

Disassembling

Disassemble a function `disassemble vuln`

Disassemble at address `disassemble 0x400566`

Running

Run until termination or breakpoint `r`

Run and pause at `main()` `start`

Run and provide arguments `r arg1 arg2`

If binary prompts for input once through `stdin`, pass input via file `r < in.txt`

If binary prompts for input more than once through `stdin` `r < <(echo "input1"; echo "input2")`

Stepping

Continue execution `c`

Execute next instruction and step over a function `ni`

Execute instruction and step into a function `si`

Breakpoints

Set breakpoint at function `bp vuln`

Set breakpoint at address `bp 0x4005b5`

Set breakpoint at function + offset `bp vuln+47`

List breakpoints `bl`

Delete all breakpoints `d br`

Disable breakpoint 2 `bd 2`

Enable breakpoint 2 `be 2`

Examining data

Examine two 8-byte values at RBP in hex `x/2gx $rbp`

Examine 10 instructions at `main+25` `x/10i *main+25`

Examine 4-bytes of RAX in hex `x/wx $rax`

Print R10 in decimal `p/d $r10`

Print sum of 0x500 and 0x39 in decimal `p/d 0x500 + 0x39`

Print the address of `vuln()` `p vuln`

Using the **x** or **p** command followed by the **size** of the data to examine, and **format** letters

Sizes include **byte**, **word**, **halfword**, and **giant**.

Format letters include **octal**, **hex**, **decimal**, **instruction**, **char**, and **string**.

Modifying data

Set the RAX register to 5 `set $rax = 5`

Set the value pointed to by an address to 5 `set *0x7fffffe280 = 5`

Set the value pointed to by RAX-8 to 5 `set *($rax-8) = 5`

Print out state of the program

`context`

Get address of saved return pointer

Return address of current stack frame `x/gx $rbp+8`

Discovered return addresses on the stack `retaddr`

Search for a string in memory

Look for "Hello" `search Hello`

Get distance between addresses

Using `p` `p/d 0x7fffffe278 - 0x7fffffe220`

Using `distance` `distance 0x7fffffe220 0x7fffffe278`

Note that using **distance** reverses the operands.

Print hexdump

`hexdump $rbp`

Display stack

View the stack `stack`

View 30 rows of the stack `stack 30`

Print virtual memory map pages

`vmmap`

Check security settings

`checksec`

C

By **superkojiman**
(cactuarnation)

cheatography.com/cactuarnation/

Not published yet.

Last updated 3rd February, 2018.

Page 1 of 1.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopadd.com>