Assignment3 20131329 신선우 (전기전자컴퓨터공학부/학생)

Phase\_1

gdb breakpoint phase\_1 -> callq <strings\_not\_equal>

before this break point

mov $0x402260, %esi

x/s $0x402260 = “When a problem comes along, you must zip it!”

phase\_1 clear

**Answer : When a problem comes along, you must zip it**

phase\_2

<+9> callq <read\_six\_numbers> -> input is six number

<+14> cmpl $0x0 -> first number is 0

<+18> jns <phase\_2+56> -> No boom, goto +56

<+56> lea -> second number

<+61> mov -> %ebx = 1

<+27> mov -> %eax = %ebx

<+29> add -> first number + %eax = %eax

<+32> cmp -> second number = %eax

<+42> add -> %ebx = %ebx + 1

<+45> add -> address pointer add

<+49> cmp -> if %ebx=6?

Because of this is loop so we can expect

**Answer : 0 1 3 6 10 15**

Phase\_3

<+4> lea -> %rcx = address of %rsp + 8 -> second number

<+9> lea -> %rdx = address of %rsp + c -> first number

<+19> mov -> %eax = 0

<+29> cmp $0x1,%eax

<+32> jg -> 1>0 -> then goto +39

<+39> cmpl $0x7,0xc(%rsp)

<+44> ja -> first number > 7 -> then boom

<+46> mov -> %eax = first number -> first number = 0~7

<+50> jmpq -> switch argument

<+123> cmp 0x8(%rsp),%eax -> second number = 0x39d, 0x113, 0x354, 0x20a, 0x248, 0x169 0x3a6

<+127> je -> equal then end.

Multiple choice

**Answer = 0 925, 2 275, 3 852, 4 522, 5 584, 6 361, 7 934**

Phase\_4

<+34> cmpl $0xe,0xc(%rsp) -> compare first number and 11

<+39> jbe -> first number > 14 -> then boom

<+70> cmpl $0xb,0x8(%rsp) -> compare 11 and second number /// second number = 11 always

<+75> je 0x400fc7 <phase\_4+82> -> if same then pass

**Answer = 0 11**

Phase\_5

<+8>: callq 0x401241 <string\_length>

<+13>: cmp $0x6,%eax

<+16>: je 0x401022 <phase\_5+86> -> length6 string

Breaking point on 0x000000000040100d <+65>: callq 0x40125e <strings\_not\_equal>

x/s $rsi = “devils”

open bomb by text then there is “maduiersnfotvbyl”

<+29> and -> %edx // a=0x1, b=0x2, c=0x3……

<+46> cmp $0x6, %rax

<+50> jne -> six alphabet convert to number

The code seems like substitution cipher

m a d u I e r s n f o t v b y l

a b c d e f g h I j k l m n o

Then devils => beldog

**Answer = beldog**

Phase\_6

<+20>: callq 0x401381 <read\_six\_numbers> -> input is six number

<+41>: sub $0x1,%eax

<+44>: cmp $0x5,%eax

<+47>: jbe 0x401065 <phase\_6+54> -> if inputnumber is bigger than (5+1) then boom!

<+49>: callq 0x40135f <explode\_bomb>

In the last loop

There is a node 1~6

So guessing with node

Input = 1 2 3 4 5 6

0x6032f0 <node1>: 0x000003cd 0x00000001 0x00000000

0x603300 <node2>: 0x000003b6 0x00000002 0x006032f0

0x603310 <node3>: 0x00000124 0x00000003 0x00603300

0x603320 <node4>: 0x000000e4 0x00000004 0x00603310

0x603330 <node5>: 0x00000331 0x00000005 0x00603320

0x603340 <node6>: 0x00000386 0x00000006 0x00603330

By descending order

Node1 – node2 – node6 – node5 – node3 – node4

I can’t understand why 7 – nodenumber is answer

I put number landomly and found answer

**Answer = 6 5 1 2 4 3**