GDB Basics

A walkthrough with example

- *Cyware* 20

What is GDB?

- The “GNU Debugger”
- Supports several languages like C/C++, Go, Rust, Assembly
- Useful in dynamic analysis
- Can be useful is following situations
  - Start your program, specifying anything that might affect its behavior
  - Make your program stop on specified conditions
  - Examine what has happened, when your program has stopped
  - Change things in your program, so you can experiment with correcting the effects of one bug and go on to learn about another

GDB Commands

- **help** <command>

  Displays help page for different command

- **[b]reak** <function name or *memory address>

  Sets a breakpoint on either a function or the instruction located at a particular addr

- **[d]elete** <breakpoint>

  Removes a breakpoint. Use **i b** for breakpoint info

- **[i]nfo** (about)

  about can be:

  - [f]rame - List info about current stack frame
  - [s]tack - List the stack backtrace, function calls that have been made
  - [r]egisters - List the contents of each register
  - [b]reak - List the breakpoints
  - [fu]nctions - List all functions with addresses
  - [p]roc - Shows additional info about process
    - [map]pings - List memory regions mapped by the specified process

- **[disas]semble** <function name>

  Disassemble a specified section of memory

- **[r]un** (arg1 arg2 …. argn)

  Runs the executable with arguments

- **[c]ontinue**

  Resumes program execution until next breakpoint

- **[s]tep**

  Step program until it reaches a different source line

- **[s]tep[i]**

  Steps through a single x86 instruction

- **[n]ext**

  Unlike "step", if the current source line calls a subroutine,

  this command does not enter the subroutine, but instead steps over

  the call, in effect treating it as a single source line

- **[n]ext[i]**

  Steps through a single x86 instruction

- **[file]** <filename of executable>

  Loads the specified file into gdb

- **[p]rint** <$register or function>

  Print value of register or function

- **[x]/(number)(format)** <address>

  Examines the data located in memory at address

Dynamic analysis of a crackme