Introduction : GDB?

- GNU Project Debugger

- Find bugs, and fix them!
GDB Debugging Process

1. Compile a program for gdb
2. Start gdb!
3. Set breakpoints
4. Run program!
5. Stop at the breakpoint
6. Observe expressions
7. Execute single line of code
8. Find bugs!
9. Fix them!
10. And…
GDB Debugging Process

- Compile a program for gdb
  - gcc -g src.c -o obj
- Start gdb!
  - gdb obj

root@shin-VirtualBox:/home/shin/bomlab_practice/bomb1# gdb bomb
GNU gdb (Ubuntu 7.11.1-0ubuntu1-16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
Find the GDB manual and other documentation resources online at:
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from bomb...done.
(gdb) quit

root@shin-VirtualBox:/home/shin/bomlab_practice/bomb1#
Breakpoint

- Set breakpoints
  - (gdb) break function_name
  - (gdb) line_number

- Show breakpoint
  - (gdb) info break

- Delete breakpoint
  - (gdb) delete breakpoint_number
Watchpoint

- Set watchpoint for an expression
  - (gdb) watch expression
  - (gdb) watch $rax

- Show watchpoint
  - (gdb) info watch

- Delete watchpoint
  - (gdb) delete breakpoint_number
List & disassemble

- List parts of the source code
  - (gdb) list
  - (gdb) list function_name
  - (gdb) list line_number

- Examine the assembly code
  - (gdb) disas
  - (gdb) disas function_name
  - (gdb) disas address
Let’s run it!

- Run the program
  - (gdb) run
  - (gdb) run arglist < input > output
- Continue the program
  - (gdb) continue
- Run until current function returns
  - (gdb) finish
- Stop the program
  - (gdb) kill
Step

- Step into current source line
  - (gdb) step
  - (gdb) step number_of_line

- Step over current source line
  - (gdb) next
  - (gdb) next number_of_line

- Step into current instruction
  - (gdb) stepi
  - (gdb) stepi number_of_line

- Step over current instruction
  - (gdb) nexti
  - (gdb) nexti number_of_line
Display & Print

- Display value of expression
  - Display expression every time the program pauses
  - (gdb) disp expression

- Show display list
  - (gdb) info disp

- Undisplay expression
  - (gdb) undisp display_number
Display & Print

- Print value of expression
  - (gdb) p expression
  - (gdb) p $eax
- (gdb) p constant
- (gdb) p /x expression
  - Result in hexadecimal form
- (gdb) p /t expression
  - Result in Binary form
- (gdb) p *(long *) address
  - Result in long form
Examine Memory

Memory reference & display.

- (gdb) x address
- (gdb) x $rax
- (gdb) x function_name
- (gdb) x/2g address
- (gdb) x/20b address
Useful Information

- (gdb) info frame
  - Information about current stack frame
- (gdb) info registers
  - Values of all the registers
- (gdb) info locals
  - Print all local variables
- (gdb) help
  - Get information about gdb