from the Video menu won't work.

SVGAlib does let you define new video modes for some chipsets in `libvga.config`, so you may be able create your own video mode this way. See the SVGAlib documentation for more detail on this topic.

When I try connect to a Quake II server that's running a mod, quake2 crashes with the message "Error: VID: Could not open display". **

According to Zoid, this problem occurs when the video system restarts due to a gamedir change. Several libraries get unloaded and reloaded during the restart, and this apparently causes quake2 to blow chunks on many systems.

The workaround solution is to set the `game` CVAR on the command line before you start up `quake2`. So if you're going to connect to a CTF server, you'd start `quake2` like:

```
./quake2 +set game ctf ...
```

This may seem inconvenient if you're used to connecting to new servers without leaving the quake2 program, but unfortunately it's the only way around this problem right now. A frontend program like XQF will automatically do this command line stuff for you, so you ought to consider using one if you're not currently doing so.

6. Tips & Tricks

6.1 Running X and GL games without setuid

If you only run the X and GL versions of Quake, QuakeWorld or Quake II, you don't need to run them with root permissions. SVGA is the only mode that must be run as root. The X versions just need access to `/dev/dsp`, the sound device. The GL versions need access to the 3Dfx card as well as to `/dev/dsp`.

`/dev/dsp` needs to be readable and writeable by Quake. Most distributions give it 662 (`rw−rw−−w−`) permissions by default. The simplest solution is to just `chmod 666 /dev/dsp`. On most systems, the ability to read from the sound device will not pose a significant security threat. If this approach is unacceptable for your system, create a group that owns `/dev/dsp` and make your Quake players members of that group.

You need the `/dev/3dfx` driver from Daryll Strauss' glide page (http://glide.xxedgexx.com/3DfxRPMS.html) in order run glide applications (like GLQuake) non−root. Download the `Device3Dfx.xxx.rpm` package and install according to the instructions on the web page. After you've installed the driver, make sure `/dev/3dfx` has 666 permissions (`chmod 666 /dev/3dfx`).

When `/dev/dsp` and `/dev/3dfx` are properly set up, you can remove the setuid bit from your Quake/QW/Q2 executables. Just do (as root) `chmod 0755 XXXXX`, where `XXXXX` is either `glquake`, `quake.x11`, or `quake2`.

If you've been playing as root prior to making these changes, many of your Quake files (like savegames) may be owned by root and inaccessible to a normal user, so remember to change the files' ownership before you attempt to play the game non−root.

6.2 Running SVGA and GL games from X

The GLX Quakeworld and Quake II clients are native X applications, but since they use Mesa rather than the 3Dfx mini−driver, they're slower than the `lib3dfxgl.so` versions. For this reason, you may still favor this way of starting games from X over using the GLX clients.

This is based on a Linux Gazette 2 Cent Tip by Joey Hess (joey@kite.ml.org) The original is at http://www.ssc.com/lg/issue20/lg_tips20.html#squake

Yes, it's possible to run the Quake games from X if you're root, but such behavior is naughty, and you still run the risk of having Quake crash and leave the console unresponsive. With a little work you can make it possible for a regular user to run SVGA and GL Quake from X *AND* automatically switch back to X when the program is finished, regardless of whether it exited normally or not.

Note: when I say "Quake" in the text below, I really mean "quake, glquake squake, qwcl, glqwcl, qwcl.x11 or quake2".

- First, you'll need the **open(1)** package by Jon Tombs. This is a set of two very small programs that allow you to switch between virtual consoles and start programs on them. Download open from http://sunsite.unc.edu/pub/Linux/utils/console/. You shouldn't have to do much more than `make;make install` to compile and install it. Once it's installed, you need to make the `open` and `switchto` executables setuid root. So do this:

```
cd /usr/local/bin
chown root open switchto
chmod 4755 open switchto
```

- Next, save the following code to a file called `getvc.c`:

```
/* getvc.c
 * Prints the number of the current VC to stdout. Most of this code
 * was ripped from the open program, and this code is GPL'd
 *
 * Joey Hess, Fri Apr  4 14:58:50 EST 1997
 */

 #include <sys/vt.h>
 #include <fcntl.h>

 main () {
   int fd = 0;
   struct vt_stat vt;

   if ((fd = open("/dev/console",O_WRONLY,0)) < 0) {
     perror("Failed to open /dev/console\n");
     return(2);
   }
   if (ioctl(fd, VT_GETSTATE, &vt) < 0) {
     perror("can't get VTstate\n");
     close(fd);
     return(4);
   }
   printf("%d\n",vt.v_active);
 }

 /* End of getvc.c */
```

Compile it and install it somewhere in your `$PATH`:

```
gcc getvc.c -o getvc
strip getvc
mv getvc /usr/local/bin
```

- Now create a script called `runvc`: your `$PATH`:

```
#!/bin/sh
# Run something on a VC, from X, and switch back to X when done.
# GPL Joey Hess, Thu, 10 Jul 1997 23:27:08 -0400
exec open -s -- sh -c "$* ; chvt `getvc`"
```

Make it executable and put it somewhere in your `$PATH`:

```
chmod 755 runvc
mv runvc /usr/local/bin
```

Now you can use the `runvc` command to start Quake. Continue to use whatever command line you usually use to start your game, but put `runvc` at the beginning:

```
runvc ./quake2 +set vid_ref gl +connect quake.foo.com
```

You'll automatically switch to a VC, run Quake and then switch back to X when it's done!

- **Update:** Joey Hess wrote in to say:

  I'd like to point out that some people may find that this doesn't work. If your /dev/ttyxx devices do not let you write to them by default (a good idea), then the open command won't be able to switch to them. In that case, you need to chown the next tty that is free after you start up X. I do this in xdm's Xtartup_0 script, as follows:

  ```
  # Set up tty8 as a console that is writable by the current user, so
  # open -s commands can be used to run stuff there.
  # Tty8 is used becuase it's the next free tty after 7 where X runs,
  # and so open tries to use it.
  chmod 640 /dev/tty8
  chown $USER.root /dev/tty8
  # This does open up a tiny security hole:
  #       user_a logs in with xdm, then logs out.
  #       user_b logs in at console.
  #       user_a can now redirect user_b to tty9 via the open command,
  #       and plop them down into some program that, perhaps, tries to
  #       persuade them to enter their password.
  ```

6.3 3Dfx "tweak" settings work in Linux too

You may have seen or heard mention of ways to "tweak" Quake's OpenGL performance by setting various environment variables. These variables are equally valid under Linux. You just set them slightly differently. If a Windows/DOS "tweak" guide tells you to:

```
SET SST_GRXCLK=59
```

Under Linux, do it like this:

```
export SST_GRXCLK=59
```

6.4 The Poor Man's Server Browser

Joe S. (jszabo@eden.rutgers.edu) suggests:

```
a trick I do with qstat:
I make a file of my favorite servers, then do
qstat -f /C/quake2/file.txt | less
```

hit control z to suspend, then copy the ip address or hostname I want with gpm to a command line like:

```
./quake2 +connect expert.eqclans.com
```

Then I can use fg to go back to the list later...

6.5 Using `lib3dfxgl.so` for Quake I

You can use the 3Dfx mini−driver (`lib3dfxgl.so`) from Quakeworld or Quake II with `glquake` too. Simply copy the `glqwcl.3dfxgl` or `quake2.3dfxgl` scripts to a new file, say `glquake.3dfxgl`. Then edit the `glquake.3dfxgl` script so it runs `glquake` instead of `glqwcl`. The same restrictions about running as root apply here, as do the differences between running under glibc and libc5. See the Quakeworld or Quake II sections for more information on `lib3dfxgl.so`.

6.6 Getting rid of that annoying "POSSIBLE SCAN CODE ERROR 57" message in GLQuake. ++

If you're sick of seeing that useless message every time you press the space bar in GLQuake, execute the following command from your quake directory and you'll never have to deal with it again. This little piece of Perl code just replaces the first character of the message with a null character, which signifies the end of a string in C programs. Now when GLQuake thinks it needs to tell us about a possible scan code error, it outputs an empty string instead! Oh, and your original `glquake` binary gets automatically backed up to `glquake.bak` too.

```
        perl −i.bak −0777pe 's/P(ossible unknown scancode)/\0$1/g' glquake
```

Thanks to Andrew Chase (fixy@fixy.org) for this.

7. Administrivia

7.1 New Versions of This Document

New versions of this document will be periodically posted to comp.os.linux.answers and rec.games.computer.quake.misc. They will also be uploaded to various WWW and FTP sites, including the LDP home page.

New versions of this document will be periodically posted to rec.games.computer.quake.misc and comp.os.linux.misc.

The latest version of the Linux Quake HOWTO can always be found at the following sites:

- http://www.linuxquake.com/
- http://webpages.mr.net/bobz/
- http://www.linuxgames.com/quake

7.2 Distribution Policy

Copyright (c) 1998, Bob Zimbinski, Brett A. Thomas and Mike Hallock. This document may be distributed under the terms set forth in the LDP license at sunsite.unc.edu/LDP/COPYRIGHT.html.

This HOWTO is free documentation; you can redistribute it and/or modify it under the terms of the LDP license. This document is distributed in the hope that it will be useful, but *without any warranty*; without even the implied warranty of *merchantability* or *fitness for a particular purpose*. Heck, even if this document *kills your dog*, it's not our fault.

See the LDP license for more details.

7.3 Revision History

Sections of this document that were updated in the last revision have a ** after the section heading. Sections updated in the revision prior to the last are marked with a ++.

Below is the full revision history of this document. It is not intended to be useful to anyone but the document authors.

```
$Log: Quake−HOWTO.sgml,v $
Revision 1.0.1.18  1998/12/19 00:56:31  bobz
Minor typo removed

Revision 1.0.1.17  1998/12/19 00:45:01  bobz
Moved "Other Formats" section to top of the document for better visibility
Added troubleshooting topic about crashes upon connecting to a server w/different gamedir
Added mention of RedHat's broken ld.so configuration to Mesa sections
Added details about using the new gl_driver cvar
Added troubleshooting bit about mouse permissions

Revision 1.0.1.16  1998/12/12 05:34:02  bobz
updated file links for quake2 version 3.20
URL change for Linux Game Tome
Added linuxgames.org to list of Linux Game sites
Removed nasty references to /usr/games/quake as installation dir
Removed info about compiling libc5 libraries for use with Q2
Added a few sentences about /dev/3dfx and mtrr support in the Q1 & Q2 glide sections.
Added a warning about using the glide library from 3dfx's site
mentioned that GLX versions no longer require svgalib
Updated performance FAQ
Moved mouse in X tip from Tips to FAQ
Added Vispatch to Related Software
Added quakelaunch to Related Software
Added qplog to Related Software
Added link to server setup guide at 3dgw.com
Added X color depth FAQ
Added Quake source code FAQ
Added new info from JH in Running from X Tip.
Added fix for POSSIBLE UNKNOWN SCANCODE in glquake
Minor wording changes and type fixes throughout

Revision 1.0.1.15  1998/10/05 21:09:42  bobz
Fixed a few typos and spelling errors that I missed last time around

Revision 1.0.1.14  1998/10/05 19:54:17  bobz
Removed the long−dead LinQuake page from list of Linux Quake sites
Removed renaissoft.com link
Minor wording changes throughout
Added link to svgalib libc5 binary
Added info about q2 demo
Updated q2 installation info for the 3.19 release
```

```
moved section about 3.17 textfile problems to the troubleshooting section
Added part about lib3dfxgl.so to QW section
Added part about GLX client to QW section
Added part about lib3dfxgl.so to Q2 section
Added part about GLX client to Q2 section
Updated "slower than windows" FAQ
Removed obsolete TS tip about glibc & q2
Removed obsolete TS tip about playing w/CD mounted
Changed tip about running glquake with lib3dfxgl.so
Added instructions for Q2 mission pack 2
Added pointer to svgalib libc5 binary
Added FAQ about svgalib modes in Q2
Added Cheapo proxy to software section
Added screen FAQ
Added qgraph to  software sectoin
Decided I was lying when I promised a 3Dfx tweak section in the next update

Revision 1.0.1.13  1998/09/02 01:35:18  bobz
Fixed some broken url links
Restored <author> tag
added an acknowledgement
added a reminder to the non−root tip
added qlog listing

Revision 1.0.1.12  1998/08/30 21:35:23  bobz
Updated QuakeWorld install section for 2.30 release
Updated QW file list
Updated Quake 1.01 install info per email suggestion
Added QW 2.30 caveat to "Glibc considerations" section
Updated linux vs. windows speed section for new 3dfx miniport
Added glqwcl versions FAQ
Added troubleshooting tip about windows focus and glx
Added a sentence about /dev/3dfx permissions
Made runvc script usage clearer
Added tip about using mini−driver with Quake & QuakeII

Revision 1.0.1.11  1998/08/15 19:28:49  bobz
Added glibc topic in Troubleshooting/FAQ section
Moved non−setuid info to Tips & Tricks section
Moved rcs log into Revision Hist section
Changed distribution info a little until LDP wakes up
Changed references to quake.medina.net to www.linuxquake.com
Added linuxgames.com to list of places to get current version
Changed 3dfx Howto pointer
Moved explanation of change markings (** and ++) to intro section where they'll be noticed
Mentioned zoid's GL driver port in the Win vs Linux FAQ
Fixed credit and URL in q2getty

Revision 1.0.1.10  1998/08/07 19:07:02  bobz
Changed XQF url
Added comments to the revision history section
Started marking changed sections
Added OS Considerations to troubleshooting section
Added rcon to Other Software section
Added ICE to Other Software section

Revision 1.0.1.9  1998/08/04 21:06:22  bobz
Updated Acknowledgements
Added discussion of Quake security
Added QIPX to "Related Software"
Added ts topic about −noudp
```

```
Revision 1.0.1.8  1998/08/03 22:09:28  bobz
Updated and restructured the q2 mods section
Minor change to gpm troubleshooting info
Changed sound ts info
Changed intellimouse ts info

Revision 1.0.1.7  1998/08/03 04:55:21  mikeh
Added "Playing" section of Q2 Mods & Addons section.
Added "Intelimouse" section of Troubleshooting section
Added RedHat sndconfig part to Troubleshooting section
Fixed a few spacing problems, and one error message

Revision 1.0.1.6  1998/08/03 02:01:13  bobz
Changed abstract.

Revision 1.0.1.5  1998/07/31 16:07:11  bobz
Added gpm −k mention in mouse troubleshooting
mentioned 'sensitivity' in mouse lag troubleshooting
Added qstat|less tip
Added "/dev/dsp not configured" to troubleshooting
Slightly reworded part of the mod compiling section
Added bit about running mods server side
corrected required version # in q2 mission pack 1
Updated acknowledgements section

Revision 1.0.1.4  1998/07/30 21:08:46  bobz
Fixed a problem with broken lines in <sect> tags

Revision 1.0.1.3  1998/07/30 17:40:29  bobz
Fixed broken link to Latex version.

Revision 1.0.1.2  1998/07/30 15:46:40  bobz
Test update.

Revision 1.0.1.1  1998/07/30 15:43:59  bobz
Changed version numbering scheme.

Revision 1.10  1998/07/30 13:51:24  bobz
Minor version reporting change

Revision 1.9  1998/07/29 20:42:03  bobz
Added some server links

Revision 1.8  1998/07/29 17:37:15  bobz
Added a Quake II model viewer to the Other Programs section

Revision 1.7  1998/07/29 16:02:23  bobz
Added some relative links in Other Formats to make it
more portable.

Revision 1.6  1998/07/29 15:18:12  bobz
Version and date stamps back to the way they were

Revision 1.5  1998/07/29 14:52:03  bobz
Added tags for automatic timestamping

Revision 1.4  1998/07/29 13:17:51  bobz
Changed captured.com's web address per webmaster's request

Revision 1.3  1998/07/29 02:11:02  bobz
Ooops again.  Made "Other formats" a <sect> instead of a <sect1>.
```

```
Revision 1.2  1998/07/29 01:55:41  bobz
Oops.  I deleted the doctype tag...

Revision 1.1  1998/07/29 01:48:34  bobz
Added Other Formats section.
Changed acknowledgments slightly
Added ftp.medina.net to list of ftp sites.

Revision 1.0 1998/07/28
First publicly released version

Revision 0.9 1998/07/25
Restructured,rewritten and expanded by Bob Zimbinski.

Revision 0.01 1998/06/16
First Pre−Release version v0.01by Brett A. Thomas and Mike Hallock.
```