|  |  |
| --- | --- |
|  |  |
|  | Microsoft Windows [Version 10.0.19042.1237]  (c) Microsoft Corporation. All rights reserved.  D:\one drive\OneDrive\Desktop\12190094\Assignment 1\bomb001>bash  tshering@TD:/mnt/d/one drive/OneDrive/Desktop/12190094/Assignment 1/bomb001$ gdb bomb  GNU gdb (Ubuntu 8.1.1-0ubuntu1) 8.1.1 |
|  | Copyright (C) 2021 Free Software Foundation, Inc. |
|  | License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html> |
|  | This is free software: you are free to change and redistribute it. |
|  | There is NO WARRANTY, to the extent permitted by law. |
|  | Type "show copying" and "show warranty" for details. |
|  | This GDB was configured as "x86\_64-linux-gnu". |
|  | Type "show configuration" for configuration details. |
|  | For bug reporting instructions, please see: |
|  | <https://www.gnu.org/software/gdb/bugs/>. |
|  | Find the GDB manual and other documentation resources online at: |
|  | <http://www.gnu.org/software/gdb/documentation/>. |
|  |  |
|  | For help, type "help". |
|  | Type "apropos word" to search for commands related to "word"... |
|  | Reading symbols from bomb... |
|  | (gdb) b phase\_5 |
|  | Breakpoint 1 at 0x401081 |
|  | (gdb) r answers.txt |
|  | Starting program: /mnt/d/one drive/OneDrive/Desktop/12190094/Assignment 1/bomb001/bomb answers.txt |
|  | [Thread debugging using libthread\_db enabled] |
|  | Using host libthread\_db library "/lib/x86\_64-linux-gnu/libthread\_db.so.1". |
|  | Welcome to my fiendish little bomb. You have 6 phases with |
|  | which to blow yourself up. Have a nice day! |
|  | Phase 1 defused. How about the next one? |
|  | That's number 2. Keep going! |
|  | Halfway there! |
|  | So you got that one. Try this one. |
|  | 5 1 **;test input** |
|  |  |
|  | Breakpoint 1, 0x0000000000401081 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | => 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi//answer format: %d %d |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax **;eax holds the number of inputs, eax is compared with 0x1** |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax **; value shouaxld below 0xf(f=15).** |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109**>;if eax and 0xf are equal then bomb explodes** |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax**;compare eax with 0xf** |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72**>;if eax is not equal to 0xf, then the loop starts and the instructions are in loop until the eax value is equal to 0xf. Loop will stop when eax value is equal to 0xf and execute the next instructions.** |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx//edx holds or keep tracking, the number of loops taking place. |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx **;0x4(%rsp) holds the user and ecx holds the expected input and is compared and if its equal bomb does not explode.** |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) x/s 0x4025cf |
|  | 0x4025cf: "%d %d**";input format** |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | => 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u\* 0x00000000004010a7 |
|  | 0x00000000004010a7 in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010aa in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | => 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | **;checking eax or rax value, it should greater than 1.** |
|  | End of assembler dump. |
|  | (gdb) i r |
|  | rax 0x2 2 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x0 0 |
|  | rdx 0x1 1 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010aa 0x4010aa <phase\_5+41> |
|  | eflags 0x202 [ IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | => 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010b1 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | => 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010b4 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | => 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \*0x00000000004010ba |
|  | 0x00000000004010ba in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010bd in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | => 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | //checking eax or rax value, compare eax or rax with 0xf. It should not be equal. |
|  | (gdb) i r |
|  | rax 0x5 5 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x0 0 |
|  | rdx 0x1 1 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010bd 0x4010bd <phase\_5+60> |
|  | eflags 0x297 [ CF PF AF SF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010bf in phase\_5 () |
|  | **;Since the eax value and 0xf is not equal, it executes the next instruction.** |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | => 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | **;checking the eax or rax value at 1st iteration.** |
|  | End of assembler dump. |
|  | (gdb) i r |
|  | rax 0xc 12 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0xc 12 |
|  | rdx 0x1 1 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x206 [ PF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010cc in phase\_5 () |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 2nd iteration.** |
|  | (gdb) i r |
|  | rax 0x3 3 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0xf 15 |
|  | rdx 0x2 2 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x206 [ PF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 3rd iteration**. |
|  | (gdb) i r |
|  | rax 0x7 7 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x16 22 |
|  | rdx 0x3 3 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x212 [ AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 4th iteration.** |
|  | (gdb) i r |
|  | rax 0xb 11 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x21 33 |
|  | rdx 0x4 4 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x216 [ PF AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | => 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 5th iteration**. |
|  | (gdb) i r |
|  | rax 0xd 13 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x2e 46 |
|  | rdx 0x5 5 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x206 [ PF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 6th iteration**. |
|  | (gdb) i r |
|  | rax 0x9 9 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x37 55 |
|  | rdx 0x6 6 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x212 [ AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 7th iteration.** |
|  | (gdb) i r |
|  | rax 0x4 4 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x3b 59 |
|  | rdx 0x7 7 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x202 [ IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | **;checking the eax or rax value at 8th iteration.** |
|  | (gdb) i r |
|  | rax 0x8 8 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x43 67 |
|  | rdx 0x8 8 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x212 [ AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 9th iteration.** |
|  | (gdb) i r |
|  | rax 0x0 0 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x43 67 |
|  | rdx 0x9 9 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x202 [ IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 10th iteration.** |
|  | (gdb) i r |
|  | rax 0x1 1 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x4e 78 |
|  | rdx 0xb 11 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x206 [ PF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | **;checking the eax or rax value at 11th iteration.** |
|  | (gdb) i r |
|  | rax 0x2 2 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x50 80 |
|  | rdx 0xc 12 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x216 [ PF AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 12th iteration.** |
|  | (gdb) i r |
|  | rax 0xe 14 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x5e 94 |
|  | rdx 0xd 13 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x202 [ IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 13th iteration.** |
|  | (gdb) i r |
|  | rax 0x6 6 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x64 100 |
|  | rdx 0xe 14 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x212 [ AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010c9 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | => 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) u \* 0x00000000004010d7 |
|  | 0x00000000004010d7 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **;checking the eax or rax value at 14th iteration.** |
|  | (gdb) i r |
|  | rax 0xf 15 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x73 115 |
|  | rdx 0xf 15 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010d7 0x4010d7 <phase\_5+86> |
|  | eflags 0x212 [ AF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | => 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010da in phase\_5 () |
|  | (gdb) ni |
|  | 0x00000000004010dc in phase\_5 () |
|  | **;Since the eax value is equal to 0xf, the loop stops and executes the next instructions**. |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | => 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010e3 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | => 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) i r |
|  | rax 0xf 15 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x73 115 |
|  | rdx 0xf 15 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010e3 0x4010e3 <phase\_5+98> |
|  | eflags 0x246 [ PF ZF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010e6 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | => 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) ni |
|  | 0x00000000004010e8 in phase\_5 () |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | => 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | **; 0x4(%rsp) holds my second inputs.** |
|  | (gdb) x/d 0x4+$rsp |
|  | 0x7fffffffddb4: 1 |
|  | (gdb) i r |
|  | rax 0xf 15 |
|  | rbx 0x7fffffffdef8 140737488346872 |
|  | rcx 0x73 115//this ecx or rcx holds the second input, the second input is 115. |
|  | rdx 0xf 15 |
|  | rsi 0x1 1 |
|  | rdi 0x7fffffffd760 140737488344928 |
|  | rbp 0x2 0x2 |
|  | rsp 0x7fffffffddb0 0x7fffffffddb0 |
|  | r8 0x0 0 |
|  | r9 0x0 0 |
|  | r10 0x7ffff7f3aac0 140737353329344 |
|  | r11 0x7ffff7f3b3c0 140737353331648 |
|  | r12 0x7fffffffdef8 140737488346872 |
|  | r13 0x400d56 4197718 |
|  | r14 0x0 0 |
|  | r15 0x7ffff7ffbc40 140737354120256 |
|  | rip 0x4010e8 0x4010e8 <phase\_5+103> |
|  | eflags 0x246 [ PF ZF IF ] |
|  | cs 0x33 51 |
|  | ss 0x2b 43 |
|  | ds 0x0 0 |
|  | es 0x0 0 |
|  | fs 0x0 0 |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | gs 0x0 0 |
|  | (gdb) ni |
|  | 0x00000000004010ec in phase\_5 () |
|  | **//bomb explodes since it is not equal** |
|  | (gdb) disas |
|  | Dump of assembler code for function phase\_5: |
|  | 0x0000000000401081 <+0>: sub $0x18,%rsp |
|  | 0x0000000000401085 <+4>: mov %fs:0x28,%rax |
|  | 0x000000000040108e <+13>: mov %rax,0x8(%rsp) |
|  | 0x0000000000401093 <+18>: xor %eax,%eax |
|  | 0x0000000000401095 <+20>: lea 0x4(%rsp),%rcx |
|  | 0x000000000040109a <+25>: mov %rsp,%rdx |
|  | 0x000000000040109d <+28>: mov $0x4025cf,%esi |
|  | 0x00000000004010a2 <+33>: call 0x400bb0 <\_\_isoc99\_sscanf@plt> |
|  | 0x00000000004010a7 <+38>: cmp $0x1,%eax |
|  | 0x00000000004010aa <+41>: jg 0x4010b1 <phase\_5+48> |
|  | 0x00000000004010ac <+43>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010b1 <+48>: mov (%rsp),%eax |
|  | 0x00000000004010b4 <+51>: and $0xf,%eax |
|  | 0x00000000004010b7 <+54>: mov %eax,(%rsp) |
|  | 0x00000000004010ba <+57>: cmp $0xf,%eax |
|  | 0x00000000004010bd <+60>: je 0x4010ee <phase\_5+109> |
|  | 0x00000000004010bf <+62>: mov $0x0,%ecx |
|  | 0x00000000004010c4 <+67>: mov $0x0,%edx |
|  | 0x00000000004010c9 <+72>: add $0x1,%edx |
|  | 0x00000000004010cc <+75>: cltq |
|  | 0x00000000004010ce <+77>: mov 0x402480(,%rax,4),%eax |
|  | 0x00000000004010d5 <+84>: add %eax,%ecx |
|  | --Type <RET> for more, q to quit, c to continue without paging-- |
|  | 0x00000000004010d7 <+86>: cmp $0xf,%eax |
|  | 0x00000000004010da <+89>: jne 0x4010c9 <phase\_5+72> |
|  | 0x00000000004010dc <+91>: movl $0xf,(%rsp) |
|  | 0x00000000004010e3 <+98>: cmp $0xf,%edx |
|  | 0x00000000004010e6 <+101>: jne 0x4010ee <phase\_5+109> |
|  | 0x00000000004010e8 <+103>: cmp 0x4(%rsp),%ecx |
|  | => 0x00000000004010ec <+107>: je 0x4010f3 <phase\_5+114> |
|  | 0x00000000004010ee <+109>: call 0x40143d <explode\_bomb> |
|  | 0x00000000004010f3 <+114>: mov 0x8(%rsp),%rax |
|  | 0x00000000004010f8 <+119>: xor %fs:0x28,%rax |
|  | 0x0000000000401101 <+128>: je 0x401108 <phase\_5+135> |
|  | 0x0000000000401103 <+130>: call 0x400b00 <\_\_stack\_chk\_fail@plt> |
|  | 0x0000000000401108 <+135>: add $0x18,%rsp |
|  | 0x000000000040110c <+139>: ret |
|  | End of assembler dump. |
|  | (gdb) r answers.txt |
|  | The program being debugged has been started already. |
|  | Start it from the beginning? (y or n) y |
|  | Starting program: /mnt/d/one drive/OneDrive/Desktop/12190094/Assignment 1/bomb001/bomb answers.txt |
|  | [Thread debugging using libthread\_db enabled] |
|  | Using host libthread\_db library "/lib/x86\_64-linux-gnu/libthread\_db.so.1". |
|  | Welcome to my fiendish little bomb. You have 6 phases with |
|  | which to blow yourself up. Have a nice day! |
|  | Phase 1 defused. How about the next one? |
|  | That's number 2. Keep going! |
|  | Halfway there! |
|  | So you got that one. Try this one. |
|  | 5 115 **; So solution in phase 5 is: 5 115** |
|  | Good work! On to the next… |
|  |  |
|  |  |
|  |  |