

trevorpounds.com 

clean code from the digital bath

• 

23

Jan 10 [Building Older GCC Versions on x86-64 Debian/Ubuntu](#)Posted by Trevor Pounds under [c/c++](#), [compilers](#), [debian](#), [gcc](#), [GNU](#), [i386](#), [linux](#), [multilib](#), [ubuntu](#), [x86-64](#)
[42 comments](#)

Some time ago, while working on [MockitoPP](#), I was interested to see what compilers were compatible with several non-standard compiler dependent techniques (e.g. vtable, data/code pointer sizes) in use. Early on I focused most of my attention on supporting the newer, more popular GCC (i.e. 4.x) and Visual Studio (e.g. VS .Net 2003) compilers. After I hardened the code base a bit I decided to see how well it fared with older versions of GCC. For the purposes of my experiment I decided to test against the latest GCC release from each major series (i.e. 2.95.x to 4.4.x). The build and target system I used for testing was Ubuntu 9.10 on an x86-64 architecture. The main features I wanted were the standard C/C++ languages and, if available, multilib (e.g. i386/x86-64) support. Unfortunately, building an older compiler using a newer tool-chain is not a trivial task in many circumstances; normally a new compiler is compiled using an older tool-chain! One obvious problem is that GCC compiler features have changed over the years which has complicated over time with the advent of more diverse Linux distros. It gets even worse when trying to build a usable 32-bit host/target compiler on a system with a 64-bit tool-chain (see: GCC 2.95.x and 3.0.x below). Additionally, older versions of GCC have limited multilib awareness support on Linux distributions that do not use a "standard" directory structure. For example, in most LSB 64-bit distributions the following lib path convention is used:

```
/lib - 32-bit libraries
/lib64 - 64-bit libraries
```

However, on Debian based systems the following convention is used:

```
/lib - 64-bit libraries
/lib64 - symlink to /lib
/lib32 - 32-bit libraries
```

Unfortunately, this convention confuses the older GCC configuration scripts. However, this limitation has been [resolved](#) in later releases.

GCC >= 4.2.4

To start off, I'll talk about building modern GCC versions. From my limited testing I found that any version starting with 4.2.4 and later can be built by following the standard GCC build procedure outlined on the [web](#). However, for documentation purposes, I've listed the commands I used to build and install GCC 4.4.2 to a custom installation prefix (e.g. /opt).

```
tar xzf gcc-4.4.2.tar.gz
mkdir gcc-4.4.2-objdir
cd gcc-4.4.2-objdir
../gcc-4.4.2/configure --prefix=/opt/x86_64/gcc/gcc-4.4.2 --enable-languages=c,c++
make
make install
```

multilib support is auto-enabled using the above configure options but it can also be explicitly enabled by adding `--enable-multilib` to the argument list.

GCC 3.3.x to 4.2.3

Building versions 3.3.x to 4.2.3 is a little more involved but essentially boils down to applying a slightly modified version of the lib path awareness patch mentioned above. For simplicity's sake I've attached the patch ([gcc-multilib-fix-v3.3.x-to-v4.2.3.debian.x86_64.diff](#)) and used the following commands to build the compiler.

```
tar xzf gcc-3.4.6.tar.gz
cd gcc-3.4.6
patch -p0 < gcc-multilib-fix-v3.3.x-to-v4.2.3.debian.x86_64.diff
mkdir ../gcc-3.4.6-objdir
cd ../gcc-3.4.6-objdir
../gcc-3.4.6/configure --prefix=/opt/x86_64/gcc/gcc-3.4.6 --enable-languages=c,c++
make
make install
```

GCC 3.2.x

Building GCC 3.2.x requires following the same basic steps as above but requires an additional fix that prevents emitting an `.eh_frame` for the `crt*.S.o` objects. The original bug fix can be found on the [GCC mailing list](#). To build this version of GCC, download the attached patch ([gcc-multilib-fix-v3.2.x.debian.x86_64.diff](#)) and perform the following commands.

```
tar xzf gcc-3.2.3.tar.gz
cd gcc-3.2.3
patch -p0 < gcc-multilib-fix-v3.2.x.debian.x86_64.diff
mkdir ../gcc-3.2.3-objdir
cd ../gcc-3.2.3-objdir
../gcc-3.2.3/configure --prefix=/opt/x86_64/gcc/gcc-3.2.3 --enable-languages=c,c++
make
make install
```

multilib i386/x86-64 support is available in this version, however, I seem to recall reading on the web that proper x86-64 support did not become production ready until the 3.3.x releases. At the time of this post I could not find any resource to back this claim. However, I did find the quote, "*The x86-64 port has been significantly improved*," in the [3.3.x change log](#) which potentially confirms this. In any case, I compiled a few simple programs for an x86-64 target and none of them had any run-time problems as far as invalid calculations or heinously crashing but if anyone knows for sure I would be interested to know.

GCC 3.1.x

Up until now most of the builds were fairly easy to "frontport" to run on x86-64 Debian. With GCC 3.1.x, I began to run into more difficult problems. Along with the `.eh_frame` patch used above, this version needed work to fix the `lib32/lib64` path lookup mechanisms for multilib support; FWIW, this is the first GCC version that supported x86-64 host/targets. Also, an additional workaround was needed to forward some older libe symbols (e.g. `__ctype_*`) to their corresponding available equivalents (e.g. `__ctype_*_loc`). I've compiled the corresponding fixes into a patch ([gcc-multilib-fix-v3.1.x.debian.x86_64.diff](#)) which can be used to build this version.

```
tar xzf gcc-3.1.1.tar.gz
cd gcc-3.1.1
patch -p0 < gcc-multilib-fix-v3.1.x.debian.x86_64.diff
mkdir ../gcc-3.1.1-objdir
cd ../gcc-3.1.1-objdir
```

```
../gcc-3.1.1/configure --prefix=/opt/x86_64/gcc/gcc-3.1.1 --enable-languages=c,c++ --enable-threads=posix
make
make install
```

Without the `--enable-threads=posix` flag, the configure script will build using the default thread model (e.g. single). If you do not wish to support posix threads the option can be omitted.

GCC 3.0.x

As I mentioned above, the GCC 3.1.x series of compilers were the first to natively support x86-64 targets. In order to build GCC 3.0.x (for an i386 target) on a x86-64 host the source required a change to the GCC standard library prefix directories (e.g. `/lib` to `/lib32`) and a few tool-chain workarounds (e.g. `-m32` flags). Probably, the most interesting part of the changeset is the addition of `--32` to the GCC assembler spec options. Without the flag, the assembler chokes with an invalid operand error (e.g. `Error: suffix or operands invalid for `push'`). Alternatively, I could have specified -Wa,--32 on the GCC command line to workaround the tool-chain differences but doing this is more burdensome and error prone than having the default behavior compiled into the spec file. Additionally, the same __ctype_* symbol lookup change in the GCC 3.1.x patch is also required. Again, I've provided the compatibility changes in a patch (gcc-v3.0.x.debian.x86_64.diff) which can be built using the following commands.`

```
tar xzf gcc-3.0.4.tar.gz
cd gcc-3.0.4
patch -p0 < gcc-v3.0.x.debian.x86_64.diff
mkdir ../gcc-3.0.4-objdir
cd ../gcc-3.0.4-objdir
../gcc-3.0.4/configure --prefix=/opt/i386/gcc/gcc-3.0.4 --enable-languages=c,c++ --enable-threads=posix --host i386-pc-linux-gnu
make
make install
```

GCC 2.95.x

Finally! We come to the GCC 2.95.x series which, rightfully so, required quite a bit of effort to configure and build for an x86_64 host. About half of the changes needed for GCC 3.0.x are also applicable here (e.g. Makefile fixes, lib32 fix-ups, spec flags, etc). Unfortunately, the other half are workarounds for compiler and glibc nuances that are fixed in later releases. One of the big changes needed is a workaround for the internal use of an `_IO_MTSafe_IO` define. This define causes an error due to a missing header due to difference in glibc headers included with modern Linux distributions (see [here](#) and [here](#) for more info). Luckily, a straightforward workaround is to use glibc's generic `stdio-lock.h` header, which in my case needed to be pulled from the glibc 2.10.1 source to match my installed version. Additionally, another glibc fix is required due to an unnamed union declaration (see this [bug](#)) inside a pthread header which is not supported by GCC 2.95.x. To workaround the issue I pulled in the glibc 2.10.1 system dependent `pthreadtypes.h` header from the source and appended a variable name (e.g. `__gcc_295_workaround__`) to the unnamed union declaration. Build this version using the provided patch ([gcc-v2.95.x.debian.x86_64.diff](#)) and the following instructions.

```
tar xzf glibc-2.10.1.tar.gz
mkdir -p gcc-2.95.3/glibc-workaround/include/bits
cp glibc-2.10.1/bits/stdio-lock.h gcc-2.95.3/glibc-workaround/include/bits
cp glibc-2.10.1/nptl/sysdeps/unix/sysv/linux/x86_64/bits/pthreadtypes.h gcc-2.95.3/glibc-workaround/include/bits
sed -i -n '1h;11H;${};g;s/\(pthread_elist_t __list;\n[ \t]*\)\)/\1 __gcc_295_workaround__g;p;}' gcc-2.95.3/glibc-workaround/include/bits/pthreadtypes.h
tar xzf gcc-2.95.3.tar.gz
cd gcc-2.95.3
patch -p0 < gcc-v2.95.x.debian.x86_64.diff
mkdir ../gcc-2.95.3-objdir
cd ../gcc-2.95.3-objdir
../gcc-2.95.3/configure --prefix=/opt/i386/gcc/gcc-2.95.3 --enable-languages=c,c++ --enable-threads=posix --enable-shared --host i386-pc-linux-gnu
make
make install
```

Be sure to use the same version of the glibc source for the patches as the one of the target system. Since my target/host system uses glibc 2.10.1 I made sure to pull the corresponding source.

Additional Notes

I used GCC 3.4.6 in all cases when building the GCC 2.95.x and 3.x.x versions. The host compiler can be configured by setting the CC environment variable to the full path of your compiler (e.g. `CC=/opt/x86_64/gcc/gcc-3.4.6/bin/gcc`). All of the GCC 4.x.x versions can pretty much be built with any of the older GCC compilers. However, your mileage may vary in some circumstances. In those cases you can always try to build with a different host compiler when the current one is not working out. What seemed to work best for me was building each subsequent version in descending order and bootstrapping a failed build using an older GCC host if necessary.

You can follow any comments to this entry through the [RSS 2.0](#) feed. You can [leave a comment](#), or [trackback](#) from your own site.

42 comments

1. [Pratik Agarwal](#)
[February 8, 2010 at 3:11 pm](#)

I have a 32 bit machine and I am running Ubuntu 9.10 on it. I have the latest gcc compiler. The problem that I am facing is that I have a library developed in 2002. The code is correct for sure but when I try to build it, it gives me errors like `iostream.h` cannot be found, error: `fstream.h`: No such file or directory etc etc.

Can these errors be solved if solved I install gcc 3.0 ? If yes how to install it.

2. [Trevor Pounds](#)
[February 8, 2010 at 4:47 pm](#)

@Pratik Agarwal: Do you know what compiler your library was previously built against? What version of GCC are you trying to build with?

FWIW, you may be able to install an older version of GCC using a previous Ubuntu version's .deb package. For instance you could download the .deb for Jaunty [here](#). If that doesn't work out for you, you could always follow my guide to build an older version of GCC. However, keep in mind that my instructions and patches were made to build older GCC versions on an x86-64 host/target system so some of the steps may not apply in some cases. As a starting point I would recommend building GCC 3.4.6 first and see if that is sufficient for your library. If not you can always work your way backwards from there.

3. [Hugo](#)
[March 15, 2010 at 6:39 pm](#)

Hi, Trevor,

I am trying to build old gcc on Ubuntu 8.04. I have tried the gcc 2.95 and gcc 3.1.1. I follow the method and use the gcc-3.4 to compile it. Both of them meet the same problem with `/usr/bin/ld`.

When compiling gcc-2.95, the problem is
`/usr/bin/ld: cannot find -lgcc`
when compiling gcc-3.1.1, the error is
`/usr/bin/ld: cannot find -lc`

I am not quite sure what is the problem. Do you have any experience of that?

Feel free to give me message to my email.

4. [Trevor Pounds](#)
[March 16, 2010 at 9:21 am](#)

@Hugo: What command are you using to configure the builds? Building gcc-3.1.1 should be doable using 3.4.6 to bootstrap the process. Make sure you have build-essential and gcc-3.4 installed.

```
sudo apt-get install build-essential gcc-3.4
```

You should be able to build and install gcc-3.1.1 using the build steps in the article once these packages are installed. The following configure command should force gcc-3.1.1 to build using 3.4 as the bootstrap compiler. Note the use of CC before the configure command to force a particular bootstrap compiler version.

```
CC=/usr/bin/gcc-3.4 ../gcc-3.1.1/configure --prefix=/opt/x86_64/gcc/gcc-3.1.1 --enable-languages=c,c++ --enable-threads=posix
```

5. [Hugo](#)
[March 16, 2010 at 11:31 am](#)

I have installed gcc3.4 properly. What I do before is to change the CC variable in the Makefile directly, I also try your command. The same error occurs.

For your information, I just remember that I did some modification on file /usr/include/gnu/stubs.h.

I comment off

```
# include
```

The reason that I did that is that I do not find stubs-32.h under this folder in my machine (There is stubs-64.h). The machine is x64 i7 cpu. If I don't comment off that "include", the error will be something like could not find stubs-32.h

Hope this helps.

6. [Trevor Pounds](#)
[March 16, 2010 at 3:17 pm](#)

@Hugo: It sounds like you are missing the 32-bit development headers for glibc. Try installing the libc6-dev-i386 package and reconfiguring/building.

```
sudo apt-get install libc6-dev-i386
```

7. [Hugo](#)
[March 16, 2010 at 5:55 pm](#)

Thanks a lot! I made it. It works.

8. [sonal](#)
[April 30, 2010 at 2:18 pm](#)

Dear Trevor,
 thanks for this detailed workflow — I tried to use it but keep getting this error. What am I doing incorrectly here? I am building on Ubuntu Karmic.

```
/usr/include/bits/fcntl2.h:51: error: call to '__open_missing_mode' declared with attribute error: open with O_CREAT in second argument needs 3 arguments
```

9. [Trevor Pounds](#)
[May 1, 2010 at 10:21 am](#)

@sonal: I need a bit more information on the development environment before I can help. What host architecture are you building on (e.g. i386, x86_64)? What version of GCC are you trying to build? What command are you using to build it?

10. [sonal](#)
[May 3, 2010 at 9:35 am](#)

Hi Trevor,

My host architecture is: x86_64
 I am trying to build GCC 3.4.6
 I am using the commands as you listed them above

thanks,
 sonal

11. [Trevor Pounds](#)
[May 3, 2010 at 11:03 am](#)

@sonal: It looks like this is an issue related to the the compilation flag `-D_FORTIFY_SOURCE=2` added starting with Ubuntu 8.10. You can read more about it on the Ubuntu wiki @ http://wiki.ubuntu.com/CompilerFlags#-D_FORTIFY_SOURCE=2.

However, the easiest way to work around the issue is to override the implicit compiler flag as part of the configure command as follows:

```
CFLAGS=-D_FORTIFY_SOURCE=0 ../gcc-3.4.6/configure --prefix=/opt/x86_64/gcc/gcc-3.4.6 --enable-languages=c,c++
```

12. [inno](#)
[November 16, 2010 at 6:15 pm](#)

Hi Trevor,

My OS is Ubuntu 10.04 (Lucid Lynx) 64-bit
 My host architecture is: amd x86_64
 I tried to build GCC 3.4.6 as per your instructions above.

However, I get the following error:

```
ar: argv.o: No such file or directory
```

```
make[1]: *** [libiberty.a] Error 1
make[1]: Leaving directory `/home/inno/workspaces/toolchains/gcc-3.4.6-objdir/libiberty'
make: *** [all-libiberty] Error 2
```

Please may you tell me how to fix this?

Thanks in advance,
Inno.

13. *inno*

[November 16, 2010 at 11:38 pm](#)

After deleting all files and re-running your steps it worked, so the problem is solved (rather avoided). However I now have a different problem.

I tried to compile a cross compiler by following the same steps. But this time I added `-target=arm-uc-linux-elf` to as configure option.

```
The compilation fails with:
c-cppbuiltin.o: In function `c_cpp_builtins':
c-cppbuiltin.c:(.text+0xf7d): undefined reference to `LINUX_TARGET_OS_CPP_BUILTINS'
collect2: ld returned 1 exit status
make[1]: *** [cc1] Error 1
make[1]: Leaving directory `/home/inno/workspaces/toolchains/gcc-3.4.6-objdir/gcc'
make: *** [all-gcc] Error 2
```

Any help much appreciated.

Thanks in advance
Inno.

14. *Trevor Pounds*

[November 22, 2010 at 1:20 pm](#)

@inno: I haven't tried building a cross compiler for this target. The steps in this article are specific to building x86-64/i386 compilers on a x86-64 host system. If you get it working feel free to provide your steps here.

15. *mba*

[January 2, 2011 at 1:07 pm](#)

hey thanks for a nice document, I am trying to install gcc-2.95.3 in ubuntu 9.04. my machine is i686. I follow the procedure and download and cp glibc-2.10.1/bits/stdio-lock.h gcc-2.95.3/glibc-workaround/include/bits
Is there need to copy the other thing which you mention. I thinks its for debain not for ubuntu. I am new to linux, kindly guide me in detail because i need to install gcc-2.95.3 in ubuntu 9.04. Because major problem is error: lowlevellock.h not there.
Thanks in advance.

16. *Trevor Pounds*

[January 2, 2011 at 2:41 pm](#)

@mba: Your welcome! I tried to provide instructions as succinctly as possible to get baseline GCC compilers working on newer Debian/Ubuntu releases. The steps provided here were written for Ubuntu 9.10 on x86-64 architecture, but they should work reasonably well on newer Debian/Ubuntu releases and architectures as well. The biggest issue with building really old GCC versions (e.g. 2.95.x) is due to the GCC internals' tight coupling with "private" glibc methods. This appears to be much improved in later GCC releases. Please follow all of the steps within the GCC 2.95.x section in order to build this release. The procedure should be the bare minimum required to build that release. Please post back if you are still experiencing issues and I will try to help you further.

17. *mba*

[January 2, 2011 at 7:31 pm](#)

Trevor thanks for quick response. I follow the procedure and in the end i got error message:
In file included from /usr/include/sys/types.h:270,
from system.h:129,
from /gencheck.c:22:
/usr/include/bits/pthreadtypes.h:99: warning: unnamed struct/union that defines no instances
make[1]: *** [gencheck.o] Error 1
make[1]: Leaving directory `/home/mba56/gcc/gcc-2.95.3/gcc'
make: *** [all-gcc] Error 2
Trevor your help would be really really appreciated.
Thanks

18. *Trevor Pounds*

[January 4, 2011 at 9:50 pm](#)

@mba: It appears you have not applied the workaround for GCC 2.95.x's lack of unnamed union support I mentioned above. Be sure to follow all of the steps for GCC 2.95.x exactly to ensure the compiler builds properly.

19. *mba*

[January 11, 2011 at 10:31 pm](#)

Thanks Trevor, I tried many times and follow same steps, but don't know what's wrong, because sometimes i got error like:
Appending ../gcc-2.95.3/libstdc++/config/linux.mt to target-mkfrag
target-mkfrag is unchanged
Cannot find the GNU C library minor version number.
If it didn't display this error, then same error which i mentioned in my previous post.
/usr/include/bits/pthreadtypes.h:99: warning: unnamed struct/union that defines no instances
If you have any suggestion, please let me know.
Thanks in advance.

20. [Trevor Pounds](#)
[January 12, 2011 at 9:25 am](#)

@mba: Sorry I have not been of much help. Can please reply with the steps you are using? I will try to follow them exactly to reproduce the error on my end.

21. [mba](#)
[January 14, 2011 at 10:48 pm](#)

Thanks Trevor for considering my problem. Thanks alot.
 My machine has ubuntu 9.04 32 bit, i686 and it has glibc 2.9. So I am using glibc-2.9. Then following the exactly the same steps you mentioned in your article, I dont change anything.
 Even once i tried with glibc-2.10 but alas didn't work.
 Once i changed this line
 cp glibc-2.10.1/nptl/sysdeps/unix/sysv/linux/x86_64/bits/pthreadtypes.h gcc-2.95.3/glibc-workaround/include/bits
 by replacing /linux/x86_64 by /linux/i386.
 But that didn't work too.
 Your favour and suggestions are really appreciative. thanks.

22. [mba](#)
[January 22, 2011 at 11:02 am](#)

Hey Trevor Did you try the steps which I mentioned. Kindly let me know about it. Thanks

23. [Xin](#)
[January 28, 2011 at 10:36 pm](#)

Hi, Trevor,
 Thanks for your instructions for installation of GCC.
 I want to install gcc2.95.3 on my 32-bit ubuntu 10.04. I followed your steps but came across the following error when make:

```
In function 'open',
inlined from 'collect_execute' at ././gcc-2.95.3/gcc/collect2.c:1762:
/usr/include/bits/fcntl2.h:51: error: call to '__open_missing_mode' declared with attribute error: open with O_CREAT in second argument needs 3 arguments
make[1]: *** [collect2.o] Error 1
make[1]: Leaving directory '/home/yelavende/gcc_download/gcc-2.95.3-objdir/gcc'
make: *** [all-gcc] Error 2
```

Do you know how to fix it?
 Thanks in advance.

Best Regards

Xin

24. [Trevor Pounds](#)
[January 29, 2011 at 12:03 pm](#)

@mba: Sorry, I haven't had a chance to look at your problem in great detail yet. I will try to take a look at some point this week.

@Xin: Have a look at my reply to [sonal above](#). I believe you are running into the same issue related to the default `_FORTIFY_SOURCE` setting.

25. [mba](#)
[January 29, 2011 at 11:36 pm](#)

Hey Trevor, No problem at all. Now the problem/error is this:
 When I compile without using compiler flag I get the same error, which is mentioned by Xin.
 But when I compile by using `-D_FORTIFY_SOURCE=0`
 I got this error

```
Configuring in i686-pc-linux-gnu/libstdc++
Appending ./././gcc-2.95.3/libstdc++/config./././config/mh-x86pic to target-mkfrag
Appending ./././gcc-2.95.3/libstdc++/config/linux.ml to target-mkfrag
Appending ./././gcc-2.95.3/libstdc++/config/linux.mt to target-mkfrag
Cannot find the GNU C library minor version number.
Configuring in i686-pc-linux-gnu/libf2c
creating cache ./config.cache
checking if compiler f771 has been built... no
Configuring in i686-pc-linux-gnu/libchill
creating cache ./config.cache
checking if compiler cc1chill has been built... no
Configuring in i686-pc-linux-gnu/libobjc
creating cache ./config.cache
checking if compiler cc1objc has been built... no
I am sending you these errors , may be you will have some information/guess regarding its solution.
thanks.
```

26. [Eduardo](#)
[February 20, 2011 at 11:10 am](#)

Hello,

I am trying to compile gcc-3.4.6 on a Debian 6 (Squeeze) x86_64 machine, and I get this error:

```
/usr/bin/ld: /root/gcc-compile/gcc/libgcc_s.so.1: version `GCC_4.2.0' not found (required by /usr/lib/libstdc++.so.6)
collect2: ld returned 1 exit status
make[3]: *** [32/libgcc_s_32.so] Error 1
```

```
make[3]: Leaving directory `/root/gcc-compile/gcc'
make[2]: *** [stmp-multilib] Error 2
make[2]: Leaving directory `/root/gcc-compile/gcc'
make[1]: *** [stage1_build] Error 2
make[1]: Leaving directory `/root/gcc-compile/gcc'
make: *** [bootstrap] Error 2
```

Thanks

27. [Trevor Pounds](#)
[February 21, 2011 at 9:47 am](#)

@Eduardo: What options are you using for configure and what GCC version are you using to compile?

28. [Matthew Orlinski](#)
[May 4, 2011 at 4:42 am](#)

Great work. Helped me a whole bunch.

29. [Geovana](#)
[June 15, 2011 at 2:52 pm](#)

Good evening, some time I try to compile the gcc-2.95.3 on Ubuntu 10.10 and 8.04. By chance I found your page and the truth I have followed step by step, but I still leave errors. First compiled in native make bootstrap

```
/usr/include/bits/pthreadtypes.h:99: warning: unnamed struct/union that defines no instances
```

```
make[2]: *** [gencheck.o] Error 1
make[2]: se sale del directorio «/home/geovana/Descargas/gcc-2.95.3-objdir/gcc»
make[1]: *** [bootstrap] Error 2
make[1]: se sale del directorio «/home/geovana/Descargas/gcc-2.95.3-objdir/gcc»
make: *** [bootstrap] Error 2
```

second: make compilation

```
gcc -DIN_GCC -g -O2 -m32 -DHAVE_CONFIG_H -I. -I../gcc-2.95.3/gcc -I../gcc-2.95.3/gcc/config -I../gcc-2.95.3/gcc/.include \
```

```
-DTARGET_MACHINE="i686-pc-linux-gnu" \
```

```
-c `echo ../gcc-2.95.3/gcc/collect2.c | sed 's,^/,,'`
```

```
In file included from /usr/include/fcntl.h:252,
```

```
from ../gcc-2.95.3/gcc/system.h:207,
```

```
from ../gcc-2.95.3/gcc/collect2.c:30:
```

```
In function 'open',
```

```
inlined from 'collect_execute' at ../gcc-2.95.3/gcc/collect2.c:1762:
```

```
/usr/include/bits/fcntl2.h:51: error: call to '__open_missing_mode' declared with attribute error: open with O_CREAT in second argument needs 3 arguments
```

```
make[1]: *** [collect2.o] Error 1
```

```
make[1]: se sale del directorio «/home/geovana/Descargas/gcc-2.95.3-objdir/gcc»
```

```
make: *** [all-gcc] Error 2
```

I have tried these commands CFLAGS=-D_FORTIFY_SOURCE=0 and unset IO_MTSafe_IO

I hope you can help.

Greetings Geovana

30. [Trevor Pounds](#)
[June 21, 2011 at 10:12 am](#)

@Matthew Orlinski: Glad I could help out!

@Geovana: I can't tell from your post what the first error is. The second error definitely looks related to the -D_FORTIFY_SOURCE issue I mentioned above. Be sure to add -D_FORTIFY_SOURCE=0 to the CFLAGS for your build.

31. [Jonathan](#)
[June 23, 2011 at 8:41 am](#)

Nice page, very helpful. I am trying to build 4.2.x under 4.4.3. I was using a --disable-multilib which I had previously needed to build 4.2.x under 4.3.x. But, I kept getting an error at the moment of building libiberty. Your comments and examples above prompted me to omit the disable-multilib, and it gets much farther into the compile, but now it fails when it attempts to use -m32 for MULTILIB_CFLAGS. Meanwhile, I had already defined CFLAGS="-m64" from my previous efforts build under 4.3.x. Somehow between disable-multilib and CFLAGS, I have some inconsistency trying to build 4.2.x under 4.4.3. Any thoughts appreciated, thanks!

32. [Trevor Pounds](#)
[June 30, 2011 at 4:06 pm](#)

@Jonathan: Can you please provide some additional information of the host/target environment? The CFLAGS should not be unnecessary if you are trying to build a target compiler for the same system as the host (i.e. building x86_64 target compiler on x86_64 host). You can also try building 4.2.x with another compiler to see if that helps.

33. [Rekha Jain](#)
[February 9, 2012 at 9:31 pm](#)

Hello,

This is a very informative page. I am trying to compile gcc-3.1.1 on Linux lenlin78 2.6.18-92.el5 #1 SMP Tue Apr 29 13:16:15 EDT 2008 x86_64 x86_64 x86_64 GNU/Linux. But while compiling I received the following errors.

```
/home/cdot/Desktop/gcc-3.1.1/gcc/xgcc -B/home/cdot/Desktop/gcc-3.1.1/gcc/ -nostdinc++ -L/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3/src -L/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3/src/.libs -B/usr/local/x86_64-unknown-linux-gnu/bin/ -B/usr/local/x86_64-unknown-linux-gnu/lib/ -isystem /usr/local/x86_64-unknown-linux-gnu/include -nostdinc++ -L/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3/include/x86_64-unknown-linux-gnu -L/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3/include -L../libsupc++ -L../libmath -g -O2 -D_GNU_SOURCE -fno-implicit-templates -Wall -Wno-format -W -Wwrite-strings -Winline -fdiagnostics-show-location=once -ffunction-sections -fdata-sections -g -c c++locale.cc -fPIC -DPIC -o .libs/c++locale.o
c++locale.cc: In function 'void std::__convert_to_v(const char*, _Tv&,'
```

```

std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = long int]':
c++locale.cc:51: `__strtol_l' undeclared (first use this function)
c++locale.cc:51: (Each undeclared identifier is reported only once for each
function it appears in.)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = long unsigned
int]':
c++locale.cc:69: `__strtolu_l' undeclared (first use this function)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = long long
int]':
c++locale.cc:87: `__strtoll_l' undeclared (first use this function)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = long long
unsigned int]':
c++locale.cc:106: `__strtoull_l' undeclared (first use this function)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = float]':
c++locale.cc:124: `__strtof_l' undeclared (first use this function)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = double]':
c++locale.cc:141: `__strtod_l' undeclared (first use this function)
c++locale.cc: In function `void std::_convert_to_v(const char*, _Tv&,
std::_ios_iostate&, __locale_struct* const&, int) [with _Tv = long double]':
c++locale.cc:158: `__strtold_l' undeclared (first use this function)
c++locale.cc: In static member function `static void
std::locale::facet::_S_create_c_locale(__locale_struct*&, const char*,
__locale_struct*)':
c++locale.cc:170: `__newlocale' undeclared (first use this function)
c++locale.cc: In static member function `static void
std::locale::facet::_S_destroy_c_locale(__locale_struct*&)':
c++locale.cc:180: `__freelocale' undeclared (first use this function)
c++locale.cc: In static member function `static __locale_struct*
std::locale::facet::_S_clone_c_locale(__locale_struct*&)':
c++locale.cc:184: `__duplocale' undeclared (first use this function)
make[3]: *** [c++locale.lo] Error 1
make[3]: Leaving directory `/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3/src'
make[2]: *** [all-recursive] Error 1
make[2]: Leaving directory `/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3'
make[1]: *** [all-recursive-am] Error 2
make[1]: Leaving directory `/home/cdot/Desktop/gcc-3.1.1/x86_64-unknown-linux-gnu/libstdc++-v3'
make: *** [all-target-libstdc++-v3] Error 2
how to solve this problem as i have to compile this urgently.
Thanks

```

34. [Trevor Pounds](#)
[February 12, 2012 at 11:57 am](#)

@Rekha Jain: I haven't encountered this type of issue before. Can you provide more information about your host platform and the configure options you are using?

35. [Roman](#)
[March 31, 2012 at 4:28 pm](#)

Hi,

Thanks so much for such an informative page!

I'm desperately trying to compile GCC 2.95.3 on Ubuntu 10.04.2, x86_64, glibc 2.11.1.

I've followed all of your instruction and still can't get it to work. ☹

I ran into the issue with CFLAGS=-D_FORTIFY_SOURCE=0 and added that to the 'configure' line. That go me further, but now I'm starting to get errors with these eh_frames, such as:

```
/usr/bin/ld.bfd.real: error in pic/cstrmain.o(.eh_frame); no .eh_frame_hdr table will be created.
```

Thinking your gcc-multilib-fix-v3.2.x.debian.x86_64.diff fix can work here I tried using it as well, to no effect.

I uploaded my 1> (stdout) and 2> (stderr) logs to box.com @ <http://www.box.com/s/5af45dd236ee6290f1ea>, could you look them over and help me build this thing?

Thank you so much!

Cheers,
Roman.

36. [Bin](#)
[April 5, 2012 at 4:09 pm](#)

Hi,

Recently, I'm working on a project using MLC++, an old library created during 1990s. I have to use an old gcc compiler to compile the code. I chose gcc-2.95.3 as you mentioned.

Thank you so much for your instructions. These are working for me on the following machine:

1. x86_64-redhat-linux
2. current gcc version: 4.4.6 20110731 (Red Hat 4.4.6-3) (GCC)
3. new installed old GCC version: gcc-2.95.3
4. Using glibc-2.12.1.tar.gz rather than glibc-2.10.1.tar.gz

I am working on a cluster, so I only have the user right and installed gcc-2.95.3 in my own directory.

Here is my problem:

After I installed the gcc as you said, when I wanted to use it compile a simple c++ code, I got some error message like this:

```
/usr/bin/ld: crt1.o: No such file: No such file or directory
collect2: ld returned 1 exit status
```

The administrator has installed:
glibc-devel-2.12-1.47.el6_2.9.i686.rpm

And I can see the crt1.o in:
/usr/lib/ and /usr/lib64

It works good for the existing GCC 4, while doesn't work for new installed GCC 2.95.3.

What should I do to make the GCC2.95.3 work?

Thanks,

Bin

37. *RafaelMartins*
[July 4, 2012 at 3:30 pm](#)

@Bin, look at Trevor's new post on this issue: <http://www.trevorpounds.com/blog/?p=513>

38. *RafaelMartins*
[July 5, 2012 at 9:58 am](#)

Hm... actually, the new post is for "crt1.o" problems during the building GCC itself, so it doesn't apply. Actually, I also have the same problem: gcc 2.95.3 built fine, but using it to compile anything gives the crt1.o message.

39. *Leighton*
[April 11, 2013 at 9:04 pm](#)

Could there be hope? I have installed debian-6.0.7 (via debian-6.0.7-amd64-netinst.iso) on a core i5 64 bit machine. Compiling matlab R2009b mex files is failing. The word is to use the same gcc as was used to compile matlab, namely gcc-4.2.3 I've downloaded gcc-4.2.3.tar.bz2 from <http://ftp.gnu.org/gnu/gcc/> and unzipped it in /opt (previously empty!) and used your gcc-multilib-fix-v3.3.x-to-v4.2.3.debian.x86_64.diff patch as suggested. I've tried compiling with gcc-4.1 as well as the default gcc-4.4, but get

```
/usr/bin/ld: skipping incompatible /usr/lib/./lib/libc.so when searching for -lc
/usr/bin/ld: skipping incompatible /usr/lib/./lib/libc.a when searching for -lc
/usr/bin/ld: i386:x86-64 architecture of input file `./usr/lib/./lib/crt1.o' is incompatible with i386 output
/usr/bin/ld: i386:x86-64 architecture of input file `./usr/lib/./lib/crti.o' is incompatible with i386 output
/usr/bin/ld: final link failed: Invalid operation
collect2: ld returned 1 exit status
make[4]: *** [32/libgcc_s.so] Error 1
make[4]: Leaving directory `/opt/gcc-4.2.3-objdir/gcc'
make[3]: *** [stmp-multilib] Error 2
make[3]: Leaving directory `/opt/gcc-4.2.3-objdir/gcc'
make[2]: *** [all-stage1-gcc] Error 2
make[2]: Leaving directory `/opt/gcc-4.2.3-objdir'
make[1]: *** [stage1-bubble] Error 2
make[1]: Leaving directory `/opt/gcc-4.2.3-objdir'
make: *** [all] Error 2
```

The locations of relevant directories seem to differ from what is implied in your commands, eg gcc is in /usr/bin and the only x86-64 directory I could locate is /var/lib/dkms/ndiswrapper/1.56/2.6.32-5-amd64/x86_64
Can you shed any light please?

40. *Leighton*
[April 16, 2013 at 4:58 pm](#)

For Debian-6.0.7 my Debian mirror only offers gcc-4.1 gcc-4.3 and gcc-4.4 (no gcc-3.v.x)
Is it possible to build gcc-4.2.3 using one of these?

41. *Bob Nelson*
[January 25, 2014 at 4:45 pm](#)

This is a fascinating read and very well written. I just came across it today and to *some* extent I'm glad I didn't encounter it sooner. I've got a mostly home-grown Linux system loosely based on Linux from Scratch. Everything is kept fully updated, kernel, glibc, toolchain and such. Currently it's glibc-2.18, kernel 3.12.8, GNU C 3.8.2 and LLVM/clang 3.4 on an x86_64 platform.

Being interesting in historic software, particularly Emacs from the Jurassic era, I've found it easiest to build with equally ancient compilers.

I have the last of the 2.X.X (2.95.3) and 3.X.X (3.4.6) series as well as the 4.X.X major versions. I did it all on my own, fiddling with Makefiles where needed and rewriting "specs" for paths to crt objects. It was a real adventure and a great learning experience. I'm now glad to find out that someone else has done this and pursued the sometimes daunting tasks with such zeal.

And, frankly, it's fun to go down memory lane by firing up Emacsen from the days when Linus was still on a 0-dot kernel and SLS ruled the distribution(s).

Many...many...many thanks!

42. *maverick*
[November 29, 2014 at 11:45 am](#)

Hi,

I am trying to build gcc v-3.2 on x86_64 machine by following your instructions using gcc v4.6.3. But I am getting this error.

```
gcc -c -DIN_GCC -g -O2 -W -Wall -Wwrite-strings -Wstrict-prototypes -Wmissing-prototypes -Wtraditional -pedantic -Wno-long-long -DHAVE_CONFIG_H -I. -I../..
/srdirgccv32/gcc -I../.. /srdirgccv32/gcc/. -I../.. /srdirgccv32/gcc/config -I../.. /srdirgccv32/gcc/. /include underscore.c -o underscore.o
gcc: fatal error: braced spec '$' %l %MD:-MD %!o:%b.d}%{o*:%d%*}%} %MMD:-MMD %!o:%b.d}%{o*:%d%*}%} %M} %MM} %MF*} %MG} %MP}
%MQ*} %MT*} %!E:%!M:%!MM:%MDiMMD:%{o*:-MQ %*}%} %!no-gcc:-D__GNUC__=%v1 -D__GNUC_MINOR__=%v2
-D__GNUC_PATCHLEVEL__=%v3 -D__GXX_ABI_VERSION=102} %!{!undef:%!ansi:%!std=%:ep}%{std=gnu*:%p}%} %P} %trigraphs} %Os:-
D__OPTIMIZE_SIZE_}%{O*:%!O0:-D__OPTIMIZE_}%} %fno-inline!O0!O*:-D__NO_INLINE_}%{ffast-math:-D__FAST_MATH_}%{ffreestanding:-
D__STDC_HOSTED__=0} %fno-hosted:-D__STDC_HOSTED__=0} %!{!freestanding:%!fno-hosted:-D__STDC_HOSTED__=1}%} %remap} %g3:-dD} %H} %C
%{fshort-wchar:-U__WCHAR_TYPE__ -D__WCHAR_TYPE__=short\ unsigned\ int} %D*&U*&A*} %i*} %Z %i %EIMMM:%W{o*}%}' is invalid at '$'
compilation terminated.
make[1]: *** [underscore.o] Error 4
```

Could you please help.

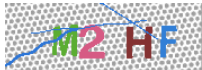
Leave a comment

Name (required)

Email (required)

Website

Comment



CAPTCHA Code *

• About

I'm Trevor, a software developer at Amazon.com, previously I worked for ReachForce and AMD. In my spare time I like drinking craft beers, contributing to open source, and listening to heavy metal.

Social

[@Facebook](#)
[@Google+](#)
[@LinkedIn](#)
[@Twitter](#)

Development

[@code.google.com](#)
[@github.com](#)

• Tags

[annotations](#) [c/c++ compilers](#) [debian](#) [dom](#) [gcc](#) [glibc](#) [GNU](#) [i386](#) [java](#) [jdk](#) [jni](#) [junitbench](#) [libstdc++](#) [linkers](#) [linux](#) [multilib](#) [sccons](#) [symbol versioning](#) [symver](#) [ubuntu](#) [x86-64](#) [xml](#)