**Phase 1:**

000000000040145d <phase\_1>:

40145d: 53 push %rbx

40145e: 48 89 fb mov %rdi,%rbx

401461: 80 7f 01 00 cmpb $0x0,0x1(%rdi)

401465: 75 07 jne 40146e <phase\_1+0x11>

401467: 80 3b 79 cmpb $0x79,(%rbx)

40146a: 75 09 jne 401475 <phase\_1+0x18>

40146c: 5b pop %rbx

40146d: c3 retq

40146e: e8 bf 07 00 00 callq 401c32 <explode\_bomb>

401473: eb f2 jmp 401467 <phase\_1+0xa>

401475: e8 b8 07 00 00 callq 401c32 <explode\_bomb>

40147a: eb f0 jmp 40146c <phase\_1+0xf>

**Put rbx on the stack**

Mov rdi which is the input into rbx

Compare the input with 0

—> jump if rdi was not 0 to exploding the bomb

**Compare $0x79 with the input at rbx**

—-> if not equal explode bomb

**Otherwise pop from the stack and return**

**So the input for phase 1 should be ‘y’**

**Phase 2:**

00000000004014c7 <phase\_2>:

4014c7: 53 push %rbx

4014c8: 48 83 ec 10 sub $0x10,%rsp

4014cc: c7 44 24 0c 00 00 00 movl $0x0,0xc(%rsp)

4014d3: 00

4014d4: c7 44 24 08 00 00 00 movl $0x0,0x8(%rsp)

4014db: 00

4014dc: 48 8d 54 24 08 lea 0x8(%rsp),%rdx

4014e1: 48 8d 74 24 0c lea 0xc(%rsp),%rsi

4014e6: e8 3a 0a 00 00 callq 401f25 <read\_two\_numbers>

4014eb: 8b 44 24 08 mov 0x8(%rsp),%eax

4014ef: 39 44 24 0c cmp %eax,0xc(%rsp)

4014f3: 74 1e je 401513 <phase\_2+0x4c>

4014f5: 8b 7c 24 0c mov 0xc(%rsp),%edi

4014f9: e8 7e ff ff ff callq 40147c <func\_switch>

4014fe: 89 c3 mov %eax,%ebx

401500: 8b 7c 24 08 mov 0x8(%rsp),%edi

401504: e8 73 ff ff ff callq 40147c <func\_switch>

401509: 39 c3 cmp %eax,%ebx

40150b: 75 0d jne 40151a <phase\_2+0x53>

40150d: 48 83 c4 10 add $0x10,%rsp

401511: 5b pop %rbx

401512: c3 retq

401513: e8 1a 07 00 00 callq 401c32 <explode\_bomb>

401518: eb db jmp 4014f5 <phase\_2+0x2e>

40151a: e8 13 07 00 00 callq 401c32 <explode\_bomb>

40151f: eb ec jmp 40150d <phase\_2+0x46>

000000000040147c <func\_switch>:

40147c: 83 ff 07 cmp $0x7,%edi

40147f: 77 2d ja 4014ae <func\_switch+0x32>

401481: 89 ff mov %edi,%edi

401483: ff 24 fd 80 31 40 00 jmpq \*0x403180(,%rdi,8)

40148a: b8 03 01 00 00 mov $0x103,%eax

40148f: c3 retq

401490: b8 bf 00 00 00 mov $0xbf,%eax

401495: c3 retq

401496: b8 37 00 00 00 mov $0x37,%eax

40149b: c3 retq

40149c: b8 81 01 00 00 mov $0x181,%eax

4014a1: c3 retq

4014a2: b8 dd 00 00 00 mov $0xdd,%eax

4014a7: c3 retq

4014a8: b8 25 02 00 00 mov $0x225,%eax

4014ad: c3 retq

4014ae: 48 83 ec 08 sub $0x8,%rsp

4014b2: e8 7b 07 00 00 callq 401c32 <explode\_bomb>

4014b7: b8 00 00 00 00 mov $0x0,%eax

4014bc: 48 83 c4 08 add $0x8,%rsp

4014c0: c3 retq

4014c1: b8 cd 00 00 00 mov $0xcd,%eax

4014c6: c3 retq

Printing the jump table and seeing that the values that print the same value are 0 and 6

Answer: 0 6

**Phase 3:**

0000000000401521 <phase\_3>:

401521: 41 55 push %r13

401523: 41 54 push %r12

401525: 55 push %rbp

401526: 53 push %rbx

401527: 48 83 ec 28 sub $0x28,%rsp

40152b: 48 89 e6 mov %rsp,%rsi

40152e: e8 8e 09 00 00 callq 401ec1 <read\_six\_numbers>

401533: 8b 2c 24 mov (%rsp),%ebp

401536: 44 8b 64 24 04 mov 0x4(%rsp),%r12d

40153b: 89 e8 mov %ebp,%eax

40153d: 44 09 e0 or %r12d,%eax

401540: 74 07 je 401549 <phase\_3+0x28>

401542: bb 02 00 00 00 mov $0x2,%ebx

401547: eb 10 jmp 401559 <phase\_3+0x38>

401549: e8 e4 06 00 00 callq 401c32 <explode\_bomb>

40154e: eb f2 jmp 401542 <phase\_3+0x21>

401550: 83 c3 01 add $0x1,%ebx

401553: 44 89 e5 mov %r12d,%ebp

401556: 45 89 ec mov %r13d,%r12d

401559: 83 fb 05 cmp $0x5,%ebx

40155c: 7f 16 jg 401574 <phase\_3+0x53>

40155e: 48 63 c3 movslq %ebx,%rax

401561: 44 8b 2c 84 mov (%rsp,%rax,4),%r13d

401565: 44 01 e5 add %r12d,%ebp

401568: 44 39 ed cmp %r13d,%ebp

40156b: 74 e3 je 401550 <phase\_3+0x2f>

40156d: e8 c0 06 00 00 callq 401c32 <explode\_bomb>

401572: eb dc jmp 401550 <phase\_3+0x2f>

401574: 44 89 25 f9 52 00 00 mov %r12d,0x52f9(%rip) # 40687 4 <last>

40157b: 48 83 c4 28 add $0x28,%rsp

40157f: 5b pop %rbx

401580: 5d pop %rbp

401581: 41 5c pop %r12

401583: 41 5d pop %r13

401585: c3 retq

Input: 1 1 2 3 5 8

## 

0000000000401586 <phase\_4>:

401586: 48 83 ec 18 sub $0x18,%rsp  **allocate 24 bytes on the stack**

40158a: 48 8d 54 24 0c lea 0xc(%rsp),%rdx

40158f: 48 8d 74 24 08 lea 0x8(%rsp),%rsi

401594: e8 8c 09 00 00 callq 401f25 <read\_two\_numbers>

401599: 8b 44 24 0c mov 0xc(%rsp),%eax

40159d: 83 e8 02 sub $0x2,%eax

4015a0: 83 f8 02 cmp $0x2,%eax

4015a3: 77 17 ja 4015bc <phase\_4+0x36>

4015a5: 8b 44 24 08 mov 0x8(%rsp),%eax

4015a9: 83 e0 0f and $0xf,%eax

4015ac: 89 44 24 08 mov %eax,0x8(%rsp)

4015b0: ba 00 00 00 00 mov $0x0,%edx

4015b5: b9 00 00 00 00 mov $0x0,%ecx

4015ba: eb 1d jmp 4015d9 <phase\_4+0x53>

4015bc: e8 71 06 00 00 callq 401c32 <explode\_bomb>

4015c1: eb e2 jmp 4015a5 <phase\_4+0x1f>

4015c3: 8b 44 24 08 mov 0x8(%rsp),%eax

4015c7: 01 c1 add %eax,%ecx

4015c9: 48 98 cltq

4015cb: 8b 04 85 40 32 40 00 mov 0x403240(,%rax,4),%eax

4015d2: 89 44 24 08 mov %eax,0x8(%rsp)

4015d6: 83 c2 01 add $0x1,%edx

4015d9: 39 54 24 0c cmp %edx,0xc(%rsp)

4015dd: 7f e4 jg 4015c3 <phase\_4+0x3d>

4015df: 83 f9 15 cmp $0x15,%ecx

4015e2: 75 05 jne 4015e9 <phase\_4+0x63>

4015e4: 48 83 c4 18 add $0x18,%rsp

4015e8: c3 retq

4015e9: e8 44 06 00 00 callq 401c32 <explode\_bomb>

4015ee: eb f4 jmp 4015e4 <phase\_4+0x5e>

(gdb) info registers

rax 0x6 6

rbx 0x1 1

rcx 0x11 17

rdx 0x3 3

rsi 0x3 3

rdi 0x7fffffffd510 140737488344336

rbp 0x7fffffffdcb8 0x7fffffffdcb8

rsp 0x7fffffffdb68 0x7fffffffdb68

r8 0x1999999999999999 1844674407370955161

r9 0x0 0

r10 0x7ffff7d9eac0 140737351641792

r11 0x0 0

r12 0x7fffffffdcb8 140737488346296

r13 0x401316 4199190

r14 0x405e08 4218376

r15 0x7ffff7ffd000 140737354125312

rip 0x401c32 0x401c32 <explode\_bomb>

eflags 0x297 [ CF PF AF SF IF ]

cs 0x33 51

ss 0x2b 43

ds 0x0 0

es 0x0 0

fs 0x0 0

(gdb) x/60c 0x403240 : printing the array

10 2 14 7 8 1 0 11 0 4 1 13 3 9 6

S+array[s]+array[array[s]] == 21

**Phase 5 :GCD**

00000000004015f0 <recurse>:

4015f0: 89 f8 mov %edi,%eax

4015f2: 85 f6 test %esi,%esi

4015f4: 75 01 jne 4015f7 <recurse+0x7>

4015f6: c3 retq

4015f7: 48 83 ec 08 sub $0x8,%rsp

4015fb: 89 f7 mov %esi,%edi

4015fd: 99 cltd

4015fe: f7 fe idiv %esi

401600: 89 d6 mov %edx,%esi

401602: e8 e9 ff ff ff callq 4015f0 <recurse>

401607: 48 83 c4 08 add $0x8,%rsp

40160b: c3 retq

000000000040160c <phase\_5>:

40160c: 55 push %rbp

40160d: 53 push %rbx

40160e: 48 83 ec 18 sub $0x18,%rsp

401612: 48 8d 54 24 08 lea 0x8(%rsp),%rdx

401617: 48 8d 74 24 0c lea 0xc(%rsp),%rsi

40161c: e8 04 09 00 00 callq 401f25 <read\_two\_numbers>

401621: 8b 44 24 08 mov 0x8(%rsp),%eax

401625: 39 44 24 0c cmp %eax,0xc(%rsp)

401629: 74 2a je 401655 <phase\_5+0x49>

40162b: 8b 5c 24 08 mov 0x8(%rsp),%ebx

40162f: 8b 6c 24 0c mov 0xc(%rsp),%ebp

401633: 89 de mov %ebx,%esi

401635: 89 ef mov %ebp,%edi

401637: e8 b4 ff ff ff callq 4015f0 <recurse>

40163c: 83 f8 07 cmp $0x7,%eax

40163f: 75 08 jne 401649 <phase\_5+0x3d>

401641: 39 c3 cmp %eax,%ebx

401643: 74 04 je 401649 <phase\_5+0x3d>

401645: 39 c5 cmp %eax,%ebp

401647: 75 05 jne 40164e <phase\_5+0x42>

401649: e8 e4 05 00 00 callq 401c32 <explode\_bomb>

40164e: 48 83 c4 18 add $0x18,%rsp

401652: 5b pop %rbx

401653: 5d pop %rbp

401654: c3 retq

401655: e8 d8 05 00 00 callq 401c32 <explode\_bomb>

40165a: eb cf jmp 40162b <phase\_5+0x1f>

## 

## 

## 

## 

## 

## 

## **Phase 6**

000000000040165c <init\_treenode>:

40165c: 89 37 mov %esi,(%rdi)

40165e: 48 89 57 08 mov %rdx,0x8(%rdi)

401662: 48 89 4f 10 mov %rcx,0x10(%rdi)

401666: c3 retq

0000000000401667 <foo>:

401667: 53 push %rbx

401668: bb 00 00 00 00 mov $0x0,%ebx

40166d: eb 1b jmp 40168a <foo+0x23>

40166f: bf 18 00 00 00 mov $0x18,%edi

401674: e8 e7 fa ff ff callq 401160 <malloc@plt>

401679: 48 89 c2 mov %rax,%rdx

40167c: 48 63 c3 movslq %ebx,%rax

40167f: 48 89 14 c5 20 66 40 mov %rdx,0x406620(,%rax,8)

401686: 00

401687: 83 c3 01 add $0x1,%ebx

40168a: 83 fb 3f cmp $0x3f,%ebx

40168d: 7e e0 jle 40166f <foo+0x8>

40168f: b8 00 00 00 00 mov $0x0,%eax

401694: eb 6c jmp 401702 <foo+0x9b>

401696: 8d 54 00 01 lea 0x1(%rax,%rax,1),%edx

40169a: 8d 58 01 lea 0x1(%rax),%ebx

40169d: 8d 3c 1b lea (%rbx,%rbx,1),%edi

4016a0: 83 fa 3f cmp $0x3f,%edx

4016a3: 7f 7f jg 401724 <foo+0xbd>

4016a5: 83 ff 3f cmp $0x3f,%edi

4016a8: 0f 8f 80 00 00 00 jg 40172e <foo+0xc7>

4016ae: 41 89 c0 mov %eax,%r8d

4016b1: 41 83 e0 01 and $0x1,%r8d

4016b5: 8d 70 ff lea -0x1(%rax),%esi

4016b8: 89 f1 mov %esi,%ecx

4016ba: c1 e9 1f shr $0x1f,%ecx

4016bd: 01 f1 add %esi,%ecx

4016bf: d1 f9 sar %ecx

4016c1: 3d 82 00 00 00 cmp $0x82,%eax

4016c6: 7f 70 jg 401738 <foo+0xd1>

4016c8: 48 be 73 e3 81 99 de movabs $0x3a5d34de9981e373,%rsi

4016cf: 34 5d 3a

4016d2: 48 d3 ee shr %cl,%rsi

4016d5: 83 e6 01 and $0x1,%esi

4016d8: 48 63 ff movslq %edi,%rdi

4016db: 48 8b 0c fd 20 66 40 mov 0x406620(,%rdi,8),%rcx

4016e2: 00

4016e3: 48 63 d2 movslq %edx,%rdx

4016e6: 48 8b 14 d5 20 66 40 mov 0x406620(,%rdx,8),%rdx

4016ed: 00

4016ee: 44 31 c6 xor %r8d,%esi

4016f1: 48 98 cltq

4016f3: 48 8b 3c c5 20 66 40 mov 0x406620(,%rax,8),%rdi

4016fa: 00

4016fb: e8 5c ff ff ff callq 40165c <init\_treenode>

401700: 89 d8 mov %ebx,%eax

401702: 83 f8 3f cmp $0x3f,%eax

401705: 7e 8f jle 401696 <foo+0x2f>

401707: 48 8b 05 12 4f 00 00 mov 0x4f12(%rip),%rax # 406620 <n>

40170e: c7 00 01 00 00 00 movl $0x1,(%rax)

401714: 48 8b 05 05 4f 00 00 mov 0x4f05(%rip),%rax # 406620 <n>

40171b: 48 89 05 de 4e 00 00 mov %rax,0x4ede(%rip) # 406600 <tree\_root>

401722: 5b pop %rbx

401723: c3 retq

401724: ba 02 00 00 00 mov $0x2,%edx

401729: e9 77 ff ff ff jmpq 4016a5 <foo+0x3e>

40172e: bf 01 00 00 00 mov $0x1,%edi

401733: e9 76 ff ff ff jmpq 4016ae <foo+0x47>

401738: b9 00 00 00 00 mov $0x0,%ecx

40173d: eb 89 jmp 4016c8 <foo+0x61>

000000000040173f <phase\_6>:

40173f: 48 83 ec 18 sub $0x18,%rsp

401743: 48 8d 74 24 0c lea 0xc(%rsp),%rsi

401748: e8 b3 07 00 00 callq 401f00 <read\_one\_number>

40174d: 48 8b 0d ac 4e 00 00 mov 0x4eac(%rip),%rcx # 406600 <tree\_root>

401754: ba 00 00 00 00 mov $0x0,%edx

401759: be 00 00 00 00 mov $0x0,%esi

40175e: eb 0f jmp 40176f <phase\_6+0x30>

401760: 48 8b 49 08 mov 0x8(%rcx),%rcx

401764: 03 31 add (%rcx),%esi

401766: d1 f8 sar %eax

401768: 89 44 24 0c mov %eax,0xc(%rsp)

40176c: 83 c2 01 add $0x1,%edx

40176f: 83 fa 07 cmp $0x7,%edx

401772: 7f 0e jg 401782 <phase\_6+0x43>

401774: 8b 44 24 0c mov 0xc(%rsp),%eax

401778: a8 01 test $0x1,%al

40177a: 74 e4 je 401760 <phase\_6+0x21>

40177c: 48 8b 49 10 mov 0x10(%rcx),%rcx

401780: eb e2 jmp 401764 <phase\_6+0x25>

401782: 83 fe 08 cmp $0x8,%esi

401785: 75 16 jne 40179d <phase\_6+0x5e>

401787: c7 05 df 50 00 00 de movl $0xfacade,0x50df(%rip) # 406870 <p6c>

40178e: ca fa 00

401791: 83 7c 24 0c 21 cmpl $0x21,0xc(%rsp)

401796: 74 0c je 4017a4 <phase\_6+0x65>

401798: 48 83 c4 18 add $0x18,%rsp

40179c: c3 retq

40179d: e8 90 04 00 00 callq 401c32 <explode\_bomb>

4017a2: eb e3 jmp 401787 <phase\_6+0x48>

4017a4: e8 89 04 00 00 callq 401c32 <explode\_bomb>

4017a9: eb ed jmp 401798 <phase\_6+0x59>

Cyclic structure!

**240 yielded 5, so did 255**

**B \*40175e**

**RRRLLLLL → LLLLLRRR**

RRRLLLLLLL

**Summary Table**

| **Phase** | **Input** |
| --- | --- |
| Phase 1 | y |
| Phase 2 | 0 6 |
| Phase 3 | 1 1 2 3 5 8 |
| Phase 4 | 3 3 |
| Phase 5 | 14 49 |
| Phase 6 | 133 |

(gdb) x/6gx 0x4072a0

Value:1

R: 0x00000000004072e0

Value: 1

L:0x407340

Value:1

R: 0x407420

Value:1

L: 0x4075c0

Value:1

L: 0x407900

Value:1

L: 0x4072e0

L:0x407340

Value:1

R: 0x407420

Value:1

**10000101**

**133  
  
Secret Phase**

00000000004017ab <s3cret\_ph4se>:

4017ab: 41 56 push %r14

4017ad: 41 55 push %r13

4017af: 41 54 push %r12

4017b1: 55 push %rbp

4017b2: 53 push %rbx

4017b3: e8 95 07 00 00 callq 401f4d <read\_line>

4017b8: 49 89 c5 mov %rax,%r13

4017bb: bb 01 00 00 00 mov $0x1,%ebx

4017c0: 41 bc 01 00 00 00 mov $0x1,%r12d

4017c6: b8 00 00 00 00 mov $0x0,%eax

4017cb: eb 1f jmp 4017ec <s3cret\_ph4se+0x41>

4017cd: 83 eb 01 sub $0x1,%ebx

4017d0: 44 89 e0 mov %r12d,%eax

4017d3: 89 da mov %ebx,%edx

4017d5: 44 0f b6 b4 d0 40 61 movzbl 0x406140(%rax,%rdx,8),%r14d

4017dc: 40 00

4017de: 41 80 fe ff cmp $0xff,%r14b

4017e2: 74 45 je 401829 <s3cret\_ph4se+0x7e>

4017e4: 41 80 fe 21 cmp $0x21,%r14b

4017e8: 74 46 je 401830 <s3cret\_ph4se+0x85>

4017ea: 89 e8 mov %ebp,%eax

4017ec: 83 f8 63 cmp $0x63,%eax

4017ef: 7f 57 jg 401848 <s3cret\_ph4se+0x9d>

4017f1: 8d 68 01 lea 0x1(%rax),%ebp

4017f4: 48 98 cltq

4017f6: 41 0f b6 44 05 00 movzbl 0x0(%r13,%rax,1),%eax

4017fc: 3c 73 cmp $0x73,%al

4017fe: 74 cd je 4017cd <s3cret\_ph4se+0x22>

401800: 83 e8 62 sub $0x62,%eax

401803: 3c 0d cmp $0xd,%al

401805: 77 1b ja 401822 <s3cret\_ph4se+0x77>

401807: 0f b6 c0 movzbl %al,%eax

40180a: ff 24 c5 c0 31 40 00 jmpq \*0x4031c0(,%rax,8)

401811: 41 83 ec 01 sub $0x1,%r12d

401815: eb b9 jmp 4017d0 <s3cret\_ph4se+0x25>

401817: 83 c3 01 add $0x1,%ebx

40181a: eb b4 jmp 4017d0 <s3cret\_ph4se+0x25>

40181c: 41 83 c4 01 add $0x1,%r12d

401820: eb ae jmp 4017d0 <s3cret\_ph4se+0x25>

401822: e8 0b 04 00 00 callq 401c32 <explode\_bomb>

401827: eb a7 jmp 4017d0 <s3cret\_ph4se+0x25>

401829: e8 04 04 00 00 callq 401c32 <explode\_bomb>

40182e: eb b4 jmp 4017e4 <s3cret\_ph4se+0x39>

401830: bf 80 32 40 00 mov $0x403280,%edi

401835: e8 46 f8 ff ff callq 401080 <puts@plt>

40183a: e8 b8 03 00 00 callq 401bf7 <phase\_defused>

40183f: 5b pop %rbx

401840: 5d pop %rbp

401841: 41 5c pop %r12

401843: 41 5d pop %r13

401845: 41 5e pop %r14

401847: c3 retq

401848: e8 e5 03 00 00 callq 401c32 <explode\_bomb>

40184d: eb f0 jmp 40183f <s3cret\_ph4se+0x94>

**Lookup table:**

(gdb) x/16gx 0x4031c0

0x4031c0: 0x000000000040181c 0x0000000000401822

0x4031d0: 0x0000000000401822 0x0000000000401822

0x4031e0: 0x0000000000401822 0x0000000000401822

0x4031f0: 0x0000000000401822 0x0000000000401822

0x403200: 0x0000000000401822 0x0000000000401822

0x403210: 0x0000000000401822 0x0000000000401811

0x403220: 0x0000000000401822 0x0000000000401817

0x403230: 0x0000000000000000 0x0000000000000000

Symbol Table:  
SYMBOL TABLE:

0000000000400318 l d .interp 0000000000000000 .interp

0000000000400338 l d .note.gnu.property 0000000000000000 .note.gnu.property

0000000000400358 l d .note.gnu.build-id 0000000000000000 .note.gnu.build-id

000000000040037c l d .note.ABI-tag 0000000000000000 .note.ABI-tag

00000000004003a0 l d .gnu.hash 0000000000000000 .gnu.hash

00000000004003d0 l d .dynsym 0000000000000000 .dynsym

0000000000400790 l d .dynstr 0000000000000000 .dynstr

0000000000400944 l d .gnu.version 0000000000000000 .gnu.version

0000000000400998 l d .gnu.version\_r 0000000000000000 .gnu.version\_r

00000000004009f8 l d .rela.dyn 0000000000000000 .rela.dyn

0000000000400aa0 l d .rela.plt 0000000000000000 .rela.plt

0000000000401000 l d .init 0000000000000000 .init

0000000000401020 l d .plt 0000000000000000 .plt

0000000000401230 l d .text 0000000000000000 .text

0000000000402bb0 l d .fini 0000000000000000 .fini

0000000000403000 l d .rodata 0000000000000000 .rodata

0000000000403e70 l d .eh\_frame\_hdr 0000000000000000 .eh\_frame\_hdr

0000000000403fc8 l d .eh\_frame 0000000000000000 .eh\_frame

0000000000405df8 l d .init\_array 0000000000000000 .init\_array

0000000000405e08 l d .fini\_array 0000000000000000 .fini\_array

0000000000405e10 l d .dynamic 0000000000000000 .dynamic

0000000000405fe0 l d .got 0000000000000000 .got

0000000000406000 l d .got.plt 0000000000000000 .got.plt

0000000000406120 l d .data 0000000000000000 .data

00000000004065c0 l d .bss 0000000000000000 .bss

0000000000000000 l d .comment 0000000000000000 .comment

0000000000000000 l d .annobin.notes 0000000000000000 .annobin.notes

0000000000408ec0 l d .gnu.build.attributes 0000000000000000 .gnu.build.attributes

0000000000000000 l d .debug\_aranges 0000000000000000 .debug\_aranges

0000000000000000 l d .debug\_info 0000000000000000 .debug\_info

0000000000000000 l d .debug\_abbrev 0000000000000000 .debug\_abbrev

0000000000000000 l d .debug\_line 0000000000000000 .debug\_line

0000000000000000 l d .debug\_str 0000000000000000 .debug\_str

0000000000000000 l d .debug\_line\_str 0000000000000000 .debug\_line\_str

0000000000000000 l d .debug\_loclists 0000000000000000 .debug\_loclists

0000000000000000 l df \*ABS\* 0000000000000000 /usr/lib/gcc/x86\_64-redhat-linux/11/../../../../lib64/crt1.o

000000000040037c l O .note.ABI-tag 0000000000000020 \_\_abi\_tag

0000000000000000 l df \*ABS\* 0000000000000000 crtstuff.c

0000000000401270 l F .text 0000000000000000 deregister\_tm\_clones

00000000004012a0 l F .text 0000000000000000 register\_tm\_clones

00000000004012e0 l F .text 0000000000000000 \_\_do\_global\_dtors\_aux

00000000004065e8 l O .bss 0000000000000001 completed.0

0000000000405e08 l O .fini\_array 0000000000000000 \_\_do\_global\_dtors\_aux\_fini\_array\_entry

0000000000401310 l F .text 0000000000000000 frame\_dummy

0000000000405df8 l O .init\_array 0000000000000000 \_\_frame\_dummy\_init\_array\_entry

0000000000000000 l df \*ABS\* 0000000000000000 bomb.c

0000000000000000 l df \*ABS\* 0000000000000000 phases.c

0000000000403240 l O .rodata 0000000000000040 array.0

0000000000000000 l df \*ABS\* 0000000000000000 support.c

000000000040184f l F .text 0000000000000051 sig\_handler

0000000000000000 l df \*ABS\* 0000000000000000 driverlib.c

000000000040206e l F .text 0000000000000012 rio\_readinitb

00000000004020a9 l F .text 00000000000000cb urlencode

0000000000402174 l F .text 000000000000005e rio\_writen

00000000004021d2 l F .text 000000000000008a rio\_read

000000000040225c l F .text 000000000000007d rio\_readlineb

0000000000000000 l df \*ABS\* 0000000000000000 crtstuff.c

00000000004045c8 l O .eh\_frame 0000000000000000 \_\_FRAME\_END\_\_

0000000000000000 l df \*ABS\* 0000000000000000

0000000000405e10 l O .dynamic 0000000000000000 \_DYNAMIC

0000000000403e70 l .eh\_frame\_hdr 0000000000000000 \_\_GNU\_EH\_FRAME\_HDR

0000000000406000 l O .got.plt 0000000000000000 \_GLOBAL\_OFFSET\_TABLE\_

00000000004015f0 g F .text 000000000000001c recurse

0000000000401aeb g F .text 0000000000000041 skip

0000000000000000 F \*UND\* 0000000000000000 getenv@GLIBC\_2.2.5

0000000000401f25 g F .text 0000000000000028 read\_two\_numbers

0000000000401bf7 g F .text 000000000000003b phase\_defused

0000000000000000 F \*UND\* 0000000000000000 putchar@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 strcasecmp@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 \_\_libc\_start\_main@GLIBC\_2.34

0000000000000000 F \*UND\* 0000000000000000 \_\_errno\_location@GLIBC\_2.2.5

0000000000000000 w \*UND\* 0000000000000000 \_ITM\_deregisterTMCloneTable

00000000004065c0 g O .bss 0000000000000008 stdout@GLIBC\_2.2.5

0000000000406120 w .data 0000000000000000 data\_start

0000000000406880 g O .bss 0000000000000640 input\_strings

0000000000000000 F \*UND\* 0000000000000000 strcpy@GLIBC\_2.2.5

000000000040147c g F .text 000000000000004b func\_switch

0000000000000000 F \*UND\* 0000000000000000 puts@GLIBC\_2.2.5

0000000000406870 g O .bss 0000000000000004 p6c

00000000004065d0 g O .bss 0000000000000008 stdin@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 write@GLIBC\_2.2.5

0000000000406620 g O .bss 0000000000000200 n

00000000004065c0 g .data 0000000000000000 \_edata

0000000000406140 g O .data 0000000000000040 array\_2d

0000000000402bb0 g F .fini 0000000000000000 .hidden \_fini

0000000000000000 F \*UND\* 0000000000000000 strlen@GLIBC\_2.2.5

0000000000406878 g O .bss 0000000000000004 num\_input\_strings

0000000000000000 F \*UND\* 0000000000000000 printf@GLIBC\_2.2.5

000000000040160c g F .text 0000000000000050 phase\_5

0000000000401aaf g F .text 0000000000000001 initialize\_bomb\_solve

0000000000401ab0 g F .text 000000000000003b blank\_line

0000000000000000 F \*UND\* 0000000000000000 snprintf@GLIBC\_2.2.5

00000000004022d9 g F .text 00000000000006bc submitr

0000000000401f00 g F .text 0000000000000025 read\_one\_number

0000000000000000 F \*UND\* 0000000000000000 gai\_strerror@GLIBC\_2.2.5

0000000000406874 g O .bss 0000000000000004 last

0000000000401521 g F .text 0000000000000065 phase\_3

000000000040145d g F .text 000000000000001f phase\_1

00000000004018a0 g F .text 0000000000000020 invalid\_phase

00000000004029be g F .text 0000000000000185 init\_driver

0000000000000000 F \*UND\* 0000000000000000 alarm@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 close@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 read@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 fgets@GLIBC\_2.2.5

0000000000401c32 g F .text 000000000000028f explode\_bomb

0000000000406120 g .data 0000000000000000 \_\_data\_start

0000000000000000 F \*UND\* 0000000000000000 strcmp@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 signal@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 fprintf@GLIBC\_2.2.5

0000000000000000 w \*UND\* 0000000000000000 \_\_gmon\_start\_\_

0000000000403008 g O .rodata 0000000000000000 .hidden \_\_dso\_handle

0000000000000000 F \*UND\* 0000000000000000 memcpy@GLIBC\_2.14

0000000000403000 g O .rodata 0000000000000004 \_IO\_stdin\_used

00000000004061c0 g O .data 0000000000000400 host\_table

00000000004018c0 g F .text 0000000000000019 string\_length

0000000000000000 F \*UND\* 0000000000000000 malloc@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 fflush@GLIBC\_2.2.5

0000000000401a00 g F .text 00000000000000af welcome\_message

0000000000000000 F \*UND\* 0000000000000000 \_\_isoc99\_sscanf@GLIBC\_2.7

0000000000401667 g F .text 00000000000000d8 foo

0000000000406ec0 g .bss 0000000000000000 \_end

0000000000401260 g F .text 0000000000000005 .hidden \_dl\_relocate\_static\_pie

0000000000401230 g F .text 0000000000000026 \_start

00000000004065f0 g O .bss 0000000000000008 infile

0000000000402080 g F .text 0000000000000029 sigalrm\_handler

0000000000402995 g F .text 0000000000000029 init\_timeout

00000000004065c0 g .bss 0000000000000000 \_\_bss\_start

0000000000406198 g O .data 000000000000000a userid

00000000004017ab g F .text 00000000000000a4 s3cret\_ph4se

0000000000401316 g F .text 0000000000000147 main

0000000000401f4d g F .text 0000000000000121 read\_line

0000000000406600 g O .bss 0000000000000008 tree\_root

00000000004018d9 g F .text 000000000000004b strings\_not\_equal

0000000000401586 g F .text 000000000000006a phase\_4

0000000000000000 F \*UND\* 0000000000000000 fopen@GLIBC\_2.2.5

000000000040173f g F .text 000000000000006c phase\_6

0000000000406820 g O .bss 0000000000000050 scratch

0000000000402b43 g F .text 000000000000006a driver\_post

0000000000401b2c g F .text 00000000000000cb send\_msg

00000000004014c7 g F .text 000000000000005a phase\_2

0000000000000000 F \*UND\* 0000000000000000 gethostname@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 sprintf@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 exit@GLIBC\_2.2.5

00000000004061a4 g O .data 0000000000000004 bomb\_id

0000000000000000 F \*UND\* 0000000000000000 connect@GLIBC\_2.2.5

00000000004065c0 g O .data 0000000000000000 .hidden \_\_TMC\_END\_\_

0000000000000000 w \*UND\* 0000000000000000 \_ITM\_registerTMCloneTable

0000000000000000 F \*UND\* 0000000000000000 getaddrinfo@GLIBC\_2.2.5

0000000000406180 g O .data 0000000000000015 user\_password

0000000000000000 F \*UND\* 0000000000000000 sleep@GLIBC\_2.2.5

0000000000401000 g F .init 0000000000000000 .hidden \_init

0000000000401ec1 g F .text 000000000000003f read\_six\_numbers

0000000000401924 g F .text 00000000000000dc initialize\_bomb

000000000040165c g F .text 000000000000000b init\_treenode

0000000000000000 F \*UND\* 0000000000000000 \_\_ctype\_b\_loc@GLIBC\_2.3

0000000000000000 F \*UND\* 0000000000000000 freeaddrinfo@GLIBC\_2.2.5

00000000004065e0 g O .bss 0000000000000008 stderr@GLIBC\_2.2.5

0000000000000000 F \*UND\* 0000000000000000 socket@GLIBC\_2.2.5

0000000000401c32 <explode\_bomb>:

401c32: 48 81 ec d8 00 00 00 sub $0xd8,%rsp

401c39: 83 3d 38 4c 00 00 06 cmpl $0x6,0x4c38(%rip) # 406878 <num\_input\_strings>

401c40: 77 17 ja 401c59 <explode\_bomb+0x27>

401c42: 8b 05 30 4c 00 00 mov 0x4c30(%rip),%eax # 406878 <num\_input\_strings>

401c48: ff 24 c5 e0 3c 40 00 jmpq \*0x403ce0(,%rax,8)

401c4f: bf d3 38 40 00 mov $0x4038d3,%edi

401c54: e8 27 f4 ff ff callq 401080 <puts@plt>

401c59: 48 b8 53 6f 20 79 6f movabs $0x6d20756f79206f53,%rax

401c60: 75 20 6d

401c63: 48 ba 61 64 65 20 69 movabs $0x7420746920656461,%rdx

401c6a: 74 20 74

401c6d: 48 89 04 24 mov %rax,(%rsp)

401c71: 48 89 54 24 08 mov %rdx,0x8(%rsp)

401c76: 48 b8 6f 20 74 68 65 movabs $0x337320656874206f,%rax

401c7d: 20 73 33

401c80: 48 ba 63 72 65 74 20 movabs $0x6168702074657263,%rdx

401c87: 70 68 61

401c8a: 48 89 44 24 10 mov %rax,0x10(%rsp)

401c8f: 48 89 54 24 18 mov %rdx,0x18(%rsp)

401c94: 48 b8 73 65 2e 20 20 movabs $0x63694e20202e6573,%rax

401c9b: 4e 69 63

401c9e: 48 ba 65 2c 20 62 75 movabs $0x640a747562202c65,%rdx

401ca5: 74 0a 64

401ca8: 48 89 44 24 20 mov %rax,0x20(%rsp)

401cad: 48 89 54 24 28 mov %rdx,0x28(%rsp)

401cb2: 48 b8 6f 6e 27 74 20 movabs $0x6c65742074276e6f,%rax

401cb9: 74 65 6c

401cbc: 48 ba 6c 20 61 6e 79 movabs $0x656e6f796e61206c,%rdx

401cc3: 6f 6e 65

401cc6: 48 89 44 24 30 mov %rax,0x30(%rsp)

401ccb: 48 89 54 24 38 mov %rdx,0x38(%rsp)

401cd0: 48 b8 20 61 62 6f 75 movabs $0x692074756f626120,%rax

401cd7: 74 20 69

401cda: 48 ba 74 20 28 65 73 movabs $0x6365707365282074,%rdx

401ce1: 70 65 63

401ce4: 48 89 44 24 40 mov %rax,0x40(%rsp)

401ce9: 48 89 54 24 48 mov %rdx,0x48(%rsp)

401cee: 48 b8 69 61 6c 6c 79 movabs $0x6e6f20796c6c6169,%rax

401cf5: 20 6f 6e

401cf8: 48 ba 20 50 69 61 7a movabs $0x29617a7a61695020,%rdx

401cff: 7a 61 29

401d02: 48 89 44 24 50 mov %rax,0x50(%rsp)

401d07: 48 89 54 24 58 mov %rdx,0x58(%rsp)

401d0c: 48 b8 2e 0a 49 66 20 movabs $0x756f792066490a2e,%rax

401d13: 79 6f 75

401d16: 48 ba 20 64 6f 20 79 movabs $0x20756f79206f6420,%rdx

401d1d: 6f 75 20

401d20: 48 89 44 24 60 mov %rax,0x60(%rsp)

401d25: 48 89 54 24 68 mov %rdx,0x68(%rsp)

401d2a: 48 b8 77 6f 6e 27 74 movabs $0x65672074276e6f77,%rax

401d31: 20 67 65

401d34: 48 ba 74 20 61 6e 79 movabs $0x786520796e612074,%rdx

401d3b: 20 65 78

401d3e: 48 89 44 24 70 mov %rax,0x70(%rsp)

401d43: 48 89 54 24 78 mov %rdx,0x78(%rsp)

401d48: 48 b8 74 72 61 20 63 movabs $0x6465726320617274,%rax

401d4f: 72 65 64

401d52: 48 ba 69 74 21 20 61 movabs $0x77796e6120217469,%rdx

401d59: 6e 79 77

401d5c: 48 89 84 24 80 00 00 mov %rax,0x80(%rsp)

401d63: 00

401d64: 48 89 94 24 88 00 00 mov %rdx,0x88(%rsp)

401d6b: 00

401d6c: 48 b8 61 79 73 2c 0a movabs $0x6e69660a2c737961,%rax

401d73: 66 69 6e

401d76: 48 ba 64 69 6e 67 20 movabs $0x20746920676e6964,%rdx

401d7d: 69 74 20

401d80: 48 89 84 24 90 00 00 mov %rax,0x90(%rsp)

401d87: 00

401d88: 48 89 94 24 98 00 00 mov %rdx,0x98(%rsp)

401d8f: 00

401d90: 48 b8 61 6e 64 20 73 movabs $0x766c6f7320646e61,%rax

401d97: 6f 6c 76

401d9a: 48 ba 69 6e 67 20 69 movabs $0x6120746920676e69,%rdx

401da1: 74 20 61

401da4: 48 89 84 24 a0 00 00 mov %rax,0xa0(%rsp)

401dab: 00

401dac: 48 89 94 24 a8 00 00 mov %rdx,0xa8(%rsp)

401db3: 00

401db4: 48 b8 72 65 20 71 75 movabs $0x6574697571206572,%rax

401dbb: 69 74 65

401dbe: 48 ba 20 64 69 66 66 movabs $0x6572656666696420,%rdx

401dc5: 65 72 65

401dc8: 48 89 84 24 b0 00 00 mov %rax,0xb0(%rsp)

401dcf: 00

401dd0: 48 89 94 24 b8 00 00 mov %rdx,0xb8(%rsp)

401dd7: 00

401dd8: 48 b8 6e 74 2e 2e 2e movabs $0x2e2e2e746e,%rax

401ddf: 00 00 00

401de2: 48 89 84 24 c0 00 00 mov %rax,0xc0(%rsp)

401de9: 00

401dea: 83 3d 7f 4a 00 00 00 cmpl $0x0,0x4a7f(%rip) # 406870 <p6c>

401df1: 74 0d je 401e00 <explode\_bomb+0x1ce>

401df3: 83 3d 7a 4a 00 00 21 cmpl $0x21,0x4a7a(%rip) # 40687 4 <last>

401dfa: 0f 84 8c 00 00 00 je 401e8c <explode\_bomb+0x25a>

401e00: bf 2d 39 40 00 mov $0x40392d,%edi

401e05: e8 76 f2 ff ff callq 401080 <puts@plt>

401e0a: bf 36 39 40 00 mov $0x403936,%edi

401e0f: e8 6c f2 ff ff callq 401080 <puts@plt>

401e14: bf 00 00 00 00 mov $0x0,%edi

401e19: e8 0e fd ff ff callq 401b2c <send\_msg>

401e1e: bf 58 38 40 00 mov $0x403858,%edi

401e23: e8 58 f2 ff ff callq 401080 <puts@plt>

401e28: bf 08 00 00 00 mov $0x8,%edi

401e2d: e8 8e f3 ff ff callq 4011c0 <exit@plt>

401e32: bf f0 37 40 00 mov $0x4037f0,%edi

401e37: e8 44 f2 ff ff callq 401080 <puts@plt>

401e3c: e9 18 fe ff ff jmpq 401c59 <explode\_bomb+0x27>

401e41: bf d8 38 40 00 mov $0x4038d8,%edi

401e46: e8 35 f2 ff ff callq 401080 <puts@plt>

401e4b: e9 09 fe ff ff jmpq 401c59 <explode\_bomb+0x27>

401e50: bf ed 38 40 00 mov $0x4038ed,%edi

401e55: e8 26 f2 ff ff callq 401080 <puts@plt>

401e5a: e9 fa fd ff ff jmpq 401c59 <explode\_bomb+0x27>

401e5f: bf 03 39 40 00 mov $0x403903,%edi

401e64: e8 17 f2 ff ff callq 401080 <puts@plt>

401e69: e9 eb fd ff ff jmpq 401c59 <explode\_bomb+0x27>

401e6e: bf 1a 39 40 00 mov $0x40391a,%edi

401e73: e8 08 f2 ff ff callq 401080 <puts@plt>

401e78: e9 dc fd ff ff jmpq 401c59 <explode\_bomb+0x27>

401e7d: bf 23 39 40 00 mov $0x403923,%edi

401e82: e8 f9 f1 ff ff callq 401080 <puts@plt>

401e87: e9 cd fd ff ff jmpq 401c59 <explode\_bomb+0x27>

401e8c: 48 89 e7 mov %rsp,%rdi

401e8f: e8 ec f1 ff ff callq 401080 <puts@plt>

401e94: bf 18 38 40 00 mov $0x403818,%edi

401e99: e8 e2 f1 ff ff callq 401080 <puts@plt>

401e9e: e8 54 fd ff ff callq 401bf7 <phase\_defused>

401ea3: c7 05 c7 49 00 00 00 movl $0x0,0x49c7(%rip) # 406874 <last>

401eaa: 00 00 00

401ead: b8 00 00 00 00 mov $0x0,%eax

401eb2: e8 f4 f8 ff ff callq 4017ab <s3cret\_ph4se>

401eb7: bf 00 00 00 00 mov $0x0,%edi

401ebc: e8 ff f2 ff ff callq 4011c0 <exit@plt>

Secret phase:  
End with 33 for phase 3

Modify end input for phase 6