Basic Debugger Session

To debug a program, start radare with the `-d` option. Note that you can attach to a running process by specifying its PID, or you can start a new program by specifying its name and parameters:

```
$ pidof mc
32220
$ r2 -d 32220
$ r2 -d /bin/ls
$ r2 -a arm -b 16 -d gdb://192.168.1.43:9090
...
```

In the second case, the debugger will fork and load the debugee `ls` program in memory.

It will pause its execution early in `ld.so` dynamic linker. As a result, you will not yet see the entrypoint or any shared libraries at this point.

You can override this behavior by setting another name for an entry breakpoint. To do this, add a radare command `e dbg.bep=entry` or `e dbg.bep=main` to your startup script, usually it is `~/.config/radare2/radare2rc` .

Another way to continue until a specific address is by using the `dcu` command. Which means: "debug continue until" taking the address of the place to stop at. For example:

```
dcu main
```

Be warned that certain malware or other tricky programs can actually execute code before `main()` and thus you'll be unable to control them. (Like the program constructor or the tls initializers)

Below is a list of most common commands used with debugger:

```
> d?            ; get help on debugger commands
> ds 3          ; step 3 times
> db 0x8048920  ; setup a breakpoint
> db -0x8048920 ; remove a breakpoint
> dc            ; continue process execution
> dcs           ; continue until syscall
> dd            ; manipulate file descriptors
> dm            ; show process maps
> dmp A S rwx   ; change permissions of page at A and size S
> dr eax=33     ; set register value. eax = 33
```

There is another option for debugging in radare, which may be easier: using visual mode.

That way you will neither need to remember many commands nor to keep program state in your mind.

To enter visual debugger mode use `Vpp` :