

# Gentoo Linux GCC/GLIBC Updates

Contributed by Bryan

**DISCLAIMER:** Only an experienced Linux professional should perform updates to Linux servers. Doing it yourself can cause severe damage. These notes do not come with any warranty of any kind so please don't blow up your server and blame us.

Gentoo GCC Upgrade Page: [Gentoo GCC Upgrade Guide](#)

About how long this takes: Many, many more hours longer than you think, if you are not a Valley Technologies client with access to the binary depot.

## Step 1 - Prepare the System

Sync portage first...

```
# emerge --sync
```

If need be update portage itself to the latest version...

```
# emerge -ua portage
```

(Optional) Recompile wget with USE="static" just in case...

```
# USE="static" emerge wget
```

Download the dist files for everything you will need now just in case...

```
# emerge --fetchonly system
```

```
# emerge --fetchonly world
```

## Step 2 - Update GCC

Upgrade the GCC compiler...

```
# emerge -uav gcc glibc
```

Be sure to replace "i686-pc-linux-gnu-4.1.2" with YOUR GCC version and CHOST settings. You can verify your CHOST settings by viewing `/etc/make.conf`.

```
# gcc-config i686-pc-linux-gnu-4.1.2
```

```
# env-update && source /etc/profile
```

Fix and Re-build libtool

```
# fix_libtool_files.sh 4.1.1 [Replace 4.1.1 with your old gcc version]
```

```
# emerge --oneshot -av libtool
```

## Step 3 - Recompile Everything

Re-compile System and then World

Important: Please note that this is the part that takes forever if you're not a Valley Technologies client with access to the binary depot.

```
# emerge -gkeav system [Valley Technologies clients have access to a Gentoo binary depot to finish these steps without any compiling at all.]
```

```
# emerge -gkeav world
```

```
# revdep-rebuild
```

Step 4 - (optional) Mask your new GCC and GLIBC

Important: Doing this can save you a lot of time and headache later in life.

Add the following lines to `/etc/portage/package.mask` to prevent them from being updated without you knowing. Replace the versions with the current version you just installed.

```
>sys-devel/gcc-4.1.2
```

```
>sys-libs/glibc-2.6.1
```

Step 5 - (optional) Update/Recompile your kernel

Important: You can probably get by without recompiling your kernel if you used the Valley Technologies 2.6 Gentoo Kernel.

Recompile your Kernel...

Recompile your kernel against your new GCC. For more information see the Gentoo Handbook.

Step 6 - (optional) Remove Old GCC Versions

As usual, substitute `=sys-devel/gcc-3.4*` with the version you want to uninstall.

Important: In case you're upgrading from GCC-3.3, you should run `emerge --oneshot sys-libs/libstdc++-v3` to provide compatibility with older binary C++ applications.

Important: Please note that the GCC 4.1 and newer can't compile a 2.4 kernel. Don't remove your old GCC version if you want to use an older kernel.

```
# emerge -aC =sys-devel/gcc-3.4*
```