LESSON 11



ASM

Debug with GDB

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

ASSEMBLY LANGUAGE

LESSON 11

Debugging with GDB and objdump

Breakpoints, stepping, inspecting registers and memory, disassembly and reading binar

Starting GDB

terminal

```
$ nasm -f elf64 -g -F dwarf sum.asm && ld sum.o -o sum
$ gdb ./sum
(gdb) break _start
(gdb) run
Breakpoint 1, 0x0000000000401000 in _start ()
(gdb) layout regs
```

- Assemble with dash g to include debug information. Start GDB with your program, set a …
- layout asm and layout regs show the instructions and registers in split windows.

01

Stepping Through Code

GDB command	Does
`break label`	stop at a label
`stepi · si`	one instruction
`nexti · ni`	step over calls
`continue · c`	run to next break
`info registers`	show all registers

- stepi runs one instruction. nexti also runs one, but steps over calls. continue runs to the …
- Watch the registers change after each step. This is the best way to learn what instructions …

02

Inspecting Registers and Memory

terminal

```
(gdb) print/x $rax
$1 = 0x158
(gdb) x/4dw &scores
0x402000: 90  85  77  92
(gdb) x/s &msg
0x402010: "Salaam, World!\n"
(gdb) x/4xg $rsp
```

- print slash x dollar rax shows a register in hex. The x command examines memory: x slash …
- x slash s shows a string, and x slash eight x g dollar rsp shows the top of the stack.

03

objdump: Disassembly

terminal

```
$ objdump -d -M intel hello
0000000000401000 <_start>:
  401000:  b8 01 00 00 00     mov    eax,0x1
  401005:  bf 01 00 00 00     mov    edi,0x1
  40100a:  48 be 00 20 40 00  movabs rsi,0x402000
  401014:  0f 05              syscall
```

- objdump dash d disassembles a binary: it shows each instruction's address, its machine-code …
- Add dash M intel for the Intel syntax we use in this course, instead of the default AT and …

04

More Tools

terminal

```
$ strace ./hello
write(1, "Salaam, World!\n", 15) = 15
exit(0)                          = ?
$ readelf -S hello | grep -E 'text|data'
  [ 1] .text   PROGBITS  0000000000401000
  [ 2] .data   PROGBITS  0000000000402000
```

- strace shows every system call a program makes. readelf shows sections and symbols. The …
- Ghidra, a free reverse-engineering tool, turns binaries back into readable C-like code.

05

Lesson Map


1
Starting GDB
2
Stepping Through Code
3
Inspecting Registers and Memory
4
objdump: Disassembly
5
More Tools

Sources and Further Reading

terminal

```
$ nasm -f elf64 -g -F dwarf sum.asm && ld sum.o -o sum
$ gdb ./sum
(gdb) break _start
(gdb) run
Breakpoint 1, 0x0000000000401000 in _start ()
(gdb) layout regs
```

ASSEMBLY LANGUAGE · 11

terminal

```
(gdb) print/x $rax
$1 = 0x158
(gdb) x/4dw &scores
0x402000: 90  85  77  92
(gdb) x/s &msg
0x402010: "Salaam, World!\n"
(gdb) x/4xg $rsp
```

Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari