Compiling for Bifrost

Jens Låås, UU

uclibc

Bifrost binaries are preferably built using uclibc instead of the standard glibc.

uclibc has a number of advantages for us:

- Smaller binaries with static linking.
- Reduces external dependencies for statically linked programs. In particular uclibc does not use dlopen to access code for nss (name service switch).
- Very good POSIX implementation.

You use uclibc by installing a complete chroot environment that is made available by the developers of uclibc.

http://www.uclibc.org/

mini-root

http://www.uclibc.org/downloads/binaries/0.9.30.1/mini-native-i586.tar.bz2
http://www.uclibc.org/downloads/binaries/0.9.30.1/mini-native-x86_64.tar.bz2

Download and unpack at a suitable place.

Create a small script to facilitate access to the chroot:
$ echo "setarch i586 chroot /home/uclibc/mini-native-i586 /bin/chroot-setup.sh" > chroot-i586.sh
The 'setarch i586' bit is to fool the compiler that we are a 32-bit system even if we are running a 64-bit kernel.
$ chmod +x chroot-i586.sh

To compile x86_64 64-bit binaries use the x86_64 chroot environment. Do not use 'setarch' in the script. You also have to be running a 64-bit kernel.

Now that you have your chroot setup with a script for easy usage you can do:
root@laas:/home/uclibc# ./chroot-i586.sh
Type exit when done.
/ # bash
bash-3.2# cd /usr/src/
And you are all set to go!

Fooling libtool

Libtool is a common problem encountered when you start building static binaries.
This is due to the fact that libtool sifts the LDFLAGS and will not pass *-static* at the linking stage!
Instead they have invented their own pseudo-flag -all-static for this. The problem is that almost none of the packages have configure scripts that accept this flag. So it appears that we are stuck.

Never fear though! This is easily fixed by applying a tiny patch to libtool itself. The libtool script is generated by configure. The libtool 'source' is usually called ltmain.sh.

So start by looking for ltmain.sh:
$ find . -name ltmain.sh

Configure help for Bifrost

You may use a script to help you pass the basic needed flags to the configure script.

B-configure:
#!/bin/bash
CFLAGS="-march=i586 -Os -g" LDFLAGS="-static" ./configure $@
Link: http://laas.mine.nu/uclibc/B-configure

To configure and compile a Bifrost package:

$ cd /usr/src/<package-source>
$ ../B-configure --prefix=/opt/package-name
$ make install

Creating a Bifrost packageIf you want Bifrost to handle the package automatically at boot you need to add a few things to the package:

Complete package under /opt/X:

/opt/X/etc/config.flags/X
/opt/X/rc.d/rc.X

You need to edit 'rc.d/rc.<pkg>' to actually start and stop the service.

Troublemakers

- libtool
- Makefiles (may need to add -static)
- ctype
- pthread
- libdl
- libutil
- PIC (Position independent code)

Configure options

--disable-nls
--enable-static-link
--enable-shared=no
--disable-shared-plugins
--prefix=/opt/<pkg>
--enable-static=yes
--disable-dynamicplugin
--disable-tls // IMPORTANT when option is available (e2fsprogs)
--disable-pie // (quagga)