Subject: Re: Using Valgrind output

Posted by Novo on Wed, 12 Jun 2019 13:21:07 GMT

View Forum Message <> Reply to Message

slashupp wrote on Wed, 12 June 2019 02:18Novo,

I finally went so far as to install clang & llvm ..

set clang as compiler in Build methods and

added -fsanitize=leak to Debug link options for testing

compiled my project and try to run in debugger, but get:

Quote: Failed to obtain information about threads. Make sure your application posses debug info.

The

debugger and debugge proceses will be stoped!

I've set all to Full in [Build::Output mode], so what is this 'debug info'

it is complaining about and how do I fix it?

I attached my own build method for memory sanitizer.

I'm using "common options", both compile and link.

.USEMALLOC is required with only some of sanitizers.

AFAIK, -fsanitize=leak is a limited version of the memory sanitizer.

Make sure your application posses debug info.

It is likely that your app for some reason doesn't contain debug info.

It is easy to check:

- 1) find location of your executable by either
- a) looking at the build console output, or
- b) Build --> "Open output directory"
- 2) run "file ./your_app"

you should see the output similar to one below.

\$ file ./my_app

./my_app: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=d5ee88481e49f7a1820a7f23711a256a27bbf01f, for GNU/Linux 3.2.0, stripped

The critical part is the word stripped

If you see it, then debug info was stripped out of your executable, or you compiled it without debug info enabled.

you should see not stripped if you want a debugger to work.

Sanitizers print out error messages into cerr, I believe, so you need to run your app from a console to be able to see the output. (or you can change settings in "Debug" --> "Run options")

what is this 'debug info'

'debug info' is a debug info level in a build method (Full/Minimal/None), or the same thing in Output mode

In "Output mode" left part of the dialog is responsible for the Debug configuration, and the right part is responsible for Release ...

Hope this helps.

File Attachments
1) CLANGcpp17msan.bm, downloaded 297 times