https://sourceware.org/gdb/wiki/BuildingOnDarwin

Building GDB for Darwin

Creating the binary for Darwin isn't very difficult. Download a release snapshot or get the current source via git/CVS/FTP, then configure and make as usual.

Building the 7.0 release unfortunately results in many "warning: format not a string literal and no format arguments" warnings. This problem has been fixed in CVS. To avoid such warnings building 7.0, configure with --disable-intl.

Giving gdb permission to control other processes

If you try to use your freshly built gdb, you will get an error message such as:

```
Starting program: /x/y/foo
Unable to find Mach task port for process-id 28885: (os/kern) failure
(0x5).
 (please check gdb is codesigned - see taskgated(8))
```

This is because the Darwin kernel will refuse to allow gdb to debug another process if you don't have special rights, since debugging a process means having full control over that process, and that isn't allowed by default since it would be exploitable by malware. (The kernel won't refuse if you are root, but of course you don't want to be root to debug.)

The most up to date method to allow gdb to control another process is to sign it with any system-trusted code signing authority. This is an easy process once you have a certificate (see the section below). If the certificate is known as `gdb-cert`, just use:

```
$ codesign -s gdb-cert gdb
```

Old notes: In Tiger, the kernel would accept processes whose primary effective group is procmod or procview. That means that making gdb setgid procmod should work. Later versions of Darwin should accept this convention provided that taskgated (the daemon that control the access) is invoked with option '-p'. This daemon is configured by `/System/Library/LaunchDaemons/com.apple.taskgated.plist`. I was able to use this rule provided that I am also a member of the procmod group.

Creating a certificate

Start Keychain Access application (`/Applications/Utilities/Keychain Access.app`)

Open menu `/Keychain Access/Certificate Assistant/Create a Certificate...`

Choose a name (`gdb-cert` in the example), set `Identity Type` to

Self Signed Root, set Certificate Type to Code Signing and select the Let me override defaults. Click several times on Continue until you get to the Specify a Location For The Certificate screen, then set Keychain to System.

If you can't store the certificate in the System keychain, create it in the login keychain, then exported it. You can then imported it into the System keychain.

Finally, using the contextual menu for the certificate, select Get Info, open the Trust item, and set Code Signing to Always Trust.

You must quit Keychain Access application in order to use the certificate and restart taskgated service by killing the current running taskgated process (so before using gdb).