```
Last login: Sat Jan 23 15:09:58 on ttys001
Dhrumils-Air:~ Dhrumil$ ls
Applications     Documents       IdeaProjects    Movies          Pictures
Desktop          Downloads       Library         Music           Public
Dhrumils-Air:~ Dhrumil$ ssh Dpate85@23.99.192.124
Dpate85@23.99.192.124's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-43-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Sat Jan 23 20:43:22 UTC 2016

  System load:  0.0               Processes:           117
  Usage of /:   5.8% of 28.80GB   Users logged in:     0
  Memory usage: 6%                IP address for eth0: 10.2.0.4
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud


Last login: Sat Jan 23 20:43:23 2016 from 104-1-26-208.lightspeed.cicril.sbcglobal.net
Dpate85@Dhrumil:~$ ls
dpate85  public
Dpate85@Dhrumil:~$ cd dpate85
Dpate85@Dhrumil:~/dpate85$ ls
hw1  hw2  README.md
Dpate85@Dhrumil:~/dpate85$ cd hw2
Dpate85@Dhrumil:~/dpate85/hw2$ ls
puzzles
Dpate85@Dhrumil:~/dpate85/hw2$ cd puzzles
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./2
The password is "yes" without quotes if the call to curl_easy_perform was successful; the password is "no" without quotes if the call to curl_easy_perform as unsuccessful.
You may not use gdb to answer this question.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -d 2

2:     file format elf64-x86-64


Disassembly of section .init:

0000000000400600 <_init>:
  400600:       48 83 ec 08             sub    $0x8,%rsp
  400604:       48 8b 05 ed 19 20 00    mov    0x2019ed(%rip),%rax        # 601ff8 <_fini+0x201704>
  40060b:       48 85 c0                test   %rax,%rax
  40060e:       74 05                   je     400615 <_init+0x15>
  400610:       e8 5b 00 00 00          callq  400670 <__gmon_start__@plt>
  400615:       48 83 c4 08             add    $0x8,%rsp
  400619:       c3                      retq

Disassembly of section .plt:

0000000000400620 <curl_easy_init@plt-0x10>:
  400620:       ff 35 e2 19 20 00       pushq  0x2019e2(%rip)        # 602008 <_fini+0x201714>
  400626:       ff 25 e4 19 20 00       jmpq   *0x2019e4(%rip)        # 602010 <_fini+0x20171c>
  40062c:       0f 1f 40 00             nopl   0x0(%rax)

0000000000400630 <curl_easy_init@plt>:
  400630:       ff 25 e2 19 20 00       jmpq   *0x2019e2(%rip)        # 602018 <_fini+0x201724>
  400636:       68 00 00 00 00          pushq  $0x0
  40063b:       e9 e0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400640 <curl_easy_perform@plt>:
  400640:       ff 25 da 19 20 00       jmpq   *0x2019da(%rip)        # 602020 <_fini+0x20172c>
  400646:       68 01 00 00 00          pushq  $0x1
  40064b:       e9 d0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400650 <curl_easy_setopt@plt>:
  400650:       ff 25 d2 19 20 00       jmpq   *0x2019d2(%rip)        # 602028 <_fini+0x201734>
  400656:       68 02 00 00 00          pushq  $0x2
  40065b:       e9 c0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400660 <__libc_start_main@plt>:
  400660:       ff 25 ca 19 20 00       jmpq   *0x2019ca(%rip)        # 602030 <_fini+0x20173c>
  400666:       68 03 00 00 00          pushq  $0x3
  40066b:       e9 b0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400670 <__gmon_start__@plt>:
  400670:       ff 25 c2 19 20 00       jmpq   *0x2019c2(%rip)        # 602038 <_fini+0x201744>
  400676:       68 04 00 00 00          pushq  $0x4
  40067b:       e9 a0 ff ff ff          jmpq   400620 <_init+0x20>

Disassembly of section .text:

0000000000400680 <.text>:
  400680:       31 ed                   xor    %ebp,%ebp
  400682:       49 89 d1                mov    %rdx,%r9
  400685:       5e                      pop    %rsi
  400686:       48 89 e2                mov    %rsp,%rdx
  400689:       48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40068d:       50                      push   %rax
  40068e:       54                      push   %rsp
  40068f:       49 c7 c0 f0 08 40 00    mov    $0x4008f0,%r8
  400696:       48 c7 c1 80 08 40 00    mov    $0x400880,%rcx
  40069d:       48 c7 c7 fd 07 40 00    mov    $0x4007fd,%rdi
  4006a4:       e8 b7 ff ff ff          callq  400660 <__libc_start_main@plt>
  4006a9:       f4                      hlt
  4006aa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  4006b0:       b8 57 20 60 00          mov    $0x602057,%eax
  4006b5:       55                      push   %rbp
  4006b6:       48 2d 50 20 60 00       sub    $0x602050,%rax
  4006bc:       48 83 f8 0e             cmp    $0xe,%rax
  4006c0:       48 89 e5                mov    %rsp,%rbp
  4006c3:       77 02                   ja     4006c7 <__gmon_start__@plt+0x57>
  4006c5:       5d                      pop    %rbp
  4006c6:       c3                      retq
  4006c7:       b8 00 00 00 00          mov    $0x0,%eax
  4006cc:       48 85 c0                test   %rax,%rax
  4006cf:       74 f4                   je     4006c5 <__gmon_start__@plt+0x55>
  4006d1:       5d                      pop    %rbp
  4006d2:       bf 50 20 60 00          mov    $0x602050,%edi
  4006d7:       ff e0                   jmpq   *%rax
  4006d9:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
  4006e0:       b8 50 20 60 00          mov    $0x602050,%eax
  4006e5:       55                      push   %rbp
  4006e6:       48 2d 50 20 60 00       sub    $0x602050,%rax
  4006ec:       48 c1 f8 03             sar    $0x3,%rax
  4006f0:       48 89 e5                mov    %rsp,%rbp
  4006f3:       48 89 c2                mov    %rax,%rdx
  4006f6:       48 c1 ea 3f             shr    $0x3f,%rdx
  4006fa:       48 01 d0                add    %rdx,%rax
  4006fd:       48 d1 f8                sar    %rax
  400700:       75 02                   jne    400704 <__gmon_start__@plt+0x94>
  400702:       5d                      pop    %rbp
  400703:       c3                      retq
  400704:       ba 00 00 00 00          mov    $0x0,%edx
  400709:       48 85 d2                test   %rdx,%rdx
  40070c:       74 f4                   je     400702 <__gmon_start__@plt+0x92>
  40070e:       5d                      pop    %rbp
  40070f:       48 89 c6                mov    %rax,%rsi
```

```
400712:       bf 50 20 60 00          mov    $0x602050,%edi
400717:       ff e2                   jmpq   *%rdx
400719:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
400720:       80 3d 29 19 20 00 00    cmpb   $0x0,0x201929(%rip)        # 602050 <_edata>
400727:       75 11                   jne    40073a <__gmon_start__@plt+0xca>
400729:       55                      push   %rbp
40072a:       48 89 e5                mov    %rsp,%rbp
40072d:       e8 7e ff ff ff          callq  4006b0 <__gmon_start__@plt+0x40>
400732:       5d                      pop    %rbp
400733:       c6 05 16 19 20 00 01    movb   $0x1,0x201916(%rip)        # 602050 <_edata>
40073a:       f3 c3                   repz retq
40073c:       0f 1f 40 00             nopl   0x0(%rax)
400740:       48 83 3d c8 16 20 00    cmpq   $0x0,0x2016c8(%rip)        # 601e10 <_fini+0x20151c>
400747:       00
400748:       74 1e                   je     400768 <__gmon_start__@plt+0xf8>
40074a:       b8 00 00 00 00          mov    $0x0,%eax
40074f:       48 85 c0                test   %rax,%rax
400752:       74 14                   je     400768 <__gmon_start__@plt+0xf8>
400754:       55                      push   %rbp
400755:       bf 10 1e 60 00          mov    $0x601e10,%edi
40075a:       48 89 e5                mov    %rsp,%rbp
40075d:       ff d0                   callq  *%rax
40075f:       5d                      pop    %rbp
400760:       e9 7b ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
400765:       0f 1f 00                nopl   (%rax)
400768:       e9 73 ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
40076d:       55                      push   %rbp
40076e:       48 89 e5                mov    %rsp,%rbp
400771:       5d                      pop    %rbp
400772:       c3                      retq
400773:       55                      push   %rbp
400774:       48 89 e5                mov    %rsp,%rbp
400777:       5d                      pop    %rbp
400778:       c3                      retq
400779:       55                      push   %rbp
40077a:       48 89 e5                mov    %rsp,%rbp
40077d:       5d                      pop    %rbp
40077e:       c3                      retq
40077f:       55                      push   %rbp
400780:       48 89 e5                mov    %rsp,%rbp
400783:       5d                      pop    %rbp
400784:       c3                      retq
400785:       55                      push   %rbp
400786:       48 89 e5                mov    %rsp,%rbp
400789:       5d                      pop    %rbp
40078a:       c3                      retq
40078b:       55                      push   %rbp
40078c:       48 89 e5                mov    %rsp,%rbp
40078f:       5d                      pop    %rbp
400790:       c3                      retq
400791:       55                      push   %rbp
400792:       48 89 e5                mov    %rsp,%rbp
400795:       5d                      pop    %rbp
400796:       c3                      retq
400797:       55                      push   %rbp
400798:       48 89 e5                mov    %rsp,%rbp
40079b:       5d                      pop    %rbp
40079c:       c3                      retq
40079d:       55                      push   %rbp
40079e:       48 89 e5                mov    %rsp,%rbp
4007a1:       5d                      pop    %rbp
4007a2:       c3                      retq
4007a3:       55                      push   %rbp
4007a4:       48 89 e5                mov    %rsp,%rbp
4007a7:       5d                      pop    %rbp
4007a8:       c3                      retq
4007a9:       55                      push   %rbp
4007aa:       48 89 e5                mov    %rsp,%rbp
4007ad:       5d                      pop    %rbp
4007ae:       c3                      retq
4007af:       55                      push   %rbp
4007b0:       48 89 e5                mov    %rsp,%rbp
4007b3:       5d                      pop    %rbp
4007b4:       c3                      retq
4007b5:       55                      push   %rbp
4007b6:       48 89 e5                mov    %rsp,%rbp
4007b9:       5d                      pop    %rbp
4007ba:       c3                      retq
4007bb:       55                      push   %rbp
4007bc:       48 89 e5                mov    %rsp,%rbp
4007bf:       5d                      pop    %rbp
4007c0:       c3                      retq
4007c1:       55                      push   %rbp
4007c2:       48 89 e5                mov    %rsp,%rbp
4007c5:       5d                      pop    %rbp
4007c6:       c3                      retq
4007c7:       55                      push   %rbp
4007c8:       48 89 e5                mov    %rsp,%rbp
4007cb:       5d                      pop    %rbp
4007cc:       c3                      retq
4007cd:       55                      push   %rbp
4007ce:       48 89 e5                mov    %rsp,%rbp
4007d1:       5d                      pop    %rbp
4007d2:       c3                      retq
4007d3:       55                      push   %rbp
4007d4:       48 89 e5                mov    %rsp,%rbp
4007d7:       5d                      pop    %rbp
4007d8:       c3                      retq
4007d9:       55                      push   %rbp
4007da:       48 89 e5                mov    %rsp,%rbp
4007dd:       5d                      pop    %rbp
4007de:       c3                      retq
4007df:       55                      push   %rbp
4007e0:       48 89 e5                mov    %rsp,%rbp
4007e3:       5d                      pop    %rbp
4007e4:       c3                      retq
4007e5:       55                      push   %rbp
4007e6:       48 89 e5                mov    %rsp,%rbp
4007e9:       5d                      pop    %rbp
4007ea:       c3                      retq
4007eb:       55                      push   %rbp
4007ec:       48 89 e5                mov    %rsp,%rbp
4007ef:       5d                      pop    %rbp
4007f0:       c3                      retq
4007f1:       55                      push   %rbp
4007f2:       48 89 e5                mov    %rsp,%rbp
4007f5:       5d                      pop    %rbp
4007f6:       c3                      retq
4007f7:       55                      push   %rbp
4007f8:       48 89 e5                mov    %rsp,%rbp
4007fb:       5d                      pop    %rbp
4007fc:       c3                      retq
4007fd:       55                      push   %rbp
4007fe:       48 89 e5                mov    %rsp,%rbp
400801:       48 83 ec 20             sub    $0x20,%rsp
400805:       e8 26 fe ff ff          callq  400630 <curl_easy_init@plt>
40080a:       48 89 45 f8             mov    %rax,-0x8(%rbp)
40080e:       48 83 7d f8 00          cmpq   $0x0,-0x8(%rbp)
400813:       74 5a                   je     40086f <__gmon_start__@plt+0x1ff>
400815:       c7 45 ec 12 27 00 00    movl   $0x2712,-0x14(%rbp)
40081c:       8b 4d ec                mov    -0x14(%rbp),%ecx
40081f:       48 8b 45 f8             mov    -0x8(%rbp),%rax
400823:       ba 08 09 40 00          mov    $0x400908,%edx
400828:       89 ce                   mov    %ecx,%esi
40082a:       48 89 c7                mov    %rax,%rdi
```

```
  40082d:       b8 00 00 00 00          mov    $0x0,%eax
  400832:       e8 19 fe ff ff          callq  400650 <curl_easy_setopt@plt>
  400837:       c7 45 f0 34 00 00 00    movl   $0x34,-0x10(%rbp)
  40083e:       8b 4d f0                mov    -0x10(%rbp),%ecx
  400841:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400845:       ba 01 00 00 00          mov    $0x1,%edx
  40084a:       89 ce                   mov    %ecx,%esi
  40084c:       48 89 c7                mov    %rax,%rdi
  40084f:       b8 00 00 00 00          mov    $0x0,%eax
  400854:       e8 f7 fd ff ff          callq  400650 <curl_easy_setopt@plt>
  400859:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  40085d:       48 89 c7                mov    %rax,%rdi
  400860:       e8 db fd ff ff          callq  400640 <curl_easy_perform@plt>
  400865:       89 45 f4                mov    %eax,-0xc(%rbp)
  400868:       b8 00 00 00 00          mov    $0x0,%eax
  40086d:       eb 05                   jmp    400874 <__gmon_start__@plt+0x204>
  40086f:       b8 01 00 00 00          mov    $0x1,%eax
  400874:       c9                      leaveq
  400875:       c3                      retq
  400876:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  40087d:       00 00 00
  400880:       41 57                   push   %r15
  400882:       41 89 ff                mov    %edi,%r15d
  400885:       41 56                   push   %r14
  400887:       49 89 f6                mov    %rsi,%r14
  40088a:       41 55                   push   %r13
  40088c:       49 89 d5                mov    %rdx,%r13
  40088f:       41 54                   push   %r12
  400891:       4c 8d 25 68 15 20 00    lea    0x201568(%rip),%r12        # 601e00 <_fini+0x20150c>
  400898:       55                      push   %rbp
  400899:       48 8d 2d 68 15 20 00    lea    0x201568(%rip),%rbp        # 601e08 <_fini+0x201514>
  4008a0:       53                      push   %rbx
  4008a1:       4c 29 e5                sub    %r12,%rbp
  4008a4:       31 db                   xor    %ebx,%ebx
  4008a6:       48 c1 fd 03             sar    $0x3,%rbp
  4008aa:       48 83 ec 08             sub    $0x8,%rsp
  4008ae:       e8 4d fd ff ff          callq  400600 <_init>
  4008b3:       48 85 ed                test   %rbp,%rbp
  4008b6:       74 1e                   je     4008d6 <__gmon_start__@plt+0x266>
  4008b8:       0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  4008bf:       00
  4008c0:       4c 89 ea                mov    %r13,%rdx
  4008c3:       4c 89 f6                mov    %r14,%rsi
  4008c6:       44 89 ff                mov    %r15d,%edi
  4008c9:       41 ff 14 dc             callq  *(%r12,%rbx,8)
  4008cd:       48 83 c3 01             add    $0x1,%rbx
  4008d1:       48 39 eb                cmp    %rbp,%rbx
  4008d4:       75 ea                   jne    4008c0 <__gmon_start__@plt+0x250>
  4008d6:       48 83 c4 08             add    $0x8,%rsp
  4008da:       5b                      pop    %rbx
  4008db:       5d                      pop    %rbp
  4008dc:       41 5c                   pop    %r12
  4008de:       41 5d                   pop    %r13
  4008e0:       41 5e                   pop    %r14
  4008e2:       41 5f                   pop    %r15
  4008e4:       c3                      retq
  4008e5:       66 66 2e 0f 1f 84 00    data32 nopw %cs:0x0(%rax,%rax,1)
  4008ec:       00 00 00 00
  4008f0:       f3 c3                   repz retq

Disassembly of section .fini:

00000000004008f4 <_fini>:
  4008f4:       48 83 ec 08             sub    $0x8,%rsp
  4008f8:       48 83 c4 08             add    $0x8,%rsp
  4008fc:       c3                      retq
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -d -O2 2
objdump: invalid option -- 'O'
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information

 The following switches are optional:
  -b, --target=BFDNAME           Specify the target object format as BFDNAME
  -m, --architecture=MACHINE     Specify the target architecture as MACHINE
  -j, --section=NAME             Only display information for section NAME
  -M, --disassembler-options=OPT Pass text OPT on to the disassembler
  -EB --endian=big               Assume big endian format when disassembling
  -EL --endian=little            Assume little endian format when disassembling
      --file-start-context       Include context from start of file (with -S)
  -I, --include=DIR              Add DIR to search list for source files
  -l, --line-numbers             Include line numbers and filenames in output
  -F, --file-offsets             Include file offsets when displaying information
  -C, --demangle[=STYLE]         Decode mangled/processed symbol names
                                  The STYLE, if specified, can be `auto', `gnu',
                                  `lucid', `arm', `hp', `edg', `gnu-v3', `java'
                                  or `gnat'
  -w, --wide                     Format output for more than 80 columns
  -z, --disassemble-zeroes       Do not skip blocks of zeroes when disassembling
      --start-address=ADDR       Only process data whose address is >= ADDR
      --stop-address=ADDR        Only process data whose address is <= ADDR
      --prefix-addresses         Print complete address alongside disassembly
      --[no-]show-raw-insn       Display hex alongside symbolic disassembly
      --insn-width=WIDTH         Display WIDTH bytes on a single line for -d
      --adjust-vma=OFFSET        Add OFFSET to all displayed section addresses
      --special-syms             Include special symbols in symbol dumps
      --prefix=PREFIX            Add PREFIX to absolute paths for -S
      --prefix-strip=LEVEL       Strip initial directory names for -S
      --dwarf-depth=N        Do not display DIEs at depth N or greater
      --dwarf-start=N        Display DIEs starting with N, at the same depth
                             or deeper
      --dwarf-check          Make additional dwarf internal consistency checks.

objdump: supported targets: elf64-x86-64 elf32-i386 elf32-x86-64 a.out-i386-linux pei-i386 pei-x86-64 elf64-l1om elf64-k1om elf64-little elf64-big elf32-little elf32-big pe-x86-64 pe-i386 plugin srec symb
olsrec verilog tekhex binary ihex
objdump: supported architectures: i386 i386:x86-64 i386:x64-32 i8086 i386:intel i386:x86-64:intel i386:x64-32:intel i386:nacl i386:x86-64:nacl i386:x64-32:nacl l1om l1om:intel k1om k1om:intel plugin
```

```
The following i386/x86-64 specific disassembler options are supported for use
with the -M switch (multiple options should be separated by commas):
  x86-64      Disassemble in 64bit mode
  i386        Disassemble in 32bit mode
  i8086       Disassemble in 16bit mode
  att         Display instruction in AT&T syntax
  intel       Display instruction in Intel syntax
  att-mnemonic
              Display instruction in AT&T mnemonic
  intel-mnemonic
              Display instruction in Intel mnemonic
  addr64      Assume 64bit address size
  addr32      Assume 32bit address size
  addr16      Assume 16bit address size
  data32      Assume 32bit data size
  data16      Assume 16bit data size
  suffix      Always display instruction suffix in AT&T syntax
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -O2 2
objdump: invalid option -- 'O'
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information

 The following switches are optional:
  -b, --target=BFDNAME           Specify the target object format as BFDNAME
  -m, --architecture=MACHINE     Specify the target architecture as MACHINE
  -j, --section=NAME             Only display information for section NAME
  -M, --disassembler-options=OPT Pass text OPT on to the disassembler
  -EB --endian=big               Assume big endian format when disassembling
  -EL --endian=little            Assume little endian format when disassembling
      --file-start-context       Include context from start of file (with -S)
  -I, --include=DIR              Add DIR to search list for source files
  -l, --line-numbers             Include line numbers and filenames in output
  -F, --file-offsets             Include file offsets when displaying information
  -C, --demangle[=STYLE]         Decode mangled/processed symbol names
                                  The STYLE, if specified, can be `auto', `gnu',
                                  `lucid', `arm', `hp', `edg', `gnu-v3', `java'
                                  or `gnat'
  -w, --wide                     Format output for more than 80 columns
  -z, --disassemble-zeroes       Do not skip blocks of zeroes when disassembling
      --start-address=ADDR       Only process data whose address is >= ADDR
      --stop-address=ADDR        Only process data whose address is <= ADDR
      --prefix-addresses         Print complete address alongside disassembly
      --[no-]show-raw-insn       Display hex alongside symbolic disassembly
      --insn-width=WIDTH         Display WIDTH bytes on a single line for -d
      --adjust-vma=OFFSET        Add OFFSET to all displayed section addresses
      --special-syms             Include special symbols in symbol dumps
      --prefix=PREFIX            Add PREFIX to absolute paths for -S
      --prefix-strip=LEVEL       Strip initial directory names for -S
      --dwarf-depth=N        Do not display DIEs at depth N or greater
      --dwarf-start=N        Display DIEs starting with N, at the same depth
                             or deeper
      --dwarf-check          Make additional dwarf internal consistency checks.

objdump: supported targets: elf64-x86-64 elf32-i386 elf32-x86-64 a.out-i386-linux pei-i386 pei-x86-64 elf64-l1om elf64-k1om elf64-little elf64-big elf32-little elf32-big pe-x86-64 pe-i386 plugin srec symb
olsrec verilog tekhex binary ihex
objdump: supported architectures: i386 i386:x86-64 i386:x64-32 i8086 i386:intel i386:x86-64:intel i386:x64-32:intel i386:nacl i386:x86-64:nacl i386:x64-32:nacl l1om l1om:intel k1om k1om:intel plugin

The following i386/x86-64 specific disassembler options are supported for use
with the -M switch (multiple options should be separated by commas):
  x86-64      Disassemble in 64bit mode
  i386        Disassemble in 32bit mode
  i8086       Disassemble in 16bit mode
  att         Display instruction in AT&T syntax
  intel       Display instruction in Intel syntax
  att-mnemonic
              Display instruction in AT&T mnemonic
  intel-mnemonic
              Display instruction in Intel mnemonic
  addr64      Assume 64bit address size
  addr32      Assume 32bit address size
  addr16      Assume 16bit address size
  data32      Assume 32bit data size
  data16      Assume 16bit data size
  suffix      Always display instruction suffix in AT&T syntax
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -O2 2
objdump: invalid option -- 'O'
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information
```

```
 The following switches are optional:
  -b, --target=BFDNAME           Specify the target object format as BFDNAME
  -m, --architecture=MACHINE     Specify the target architecture as MACHINE
  -j, --section=NAME             Only display information for section NAME
  -M, --disassembler-options=OPT Pass text OPT on to the disassembler
  -EB --endian=big               Assume big endian format when disassembling
  -EL --endian=little            Assume little endian format when disassembling
      --file-start-context       Include context from start of file (with -S)
  -I, --include=DIR              Add DIR to search list for source files
  -l, --line-numbers             Include line numbers and filenames in output
  -F, --file-offsets             Include file offsets when displaying information
  -C, --demangle[=STYLE]         Decode mangled/processed symbol names
                                  The STYLE, if specified, can be `auto', `gnu',
                                  `lucid', `arm', `hp', `edg', `gnu-v3', `java'
                                  or `gnat'
  -w, --wide                     Format output for more than 80 columns
  -z, --disassemble-zeroes       Do not skip blocks of zeroes when disassembling
      --start-address=ADDR       Only process data whose address is >= ADDR
      --stop-address=ADDR        Only process data whose address is <= ADDR
      --prefix-addresses         Print complete address alongside disassembly
      --[no-]show-raw-insn       Display hex alongside symbolic disassembly
      --insn-width=WIDTH         Display WIDTH bytes on a single line for -d
      --adjust-vma=OFFSET        Add OFFSET to all displayed section addresses
      --special-syms             Include special symbols in symbol dumps
      --prefix=PREFIX            Add PREFIX to absolute paths for -S
      --prefix-strip=LEVEL       Strip initial directory names for -S
      --dwarf-depth=N        Do not display DIEs at depth N or greater
      --dwarf-start=N        Display DIEs starting with N, at the same depth
                             or deeper
      --dwarf-check          Make additional dwarf internal consistency checks.

objdump: supported targets: elf64-x86-64 elf32-i386 elf32-x86-64 a.out-i386-linux pei-i386 pei-x86-64 elf64-l1om elf64-k1om elf64-little elf64-big elf32-little elf32-big pe-x86-64 pe-i386 plugin srec symb
olsrec verilog tekhex binary ihex
objdump: supported architectures: i386 i386:x86-64 i386:x64-32 i8086 i386:intel i386:x86-64:intel i386:x64-32:intel i386:nacl i386:x86-64:nacl i386:x64-32:nacl l1om l1om:intel k1om k1om:intel plugin

The following i386/x86-64 specific disassembler options are supported for use
with the -M switch (multiple options should be separated by commas):
  x86-64      Disassemble in 64bit mode
  i386        Disassemble in 32bit mode
  i8086       Disassemble in 16bit mode
  att         Display instruction in AT&T syntax
  intel       Display instruction in Intel syntax
  att-mnemonic
              Display instruction in AT&T mnemonic
  intel-mnemonic
              Display instruction in Intel mnemonic
  addr64      Assume 64bit address size
  addr32      Assume 32bit address size
  addr16      Assume 16bit address size
  data32      Assume 32bit data size
  data16      Assume 16bit data size
  suffix      Always display instruction suffix in AT&T syntax
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ gdb -d 2
GNU gdb (Ubuntu 7.7.1-0ubuntu5~14.04.2) 7.7.1
Copyright (C) 2014 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:
<http://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".

warning: /home/Dpate85/dpate85/hw2/puzzles/2 is not a directory.
(gdb) Quit
(gdb) quit
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ clear

Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -d 2

2:     file format elf64-x86-64


Disassembly of section .init:

0000000000400600 <_init>:
  400600:       48 83 ec 08             sub    $0x8,%rsp
  400604:       48 8b 05 ed 19 20 00    mov    0x2019ed(%rip),%rax        # 601ff8 <_fini+0x201704>
  40060b:       48 85 c0                test   %rax,%rax
  40060e:       74 05                   je     400615 <_init+0x15>
  400610:       e8 5b 00 00 00          callq  400670 <__gmon_start__@plt>
  400615:       48 83 c4 08             add    $0x8,%rsp
  400619:       c3                      retq

Disassembly of section .plt:

0000000000400620 <curl_easy_init@plt-0x10>:
  400620:       ff 35 e2 19 20 00       pushq  0x2019e2(%rip)        # 602008 <_fini+0x201714>
  400626:       ff 25 e4 19 20 00       jmpq   *0x2019e4(%rip)        # 602010 <_fini+0x20171c>
  40062c:       0f 1f 40 00             nopl   0x0(%rax)

0000000000400630 <curl_easy_init@plt>:
  400630:       ff 25 e2 19 20 00       jmpq   *0x2019e2(%rip)        # 602018 <_fini+0x201724>
  400636:       68 00 00 00 00          pushq  $0x0
  40063b:       e9 e0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400640 <curl_easy_perform@plt>:
  400640:       ff 25 da 19 20 00       jmpq   *0x2019da(%rip)        # 602020 <_fini+0x20172c>
  400646:       68 01 00 00 00          pushq  $0x1
  40064b:       e9 d0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400650 <curl_easy_setopt@plt>:
  400650:       ff 25 d2 19 20 00       jmpq   *0x2019d2(%rip)        # 602028 <_fini+0x201734>
  400656:       68 02 00 00 00          pushq  $0x2
  40065b:       e9 c0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400660 <__libc_start_main@plt>:
  400660:       ff 25 ca 19 20 00       jmpq   *0x2019ca(%rip)        # 602030 <_fini+0x20173c>
  400666:       68 03 00 00 00          pushq  $0x3
  40066b:       e9 b0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400670 <__gmon_start__@plt>:
  400670:       ff 25 c2 19 20 00       jmpq   *0x2019c2(%rip)        # 602038 <_fini+0x201744>
  400676:       68 04 00 00 00          pushq  $0x4
  40067b:       e9 a0 ff ff ff          jmpq   400620 <_init+0x20>

Disassembly of section .text:

0000000000400680 <.text>:
  400680:       31 ed                   xor    %ebp,%ebp
  400682:       49 89 d1                mov    %rdx,%r9
  400685:       5e                      pop    %rsi
  400686:       48 89 e2                mov    %rsp,%rdx
  400689:       48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40068d:       50                      push   %rax
  40068e:       54                      push   %rsp
  40068f:       49 c7 c0 f0 08 40 00    mov    $0x4008f0,%r8
  400696:       48 c7 c1 80 08 40 00    mov    $0x400880,%rcx
```

```
40069d:       48 c7 c7 fd 07 40 00    mov    $0x4007fd,%rdi
4006a4:       e8 b7 ff ff ff          callq  400660 <__libc_start_main@plt>
4006a9:       f4                      hlt
4006aa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
4006b0:       b8 57 20 60 00          mov    $0x602057,%eax
4006b5:       55                      push   %rbp
4006b6:       48 2d 50 20 60 00       sub    $0x602050,%rax
4006bc:       48 83 f8 0e             cmp    $0xe,%rax
4006c0:       48 89 e5                mov    %rsp,%rbp
4006c3:       77 02                   ja     4006c7 <__gmon_start__@plt+0x57>
4006c5:       5d                      pop    %rbp
4006c6:       c3                      retq
4006c7:       b8 00 00 00 00          mov    $0x0,%eax
4006cc:       48 85 c0                test   %rax,%rax
4006cf:       74 f4                   je     4006c5 <__gmon_start__@plt+0x55>
4006d1:       5d                      pop    %rbp
4006d2:       bf 50 20 60 00          mov    $0x602050,%edi
4006d7:       ff e0                   jmpq   *%rax
4006d9:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
4006e0:       b8 50 20 60 00          mov    $0x602050,%eax
4006e5:       55                      push   %rbp
4006e6:       48 2d 50 20 60 00       sub    $0x602050,%rax
4006ec:       48 c1 f8 03             sar    $0x3,%rax
4006f0:       48 89 e5                mov    %rsp,%rbp
4006f3:       48 89 c2                mov    %rax,%rdx
4006f6:       48 c1 ea 3f             shr    $0x3f,%rdx
4006fa:       48 01 d0                add    %rdx,%rax
4006fd:       48 d1 f8                sar    %rax
400700:       75 02                   jne    400704 <__gmon_start__@plt+0x94>
400702:       5d                      pop    %rbp
400703:       c3                      retq
400704:       ba 00 00 00 00          mov    $0x0,%edx
400709:       48 85 d2                test   %rdx,%rdx
40070c:       74 f4                   je     400702 <__gmon_start__@plt+0x92>
40070e:       5d                      pop    %rbp
40070f:       48 89 c6                mov    %rax,%rsi
400712:       bf 50 20 60 00          mov    $0x602050,%edi
400717:       ff e2                   jmpq   *%rdx
400719:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
400720:       80 3d 29 19 20 00 00    cmpb   $0x0,0x201929(%rip)        # 602050 <_edata>
400727:       75 11                   jne    40073a <__gmon_start__@plt+0xca>
400729:       55                      push   %rbp
40072a:       48 89 e5                mov    %rsp,%rbp
40072d:       e8 7e ff ff ff          callq  4006b0 <__gmon_start__@plt+0x40>
400732:       5d                      pop    %rbp
400733:       c6 05 16 19 20 00 01    movb   $0x1,0x201916(%rip)        # 602050 <_edata>
40073a:       f3 c3                   repz retq
40073c:       0f 1f 40 00             nopl   0x0(%rax)
400740:       48 83 3d c8 16 20 00    cmpq   $0x0,0x2016c8(%rip)        # 601e10 <_fini+0x20151c>
400747:       00
400748:       74 1e                   je     400768 <__gmon_start__@plt+0xf8>
40074a:       b8 00 00 00 00          mov    $0x0,%eax
40074f:       48 85 c0                test   %rax,%rax
400752:       74 14                   je     400768 <__gmon_start__@plt+0xf8>
400754:       55                      push   %rbp
400755:       bf 10 1e 60 00          mov    $0x601e10,%edi
40075a:       48 89 e5                mov    %rsp,%rbp
40075d:       ff d0                   callq  *%rax
40075f:       5d                      pop    %rbp
400760:       e9 7b ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
400765:       0f 1f 00                nopl   (%rax)
400768:       e9 73 ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
40076d:       55                      push   %rbp
40076e:       48 89 e5                mov    %rsp,%rbp
400771:       5d                      pop    %rbp
400772:       c3                      retq
400773:       55                      push   %rbp
400774:       48 89 e5                mov    %rsp,%rbp
400777:       5d                      pop    %rbp
400778:       c3                      retq
400779:       55                      push   %rbp
40077a:       48 89 e5                mov    %rsp,%rbp
40077d:       5d                      pop    %rbp
40077e:       c3                      retq
40077f:       55                      push   %rbp
400780:       48 89 e5                mov    %rsp,%rbp
400783:       5d                      pop    %rbp
400784:       c3                      retq
400785:       55                      push   %rbp
400786:       48 89 e5                mov    %rsp,%rbp
400789:       5d                      pop    %rbp
40078a:       c3                      retq
40078b:       55                      push   %rbp
40078c:       48 89 e5                mov    %rsp,%rbp
40078f:       5d                      pop    %rbp
400790:       c3                      retq
400791:       55                      push   %rbp
400792:       48 89 e5                mov    %rsp,%rbp
400795:       5d                      pop    %rbp
400796:       c3                      retq
400797:       55                      push   %rbp
400798:       48 89 e5                mov    %rsp,%rbp
40079b:       5d                      pop    %rbp
40079c:       c3                      retq
40079d:       55                      push   %rbp
40079e:       48 89 e5                mov    %rsp,%rbp
4007a1:       5d                      pop    %rbp
4007a2:       c3                      retq
4007a3:       55                      push   %rbp
4007a4:       48 89 e5                mov    %rsp,%rbp
4007a7:       5d                      pop    %rbp
4007a8:       c3                      retq
4007a9:       55                      push   %rbp
4007aa:       48 89 e5                mov    %rsp,%rbp
4007ad:       5d                      pop    %rbp
4007ae:       c3                      retq
4007af:       55                      push   %rbp
4007b0:       48 89 e5                mov    %rsp,%rbp
4007b3:       5d                      pop    %rbp
4007b4:       c3                      retq
4007b5:       55                      push   %rbp
4007b6:       48 89 e5                mov    %rsp,%rbp
4007b9:       5d                      pop    %rbp
4007ba:       c3                      retq
4007bb:       55                      push   %rbp
4007bc:       48 89 e5                mov    %rsp,%rbp
4007bf:       5d                      pop    %rbp
4007c0:       c3                      retq
4007c1:       55                      push   %rbp
4007c2:       48 89 e5                mov    %rsp,%rbp
4007c5:       5d                      pop    %rbp
4007c6:       c3                      retq
4007c7:       55                      push   %rbp
4007c8:       48 89 e5                mov    %rsp,%rbp
4007cb:       5d                      pop    %rbp
4007cc:       c3                      retq
4007cd:       55                      push   %rbp
4007ce:       48 89 e5                mov    %rsp,%rbp
4007d1:       5d                      pop    %rbp
4007d2:       c3                      retq
4007d3:       55                      push   %rbp
4007d4:       48 89 e5                mov    %rsp,%rbp
4007d7:       5d                      pop    %rbp
4007d8:       c3                      retq
4007d9:       55                      push   %rbp
```

```
  4007da:       48 89 e5                mov    %rsp,%rbp
  4007dd:       5d                      pop    %rbp
  4007de:       c3                      retq
  4007df:       55                      push   %rbp
  4007e0:       48 89 e5                mov    %rsp,%rbp
  4007e3:       5d                      pop    %rbp
  4007e4:       c3                      retq
  4007e5:       55                      push   %rbp
  4007e6:       48 89 e5                mov    %rsp,%rbp
  4007e9:       5d                      pop    %rbp
  4007ea:       c3                      retq
  4007eb:       55                      push   %rbp
  4007ec:       48 89 e5                mov    %rsp,%rbp
  4007ef:       5d                      pop    %rbp
  4007f0:       c3                      retq
  4007f1:       55                      push   %rbp
  4007f2:       48 89 e5                mov    %rsp,%rbp
  4007f5:       5d                      pop    %rbp
  4007f6:       c3                      retq
  4007f7:       55                      push   %rbp
  4007f8:       48 89 e5                mov    %rsp,%rbp
  4007fb:       5d                      pop    %rbp
  4007fc:       c3                      retq
  4007fd:       55                      push   %rbp
  4007fe:       48 89 e5                mov    %rsp,%rbp
  400801:       48 83 ec 20             sub    $0x20,%rsp
  400805:       e8 26 fe ff ff          callq  400630 <curl_easy_init@plt>
  40080a:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  40080e:       48 83 7d f8 00          cmpq   $0x0,-0x8(%rbp)
  400813:       74 5a                   je     40086f <__gmon_start__@plt+0x1ff>
  400815:       c7 45 ec 12 27 00 00    movl   $0x2712,-0x14(%rbp)
  40081c:       8b 4d ec                mov    -0x14(%rbp),%ecx
  40081f:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400823:       ba 08 09 40 00          mov    $0x400908,%edx
  400828:       89 ce                   mov    %ecx,%esi
  40082a:       48 89 c7                mov    %rax,%rdi
  40082d:       b8 00 00 00 00          mov    $0x0,%eax
  400832:       e8 19 fe ff ff          callq  400650 <curl_easy_setopt@plt>
  400837:       c7 45 f0 34 00 00 00    movl   $0x34,-0x10(%rbp)
  40083e:       8b 4d f0                mov    -0x10(%rbp),%ecx
  400841:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400845:       ba 01 00 00 00          mov    $0x1,%edx
  40084a:       89 ce                   mov    %ecx,%esi
  40084c:       48 89 c7                mov    %rax,%rdi
  40084f:       b8 00 00 00 00          mov    $0x0,%eax
  400854:       e8 f7 fd ff ff          callq  400650 <curl_easy_setopt@plt>
  400859:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  40085d:       48 89 c7                mov    %rax,%rdi
  400860:       e8 db fd ff ff          callq  400640 <curl_easy_perform@plt>
  400865:       89 45 f4                mov    %eax,-0xc(%rbp)
  400868:       b8 00 00 00 00          mov    $0x0,%eax
  40086d:       eb 05                   jmp    400874 <__gmon_start__@plt+0x204>
  40086f:       b8 01 00 00 00          mov    $0x1,%eax
  400874:       c9                      leaveq
  400875:       c3                      retq
  400876:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  40087d:       00 00 00
  400880:       41 57                   push   %r15
  400882:       41 89 ff                mov    %edi,%r15d
  400885:       41 56                   push   %r14
  400887:       49 89 f6                mov    %rsi,%r14
  40088a:       41 55                   push   %r13
  40088c:       49 89 d5                mov    %rdx,%r13
  40088f:       41 54                   push   %r12
  400891:       4c 8d 25 68 15 20 00    lea    0x201568(%rip),%r12        # 601e00 <_fini+0x20150c>
  400898:       55                      push   %rbp
  400899:       48 8d 2d 68 15 20 00    lea    0x201568(%rip),%rbp        # 601e08 <_fini+0x201514>
  4008a0:       53                      push   %rbx
  4008a1:       4c 29 e5                sub    %r12,%rbp
  4008a4:       31 db                   xor    %ebx,%ebx
  4008a6:       48 c1 fd 03             sar    $0x3,%rbp
  4008aa:       48 83 ec 08             sub    $0x8,%rsp
  4008ae:       e8 4d fd ff ff          callq  400600 <_init>
  4008b3:       48 85 ed                test   %rbp,%rbp
  4008b6:       74 1e                   je     4008d6 <__gmon_start__@plt+0x266>
  4008b8:       0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  4008bf:       00
  4008c0:       4c 89 ea                mov    %r13,%rdx
  4008c3:       4c 89 f6                mov    %r14,%rsi
  4008c6:       44 89 ff                mov    %r15d,%edi
  4008c9:       41 ff 14 dc             callq  *(%r12,%rbx,8)
  4008cd:       48 83 c3 01             add    $0x1,%rbx
  4008d1:       48 39 eb                cmp    %rbp,%rbx
  4008d4:       75 ea                   jne    4008c0 <__gmon_start__@plt+0x250>
  4008d6:       48 83 c4 08             add    $0x8,%rsp
  4008da:       5b                      pop    %rbx
  4008db:       5d                      pop    %rbp
  4008dc:       41 5c                   pop    %r12
  4008de:       41 5d                   pop    %r13
  4008e0:       41 5e                   pop    %r14
  4008e2:       41 5f                   pop    %r15
  4008e4:       c3                      retq
  4008e5:       66 66 2e 0f 1f 84 00    data32 nopw %cs:0x0(%rax,%rax,1)
  4008ec:       00 00 00 00
  4008f0:       f3 c3                   repz retq

Disassembly of section .fini:

00000000004008f4 <_fini>:
  4008f4:       48 83 ec 08             sub    $0x8,%rsp
  4008f8:       48 83 c4 08             add    $0x8,%rsp
  4008fc:       c3                      retq
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./2
The password is "yes" without quotes if the call to curl_easy_perform was successful; the password is "no" without quotes if the call to curl_easy_perform as unsuccessful.
You may not use gdb to answer this question.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf -s 2

Symbol table '.dynsym' contains 14 entries:
   Num:    Value          Size Type    Bind   Vis      Ndx Name
     0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
     1: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_init@CURL_GNUTLS_3 (2)
     2: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_perform@CURL_GNUTLS_3 (2)
     3: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _Jv_RegisterClasses
     4: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_setopt@CURL_GNUTLS_3 (2)
     5: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _ITM_deregisterTMCloneTab
     6: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.2.5 (3)
     7: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND __gmon_start__
     8: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _ITM_registerTMCloneTable
     9: 0000000000602050     0 NOTYPE  GLOBAL DEFAULT   24 _edata
    10: 0000000000602058     0 NOTYPE  GLOBAL DEFAULT   25 _end
    11: 0000000000400600     0 FUNC    GLOBAL DEFAULT   11 _init
    12: 0000000000602050     0 NOTYPE  GLOBAL DEFAULT   25 __bss_start
    13: 00000000004008f4     0 FUNC    GLOBAL DEFAULT   14 _fini
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf 2
Usage: readelf <option(s)> elf-file(s)
 Display information about the contents of ELF format files
 Options are:
  -a --all               Equivalent to: -h -l -S -s -r -d -V -A -I
  -h --file-header       Display the ELF file header
  -l --program-headers   Display the program headers
     --segments          An alias for --program-headers
  -S --section-headers   Display the sections' header
     --sections          An alias for --section-headers
  -g --section-groups    Display the section groups
  -t --section-details   Display the section details
```

```
  -e --headers           Equivalent to: -h -l -S
  -s --syms              Display the symbol table
     --symbols           An alias for --syms
  --dyn-syms             Display the dynamic symbol table
  -n --notes             Display the core notes (if present)
  -r --relocs            Display the relocations (if present)
  -u --unwind            Display the unwind info (if present)
  -d --dynamic           Display the dynamic section (if present)
  -V --version-info      Display the version sections (if present)
  -A --arch-specific     Display architecture specific information (if any)
  -c --archive-index     Display the symbol/file index in an archive
  -D --use-dynamic       Use the dynamic section info when displaying symbols
  -x --hex-dump=<number|name>
                         Dump the contents of section <number|name> as bytes
  -p --string-dump=<number|name>
                         Dump the contents of section <number|name> as strings
  -R --relocated-dump=<number|name>
                         Dump the contents of section <number|name> as relocated bytes
  -w[lLiaprmfFsoRt] or
  --debug-dump[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
               =frames-interp,=str,=loc,=Ranges,=pubtypes,
               =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
               =addr,=cu_index]
                         Display the contents of DWARF2 debug sections
  --dwarf-depth=N        Do not display DIEs at depth N or greater
  --dwarf-start=N        Display DIEs starting with N, at the same depth
                         or deeper
  -I --histogram         Display histogram of bucket list lengths
  -W --wide              Allow output width to exceed 80 characters
  @<file>                Read options from <file>
  -H --help              Display this information
  -v --version           Display the version number of readelf
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf -a 2
ELF Header:
  Magic:   7f 45 4c 46 02 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF64
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Advanced Micro Devices X86-64
  Version:                           0x1
  Entry point address:               0x400680
  Start of program headers:          64 (bytes into file)
  Start of section headers:          8600 (bytes into file)
  Flags:                             0x0
  Size of this header:               64 (bytes)
  Size of program headers:           56 (bytes)
  Number of program headers:         9
  Size of section headers:           64 (bytes)
  Number of section headers:         28
  Section header string table index: 27

Section Headers:
  [Nr] Name              Type             Address           Offset
       Size              EntSize          Flags  Link  Info  Align
  [ 0]                   NULL             0000000000000000  00000000
       0000000000000000  0000000000000000           0     0     0
  [ 1] .interp           PROGBITS         0000000000400238  00000238
       000000000000001c  0000000000000000   A       0     0     1
  [ 2] .note.ABI-tag     NOTE             0000000000400254  00000254
       0000000000000020  0000000000000000   A       0     0     4
  [ 3] .note.gnu.build-i NOTE             0000000000400274  00000274
       0000000000000024  0000000000000000   A       0     0     4
  [ 4] .gnu.hash         GNU_HASH         0000000000400298  00000298
       0000000000000038  0000000000000000   A       5     0     8
  [ 5] .dynsym           DYNSYM           00000000004002d0  000002d0
       0000000000000150  0000000000000018   A       6     1     8
  [ 6] .dynstr           STRTAB           0000000000400420  00000420
       00000000000000f4  0000000000000000   A       0     0     1
  [ 7] .gnu.version      VERSYM           0000000000400514  00000514
       000000000000001c  0000000000000002   A       5     0     2
  [ 8] .gnu.version_r    VERNEED          0000000000400530  00000530
       0000000000000040  0000000000000000   A       6     2     8
  [ 9] .rela.dyn         RELA             0000000000400570  00000570
       0000000000000018  0000000000000018   A       5     0     8
  [10] .rela.plt         RELA             0000000000400588  00000588
       0000000000000078  0000000000000018   A       5    12     8
  [11] .init             PROGBITS         0000000000400600  00000600
       000000000000001a  0000000000000000  AX       0     0     4
  [12] .plt              PROGBITS         0000000000400620  00000620
       0000000000000060  0000000000000010  AX       0     0    16
  [13] .text             PROGBITS         0000000000400680  00000680
       0000000000000272  0000000000000000  AX       0     0    16
  [14] .fini             PROGBITS         00000000004008f4  000008f4
       0000000000000009  0000000000000000  AX       0     0     4
  [15] .rodata           PROGBITS         0000000000400900  00000900
       0000000000000031  0000000000000000   A       0     0     8
  [16] .eh_frame_hdr     PROGBITS         0000000000400934  00000934
       00000000000000f4  0000000000000000   A       0     0     4
  [17] .eh_frame         PROGBITS         0000000000400a28  00000a28
       00000000000003f4  0000000000000000   A       0     0     8
  [18] .init_array       INIT_ARRAY       0000000000601e00  00001e00
       0000000000000008  0000000000000000  WA       0     0     8
  [19] .fini_array       FINI_ARRAY       0000000000601e08  00001e08
       0000000000000008  0000000000000000  WA       0     0     8
  [20] .jcr              PROGBITS         0000000000601e10  00001e10
       0000000000000008  0000000000000000  WA       0     0     8
  [21] .dynamic          DYNAMIC          0000000000601e18  00001e18
       00000000000001e0  0000000000000010  WA       6     0     8
  [22] .got              PROGBITS         0000000000601ff8  00001ff8
       0000000000000008  0000000000000008  WA       0     0     8
  [23] .got.plt          PROGBITS         0000000000602000  00002000
       0000000000000040  0000000000000008  WA       0     0     8
  [24] .data             PROGBITS         0000000000602040  00002040
       0000000000000010  0000000000000000  WA       0     0     8
  [25] .bss              NOBITS           0000000000602050  00002050
       0000000000000008  0000000000000000  WA       0     0     1
  [26] .comment          PROGBITS         0000000000000000  00002050
       000000000000004d  0000000000000001  MS       0     0     1
  [27] .shstrtab         STRTAB           0000000000000000  0000209d
       00000000000000f8  0000000000000000           0     0     1
Key to Flags:
  W (write), A (alloc), X (execute), M (merge), S (strings), l (large)
  I (info), L (link order), G (group), T (TLS), E (exclude), x (unknown)
  O (extra OS processing required) o (OS specific), p (processor specific)

There are no section groups in this file.

Program Headers:
  Type           Offset             VirtAddr           PhysAddr
                 FileSiz            MemSiz              Flags  Align
  PHDR           0x0000000000000040 0x0000000000400040 0x0000000000400040
                 0x00000000000001f8 0x00000000000001f8  R E    8
  INTERP         0x0000000000000238 0x0000000000400238 0x0000000000400238
                 0x000000000000001c 0x000000000000001c  R      1
      [Requesting program interpreter: /lib64/ld-linux-x86-64.so.2]
  LOAD           0x0000000000000000 0x0000000000400000 0x0000000000400000
                 0x00000000000000e1c 0x00000000000000e1c  R E    200000
  LOAD           0x0000000000001e00 0x0000000000601e00 0x0000000000601e00
                 0x0000000000000250 0x0000000000000258  RW     200000
  DYNAMIC        0x0000000000001e18 0x0000000000601e18 0x0000000000601e18
                 0x00000000000001e0 0x00000000000001e0  RW     8
  NOTE           0x0000000000000254 0x0000000000400254 0x0000000000400254
```

```
                0x0000000000000044 0x0000000000000044  R      4
  GNU_EH_FRAME   0x0000000000000934 0x0000000000400934 0x0000000000400934
                0x00000000000000f4 0x00000000000000f4  R      4
  GNU_STACK      0x0000000000000000 0x0000000000000000 0x0000000000000000
                0x0000000000000000 0x0000000000000000  RW     10
  GNU_RELRO      0x0000000000001e00 0x0000000000601e00 0x0000000000601e00
                0x0000000000000200 0x0000000000000200  R      1

 Section to Segment mapping:
  Segment Sections...
   00
   01     .interp
   02     .interp .note.ABI-tag .note.gnu.build-id .gnu.hash .dynsym .dynstr .gnu.version .gnu.version_r .rela.dyn .rela.plt .init .plt .text .fini .rodata .eh_frame_hdr .eh_frame
   03     .init_array .fini_array .jcr .dynamic .got .got.plt .data .bss
   04     .dynamic
   05     .note.ABI-tag .note.gnu.build-id
   06     .eh_frame_hdr
   07
   08     .init_array .fini_array .jcr .dynamic .got

Dynamic section at offset 0x1e18 contains 25 entries:
  Tag        Type                         Name/Value
 0x0000000000000001 (NEEDED)             Shared library: [libcurl-gnutls.so.4]
 0x0000000000000001 (NEEDED)             Shared library: [libc.so.6]
 0x000000000000000c (INIT)               0x400600
 0x000000000000000d (FINI)               0x4008f4
 0x0000000000000019 (INIT_ARRAY)         0x601e00
 0x000000000000001b (INIT_ARRAYSZ)       8 (bytes)
 0x000000000000001a (FINI_ARRAY)         0x601e08
 0x000000000000001c (FINI_ARRAYSZ)       8 (bytes)
 0x000000006ffffef5 (GNU_HASH)           0x400298
 0x0000000000000005 (STRTAB)             0x400420
 0x0000000000000006 (SYMTAB)             0x4002d0
 0x000000000000000a (STRSZ)              244 (bytes)
 0x000000000000000b (SYMENT)             24 (bytes)
 0x0000000000000015 (DEBUG)              0x0
 0x0000000000000003 (PLTGOT)             0x602000
 0x0000000000000002 (PLTRELSZ)           120 (bytes)
 0x0000000000000014 (PLTREL)             RELA
 0x0000000000000017 (JMPREL)             0x400588
 0x0000000000000007 (RELA)               0x400570
 0x0000000000000008 (RELASZ)             24 (bytes)
 0x0000000000000009 (RELAENT)            24 (bytes)
 0x000000006ffffffe (VERNEED)            0x400530
 0x000000006fffffff (VERNEEDNUM)         2
 0x000000006ffffff0 (VERSYM)             0x400514
 0x0000000000000000 (NULL)               0x0

Relocation section '.rela.dyn' at offset 0x570 contains 1 entries:
  Offset          Info           Type           Sym. Value    Sym. Name + Addend
000000601ff8  000700000006 R_X86_64_GLOB_DAT 0000000000000000 __gmon_start__ + 0

Relocation section '.rela.plt' at offset 0x588 contains 5 entries:
  Offset          Info           Type           Sym. Value    Sym. Name + Addend
000000602018  000100000007 R_X86_64_JUMP_SLO 0000000000000000 curl_easy_init + 0
000000602020  000200000007 R_X86_64_JUMP_SLO 0000000000000000 curl_easy_perform + 0
000000602028  000400000007 R_X86_64_JUMP_SLO 0000000000000000 curl_easy_setopt + 0
000000602030  000600000007 R_X86_64_JUMP_SLO 0000000000000000 __libc_start_main + 0
000000602038  000700000007 R_X86_64_JUMP_SLO 0000000000000000 __gmon_start__ + 0

The decoding of unwind sections for machine type Advanced Micro Devices X86-64 is not currently supported.

Symbol table '.dynsym' contains 14 entries:
   Num:    Value          Size Type    Bind   Vis      Ndx Name
     0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
     1: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_init@CURL_GNUTLS_3 (2)
     2: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_perform@CURL_GNUTLS_3 (2)
     3: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _Jv_RegisterClasses
     4: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND curl_easy_setopt@CURL_GNUTLS_3 (2)
     5: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _ITM_deregisterTMCloneTab
     6: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.2.5 (3)
     7: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND __gmon_start__
     8: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND _ITM_registerTMCloneTable
     9: 0000000000602050     0 NOTYPE  GLOBAL DEFAULT   24 _edata
    10: 0000000000602058     0 NOTYPE  GLOBAL DEFAULT   25 _end
    11: 0000000000400600     0 FUNC    GLOBAL DEFAULT   11 _init
    12: 0000000000602050     0 NOTYPE  GLOBAL DEFAULT   25 __bss_start
    13: 00000000004008f4     0 FUNC    GLOBAL DEFAULT   14 _fini

Histogram for `.gnu.hash' bucket list length (total of 3 buckets):
 Length  Number     % of total  Coverage
      0  0          (  0.0%)
      1  1          ( 33.3%)     20.0%
      2  2          ( 66.7%)    100.0%

Version symbols section '.gnu.version' contains 14 entries:
 Addr: 0000000000400514  Offset: 0x000514  Link: 5 (.dynsym)
  000:   0 (*local*)       2 (CURL_GNUTLS_3)   2 (CURL_GNUTLS_3)   0 (*local*)
  004:   2 (CURL_GNUTLS_3)   0 (*local*)       3 (GLIBC_2.2.5)   0 (*local*)
  008:   0 (*local*)       1 (*global*)      1 (*global*)      1 (*global*)
  00c:   1 (*global*)      1 (*global*)

Version needs section '.gnu.version_r' contains 2 entries:
 Addr: 0x0000000000400530  Offset: 0x000530  Link: 6 (.dynstr)
  000000: Version: 1  File: libc.so.6  Cnt: 1
  0x0010:   Name: GLIBC_2.2.5  Flags: none  Version: 3
  0x0020: Version: 1  File: libcurl-gnutls.so.4  Cnt: 1
  0x0030:   Name: CURL_GNUTLS_3  Flags: none  Version: 2

Displaying notes found at file offset 0x00000254 with length 0x00000020:
  Owner                 Data size       Description
  GNU                  0x00000010       NT_GNU_ABI_TAG (ABI version tag)
    OS: Linux, ABI: 2.6.24

Displaying notes found at file offset 0x00000274 with length 0x00000024:
  Owner                 Data size       Description
  GNU                  0x00000014       NT_GNU_BUILD_ID (unique build ID bitstring)
    Build ID: 18ada6ba56c33deac975256efe3955a54be4a407
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump 2
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
```

```
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -g 2

2:     file format elf64-x86-64

Contents of the .eh_frame section:

00000000 0000000000000014 00000000 CIE
  Version:               1
  Augmentation:          "zR"
  Code alignment factor: 1
  Data alignment factor: -8
  Return address column: 16
  Augmentation data:     1b

  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_offset: r16 (rip) at cfa-8
  DW_CFA_undefined: r16 (rip)

00000018 0000000000000014 0000001c FDE cie=00000000 pc=ffffffffffffc58..ffffffffffffc82
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000030 0000000000000014 00000000 CIE
  Version:               1
  Augmentation:          "zR"
  Code alignment factor: 1
  Data alignment factor: -8
  Return address column: 16
  Augmentation data:     1b

  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_offset: r16 (rip) at cfa-8
  DW_CFA_nop
  DW_CFA_nop

00000048 0000000000000024 0000001c FDE cie=00000030 pc=ffffffffffffbf8..ffffffffffffc58
  DW_CFA_def_cfa_offset: 16
  DW_CFA_advance_loc: 6 to ffffffffffffbfe
  DW_CFA_def_cfa_offset: 24
  DW_CFA_advance_loc: 10 to ffffffffffffc08
  DW_CFA_def_cfa_expression (DW_OP_breg7 (rsp): 8; DW_OP_breg16 (rip): 0; DW_OP_lit15; DW_OP_and; DW_OP_lit11; DW_OP_ge; DW_OP_lit3; DW_OP_shl; DW_OP_plus)
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000070 000000000000001c 00000044 FDE cie=00000030 pc=ffffffffffffd45..ffffffffffffd4b
  DW_CFA_advance_loc: 1 to ffffffffffffd46
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd49
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd4a
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000090 000000000000001c 00000064 FDE cie=00000030 pc=ffffffffffffd4b..ffffffffffffd51
  DW_CFA_advance_loc: 1 to ffffffffffffd4c
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd4f
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd50
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000000b0 000000000000001c 00000084 FDE cie=00000030 pc=ffffffffffffd51..ffffffffffffd57
  DW_CFA_advance_loc: 1 to ffffffffffffd52
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd55
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd56
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000000d0 000000000000001c 000000a4 FDE cie=00000030 pc=ffffffffffffd57..ffffffffffffd5d
  DW_CFA_advance_loc: 1 to ffffffffffffd58
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd5b
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd5c
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000000f0 000000000000001c 000000c4 FDE cie=00000030 pc=ffffffffffffd5d..ffffffffffffd63
  DW_CFA_advance_loc: 1 to ffffffffffffd5e
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd61
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd62
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000110 000000000000001c 000000e4 FDE cie=00000030 pc=ffffffffffffd63..ffffffffffffd69
  DW_CFA_advance_loc: 1 to ffffffffffffd64
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd67
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd68
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000130 000000000000001c 00000104 FDE cie=00000030 pc=ffffffffffffd69..ffffffffffffd6f
  DW_CFA_advance_loc: 1 to ffffffffffffd6a
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd6d
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd6e
  DW_CFA_def_cfa: r7 (rsp) ofs 8
```

```
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000150 000000000000001c 00000124 FDE cie=00000030 pc=ffffffffffffd6f..ffffffffffffd75
  DW_CFA_advance_loc: 1 to ffffffffffffd70
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd73
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd74
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000170 000000000000001c 00000144 FDE cie=00000030 pc=ffffffffffffd75..ffffffffffffd7b
  DW_CFA_advance_loc: 1 to ffffffffffffd76
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd79
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd7a
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000190 000000000000001c 00000164 FDE cie=00000030 pc=ffffffffffffd7b..ffffffffffffd81
  DW_CFA_advance_loc: 1 to ffffffffffffd7c
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd7f
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd80
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000001b0 000000000000001c 00000184 FDE cie=00000030 pc=ffffffffffffd81..ffffffffffffd87
  DW_CFA_advance_loc: 1 to ffffffffffffd82
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd85
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd86
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000001d0 000000000000001c 000001a4 FDE cie=00000030 pc=ffffffffffffd87..ffffffffffffd8d
  DW_CFA_advance_loc: 1 to ffffffffffffd88
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd8b
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd8c
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000001f0 000000000000001c 000001c4 FDE cie=00000030 pc=ffffffffffffd8d..ffffffffffffd93
  DW_CFA_advance_loc: 1 to ffffffffffffd8e
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd91
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd92
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000210 000000000000001c 000001e4 FDE cie=00000030 pc=ffffffffffffd93..ffffffffffffd99
  DW_CFA_advance_loc: 1 to ffffffffffffd94
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd97
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd98
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000230 000000000000001c 00000204 FDE cie=00000030 pc=ffffffffffffd99..ffffffffffffd9f
  DW_CFA_advance_loc: 1 to ffffffffffffd9a
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffd9d
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffd9e
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000250 000000000000001c 00000224 FDE cie=00000030 pc=ffffffffffffd9f..ffffffffffffda5
  DW_CFA_advance_loc: 1 to ffffffffffffda0
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffda3
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffda4
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000270 000000000000001c 00000244 FDE cie=00000030 pc=ffffffffffffda5..ffffffffffffdab
  DW_CFA_advance_loc: 1 to ffffffffffffda6
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffda9
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdaa
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000290 000000000000001c 00000264 FDE cie=00000030 pc=ffffffffffffdab..ffffffffffffdb1
  DW_CFA_advance_loc: 1 to ffffffffffffdac
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdaf
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdb0
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
```

```
000002b0 000000000000001c 00000284 FDE cie=00000030 pc=ffffffffffffdb1..ffffffffffffdb7
  DW_CFA_advance_loc: 1 to ffffffffffffdb2
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdb5
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdb6
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000002d0 000000000000001c 000002a4 FDE cie=00000030 pc=ffffffffffffdb7..ffffffffffffdbd
  DW_CFA_advance_loc: 1 to ffffffffffffdb8
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdbb
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdbc
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000002f0 000000000000001c 000002c4 FDE cie=00000030 pc=ffffffffffffdbd..ffffffffffffdc3
  DW_CFA_advance_loc: 1 to ffffffffffffdbe
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdc1
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdc2
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000310 000000000000001c 000002e4 FDE cie=00000030 pc=ffffffffffffdc3..ffffffffffffdc9
  DW_CFA_advance_loc: 1 to ffffffffffffdc4
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdc7
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdc8
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000330 000000000000001c 00000304 FDE cie=00000030 pc=ffffffffffffdc9..ffffffffffffdcf
  DW_CFA_advance_loc: 1 to ffffffffffffdca
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdcd
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdce
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000350 000000000000001c 00000324 FDE cie=00000030 pc=ffffffffffffdcf..ffffffffffffdd5
  DW_CFA_advance_loc: 1 to ffffffffffffdd0
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdd3
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc: 1 to ffffffffffffdd4
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

00000370 000000000000001c 00000344 FDE cie=00000030 pc=ffffffffffffdd5..fffffffffffffe4e
  DW_CFA_advance_loc: 1 to ffffffffffffdd6
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r6 (rbp) at cfa-16
  DW_CFA_advance_loc: 3 to ffffffffffffdd9
  DW_CFA_def_cfa_register: r6 (rbp)
  DW_CFA_advance_loc1: 116 to fffffffffffffe4d
  DW_CFA_def_cfa: r7 (rsp) ofs 8
  DW_CFA_nop
  DW_CFA_nop

00000390 0000000000000044 00000364 FDE cie=00000030 pc=ffffffffffffe58..fffffffffffffebd
  DW_CFA_advance_loc: 2 to ffffffffffffe5a
  DW_CFA_def_cfa_offset: 16
  DW_CFA_offset: r15 (r15) at cfa-16
  DW_CFA_advance_loc: 5 to ffffffffffffe5f
  DW_CFA_def_cfa_offset: 24
  DW_CFA_offset: r14 (r14) at cfa-24
  DW_CFA_advance_loc: 5 to ffffffffffffe64
  DW_CFA_def_cfa_offset: 32
  DW_CFA_offset: r13 (r13) at cfa-32
  DW_CFA_advance_loc: 5 to ffffffffffffe69
  DW_CFA_def_cfa_offset: 40
  DW_CFA_offset: r12 (r12) at cfa-40
  DW_CFA_advance_loc: 8 to ffffffffffffe71
  DW_CFA_def_cfa_offset: 48
  DW_CFA_offset: r6 (rbp) at cfa-48
  DW_CFA_advance_loc: 8 to ffffffffffffe79
  DW_CFA_def_cfa_offset: 56
  DW_CFA_offset: r3 (rbx) at cfa-56
  DW_CFA_advance_loc: 13 to ffffffffffffe86
  DW_CFA_def_cfa_offset: 64
  DW_CFA_advance_loc: 44 to ffffffffffffeb2
  DW_CFA_def_cfa_offset: 56
  DW_CFA_advance_loc: 1 to ffffffffffffeb3
  DW_CFA_def_cfa_offset: 48
  DW_CFA_advance_loc: 1 to ffffffffffffeb4
  DW_CFA_def_cfa_offset: 40
  DW_CFA_advance_loc: 2 to ffffffffffffeb6
  DW_CFA_def_cfa_offset: 32
  DW_CFA_advance_loc: 2 to ffffffffffffeb8
  DW_CFA_def_cfa_offset: 24
  DW_CFA_advance_loc: 2 to ffffffffffffeba
  DW_CFA_def_cfa_offset: 16
  DW_CFA_advance_loc: 2 to ffffffffffffebc
  DW_CFA_def_cfa_offset: 8
  DW_CFA_nop

000003d8 0000000000000014 000003ac FDE cie=00000030 pc=ffffffffffffec8..ffffffffffffeca
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop
  DW_CFA_nop

000003f0 ZERO terminator


Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -a 2
```

```
2:     file format elf64-x86-64
2

Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump  2
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -s  2

2:     file format elf64-x86-64

Contents of section .interp:
 400238 2f6c6962 36342f6c 642d6c69 6e75782d  /lib64/ld-linux-
 400248 7838362d 36342e73 6f2e3200           x86-64.so.2.
Contents of section .note.ABI-tag:
 400254 04000000 10000000 01000000 474e5500  ............GNU.
 400264 00000000 02000000 06000000 18000000  ................
Contents of section .note.gnu.build-id:
 400274 04000000 14000000 03000000 474e5500  ............GNU.
 400284 18ada6ba 56c33dea c975256e fe3955a5  ....V.=..u%n.9U.
 400294 4be4a407                             K...
Contents of section .gnu.hash:
 400298 03000000 09000000 01000000 06000000  ................
 4002a8 88c02001 00044009 09000000 0b000000  .. ...@.........
 4002b8 0d000000 4245d5ec bbe3927c b88df10e  ....BE.....|....
 4002c8 d971581c ebd3ef0e                    .qX.....
Contents of section .dynsym:
 4002d0 00000000 00000000 00000000 00000000  ................
 4002e0 00000000 00000000 6e000000 12000000  ........n.......
 4002f0 00000000 00000000 00000000 00000000  ................
 400300 8e000000 12000000 00000000 00000000  ................
 400310 00000000 00000000 24000000 20000000  ........$... ...
 400320 00000000 00000000 00000000 00000000  ................
 400330 7d000000 12000000 00000000 00000000  }...............
 400340 00000000 00000000 38000000 20000000  ........8... ...
 400350 00000000 00000000 00000000 00000000  ................
 400360 aa000000 12000000 00000000 00000000  ................
 400370 00000000 00000000 15000000 20000000  ............ ...
 400380 00000000 00000000 00000000 00000000  ................
 400390 54000000 20000000 00000000 00000000  T... ...........
 4003a0 00000000 00000000 c2000000 10001800  ................
 4003b0 50206000 00000000 00000000 00000000  P `.............
 4003c0 d5000000 10001900 58206000 00000000  ........X `.....
 4003d0 00000000 00000000 77000000 12000b00  ........w.......
 4003e0 00064000 00000000 00000000 00000000  ..@.............
 4003f0 c9000000 10001900 50206000 00000000  ........P `.....
 400400 00000000 00000000 bc000000 12000e00  ................
 400410 f4084000 00000000 00000000 00000000  ..@.............
Contents of section .dynstr:
 400420 006c6962 6375726c 2d676e75 746c732e  .libcurl-gnutls.
 400430 736f2e34 005f5f67 6d6f6e5f 73746172  so.4.__gmon_star
 400440 745f5f00 5f4a765f 52656769 73746572  t__._Jv_Register
 400450 436c6173 73657300 5f49544d 5f646572  Classes._ITM_der
 400460 65676973 74657254 4d436c6f 6e655461  egisterTMCloneTa
 400470 626c6500 5f49544d 5f726567 69737465  ble._ITM_registe
 400480 72544d43 6c6f6e65 5461626c 65006375  rTMCloneTable.cu
 400490 726c5f65 6173795f 696e6974 00637572  rl_easy_init.cur
 4004a0 6c5f6561 73795f73 65746f70 74006375  l_easy_setopt.cu
 4004b0 726c5f65 6173795f 70657266 6f726d00  rl_easy_perform.
 4004c0 6c696263 2e736f2e 36005f5f 6c696263  libc.so.6.__libc
 4004d0 5f737461 72745f6d 61696e00 5f66696e  _start_main._fin
 4004e0 69005f65 64617461 005f5f62 73735f73  i._edata.__bss_s
 4004f0 74617274 005f656e 6400474c 4942435f  tart._end.GLIBC_
 400500 322e322e 35004355 524c5f47 4e55544c  2.2.5.CURL_GNUTL
 400510 535f3300                             S_3.
Contents of section .gnu.version:
 400514 00000200 02000000 02000000 03000000  ................
 400524 00000100 01000100 01000100           ............
Contents of section .gnu.version_r:
 400530 01000100 a0000000 10000000 20000000  ............ ...
 400540 751a6909 00000300 da000000 00000000  u.i.............
 400550 01000100 01000000 10000000 00000000  ................
 400560 233d100b 00000200 e6000000 00000000  #=..............
Contents of section .rela.dyn:
 400570 f81f6000 00000000 06000000 07000000  ..`.............
 400580 00000000 00000000                    ........
Contents of section .rela.plt:
 400588 18206000 00000000 07000000 01000000  . `.............
 400598 00000000 00000000 20206000 00000000  ........  `.....
 4005a8 07000000 02000000 00000000 00000000  ................
 4005b8 28206000 00000000 07000000 04000000  ( `.............
 4005c8 00000000 00000000 30206000 00000000  ........0 `.....
 4005d8 07000000 06000000 00000000 00000000  ................
 4005e8 38206000 00000000 07000000 07000000  8 `.............
 4005f8 00000000 00000000                    ........
Contents of section .init:
 400600 4883ec08 488b05ed 19200048 85c07405  H...H.... .H..t.
 400610 e85b0000 004883c4 08c3               .[...H....
Contents of section .plt:
 400620 ff35e219 2000ff25 e4192000 0f1f4000  .5.. ..%.. ...@.
 400630 ff25e219 20006800 000000e9 e0ffffff  .%.. .h.........
 400640 ff25da19 20006801 000000e9 d0ffffff  .%.. .h.........
 400650 ff25d219 20006802 000000e9 c0ffffff  .%.. .h.........
 400660 ff25ca19 20006803 000000e9 b0ffffff  .%.. .h.........
 400670 ff25c219 20006804 000000e9 a0ffffff  .%.. .h.........
Contents of section .text:
 400680 31ed4989 d15e4889 e24883e4 f0505449  1.I..^H..H...PTI
 400690 c7c0f008 400048c7 c1800840 0048c7c7  ....@.H....@.H..
 4006a0 fd074000 e8b7ffff fff4660f 1f440000  ..@.......f..D..
 4006b0 b8572060 0055482d 50206000 4883f80e  .W `.UH-P `.H...
 4006c0 4889e577 025dc3b8 00000000 4885c074  H..w.]......H..t
 4006d0 f45dbf50 206000ff e00f1f80 00000000  .].P `..........
 4006e0 b8502060 0055482d 50206000 48c1f803  .P `.UH-P `.H...
 4006f0 4889e548 89c248c1 ea3f4801 d048d1f8  H..H..H..?H..H..
 400700 75025dc3 ba000000 004885d2 74f45d48  u.]......H..t.]H
 400710 89c6bf50 206000ff e20f1f80 00000000  ...P `..........
 400720 803d2919 20000075 11554889 e5e87eff  .=). ..u.UH...~.
 400730 ffff5dc6 05161920 0001f3c3 0f1f4000  ..]....  ......@.
```

```
 400740 48833dc8 16200000 741eb800 00000048  H.=.. ..t......H
 400750 85c07414 55bf101e 60004889 e5ffd05d  ..t.U...`.H....]
 400760 e97bffff ff0f1f00 e973ffff ff554889  .{.......s...UH.
 400770 e55dc355 4889e55d c3554889 e55dc355  .].UH..].UH..].U
 400780 4889e55d c3554889 e55dc355 4889e55d  H..].UH..].UH..]
 400790 c3554889 e55dc355 4889e55d c3554889  .UH..].UH..].UH.
 4007a0 e55dc355 4889e55d c3554889 e55dc355  .].UH..].UH..].U
 4007b0 4889e55d c3554889 e55dc355 4889e55d  H..].UH..].UH..]
 4007c0 c3554889 e55dc355 4889e55d c3554889  .UH..].UH..].UH.
 4007d0 e55dc355 4889e55d c3554889 e55dc355  .].UH..].UH..].U
 4007e0 4889e55d c3554889 e55dc355 4889e55d  H..].UH..].UH..]
 4007f0 c3554889 e55dc355 4889e55d c3554889  .UH..].UH..].UH.
 400800 e54883ec 20e826fe ffff4889 45f84883  .H.. .&...H.E.H.
 400810 7df80074 5ac745ec 12270000 8b4dec48  }..tZ.E..'...M.H
 400820 8b45f8ba 08094000 89ce4889 c7b80000  .E....@...H.....
 400830 0000e819 feffffc7 45f03400 00008b4d  ........E.4....M
 400840 f0488b45 f8ba0100 000089ce 4889c7b8  .H.E........H...
 400850 00000000 e8f7fdff ff488b45 f84889c7  .........H.E.H..
 400860 e8dbfdff ff8945f4 b8000000 00eb05b8  ......E.........
 400870 01000000 c9c3662e 0f1f8400 00000000  ......f.........
 400880 41574189 ff415649 89f64155 4989d541  AWA..AVI..AUI..A
 400890 544c8d25 68152000 55488d2d 68152000  TL.%h. .UH.-h. .
 4008a0 534c29e5 31db48c1 fd034883 ec08e84d  SL).1.H....H....M
 4008b0 fdffff48 85ed741e 0f1f8400 00000000  ...H..t.........
 4008c0 4c89ea4c 89f64489 ff41ff14 dc4883c3  L..L..D..A...H..
 4008d0 014839eb 75ea4883 c4085b5d 415c415d  .H9.u.H...[]A\A]
 4008e0 415e415f c366662e 0f1f8400 00000000  A^A_.ff.........
 4008f0 f3c3                                 ..
Contents of section .fini:
 4008f4 4883ec08 4883c408 c3                 H...H....
Contents of section .rodata:
 400900 01000200 00000000 68747470 733a2f2f  ........https://
 400910 75696363 732e6769 74687562 2e696f2f  uiccs.github.io/
 400920 63733336 312f6d61 6769632e 68746d6c  cs361/magic.html
 400930 00                                   .
Contents of section .eh_frame_hdr:
 400934 011b033b f0000000 1d000000 ecfcffff  ...;............
 400944 3c010000 4cfdffff 0c010000 39feffff  <...L.......9...
 400954 64010000 3ffeffff 84010000 45feffff  d...?.......E...
 400964 a4010000 4bfeffff c4010000 51feffff  ....K.......Q...
 400974 e4010000 57feffff 04020000 5dfeffff  ....W.......]...
 400984 24020000 63feffff 44020000 69feffff  $...c...D...i...
 400994 64020000 6ffeffff 84020000 75feffff  d...o.......u...
 4009a4 a4020000 7bfeffff c4020000 81feffff  ....{...........
 4009b4 e4020000 87feffff 04030000 8dfeffff  ................
 4009c4 24030000 93feffff 44030000 99feffff  $.......D.......
 4009d4 64030000 9ffeffff 84030000 a5feffff  d...............
 4009e4 a4030000 abfeffff c4030000 b1feffff  ................
 4009f4 e4030000 b7feffff 04040000 bdfeffff  ................
 400a04 24040000 c3feffff 44040000 c9feffff  $.......D.......
 400a14 64040000 4cffffff 84040000 bcffffff  d...L...........
 400a24 cc040000                             ....
Contents of section .eh_frame:
 400a28 14000000 00000000 017a5200 01781001  .........zR..x..
 400a38 1b0c0708 90010710 14000000 1c000000  ................
 400a48 38fcffff 2a000000 00000000 00000000  8...*...........
 400a58 14000000 00000000 017a5200 01781001  .........zR..x..
 400a68 1b0c0708 90010000 24000000 1c000000  ........$.......
 400a78 a8fbffff 60000000 000e1046 0e184a0f  ....`......F..J.
 400a88 0b770880 003f1a3b 2a332422 00000000  .w...?.;*3$"....
 400a98 1c000000 44000000 cdfcffff 06000000  ....D...........
 400aa8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400ab8 1c000000 64000000 b3fcffff 06000000  ....d...........
 400ac8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400ad8 1c000000 84000000 99fcffff 06000000  ................
 400ae8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400af8 1c000000 a4000000 7ffcffff 06000000  ................
 400b08 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400b18 1c000000 c4000000 65fcffff 06000000  ........e.......
 400b28 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400b38 1c000000 e4000000 4bfcffff 06000000  ........K.......
 400b48 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400b58 1c000000 04010000 31fcffff 06000000  ........1.......
 400b68 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400b78 1c000000 24010000 17fcffff 06000000  ....$...........
 400b88 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400b98 1c000000 44010000 fdfbffff 06000000  ....D...........
 400ba8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400bb8 1c000000 64010000 e3fbffff 06000000  ....d...........
 400bc8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400bd8 1c000000 84010000 c9fbffff 06000000  ................
 400be8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400bf8 1c000000 a4010000 affbffff 06000000  ................
 400c08 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400c18 1c000000 c4010000 95fbffff 06000000  ................
 400c28 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400c38 1c000000 e4010000 7bfbffff 06000000  ........{.......
 400c48 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400c58 1c000000 04020000 61fbffff 06000000  ........a.......
 400c68 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400c78 1c000000 24020000 47fbffff 06000000  ....$...G.......
 400c88 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400c98 1c000000 44020000 2dfbffff 06000000  ....D...-.......
 400ca8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400cb8 1c000000 64020000 13fbffff 06000000  ....d...........
 400cc8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400cd8 1c000000 84020000 f9faffff 06000000  ................
 400ce8 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400cf8 1c000000 a4020000 dffaffff 06000000  ................
 400d08 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400d18 1c000000 c4020000 c5faffff 06000000  ................
 400d28 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400d38 1c000000 e4020000 abfaffff 06000000  ................
 400d48 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400d58 1c000000 04030000 91faffff 06000000  ................
 400d68 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400d78 1c000000 24030000 77faffff 06000000  ....$...w.......
 400d88 00410e10 8602430d 06410c07 08000000  .A....C..A......
 400d98 1c000000 44030000 5dfaffff 79000000  ....D...]...y...
 400da8 00410e10 8602430d 0602740c 07080000  .A....C...t.....
 400db8 44000000 64030000 c0faffff 65000000  D...d.......e...
 400dc8 00420e10 8f02450e 188e0345 0e208d04  .B....E....E. ..
 400dd8 450e288c 05480e30 8606480e 3883074d  E.(..H.0..H.8..M
 400de8 0e406c0e 38410e30 410e2842 0e20420e  .@l.8A.0A.(B. B.
 400df8 18420e10 420e0800 14000000 ac030000  .B..B...........
 400e08 e8faffff 02000000 00000000 00000000  ................
 400e18 00000000                             ....
Contents of section .init_array:
 601e00 40074000 00000000                    @.@.....
Contents of section .fini_array:
 601e08 20074000 00000000                     .@.....
Contents of section .jcr:
 601e10 00000000 00000000                    ........
Contents of section .dynamic:
 601e18 01000000 00000000 01000000 00000000  ................
 601e28 01000000 00000000 a0000000 00000000  ................
 601e38 0c000000 00000000 00064000 00000000  ..........@.....
 601e48 0d000000 00000000 f4084000 00000000  ..........@.....
 601e58 19000000 00000000 001e6000 00000000  ..........`.....
 601e68 1b000000 00000000 08000000 00000000  ................
 601e78 1a000000 00000000 081e6000 00000000  ..........`.....
 601e88 1c000000 00000000 08000000 00000000  ................
 601e98 f5feff6f 00000000 98024000 00000000  ...o......@.....
 601ea8 05000000 00000000 20044000 00000000  ........ .@.....
 601eb8 06000000 00000000 d0024000 00000000  ..........@.....
```

```
 601ec8 0a000000 00000000 f4000000 00000000  ................
 601ed8 0b000000 00000000 18000000 00000000  ................
 601ee8 15000000 00000000 00000000 00000000  ................
 601ef8 03000000 00000000 00206000 00000000  ......... `.....
 601f08 02000000 00000000 78000000 00000000  ........x.......
 601f18 14000000 00000000 07000000 00000000  ................
 601f28 17000000 00000000 88054000 00000000  ..........@.....
 601f38 07000000 00000000 70054000 00000000  ........p.@.....
 601f48 08000000 00000000 18000000 00000000  ................
 601f58 09000000 00000000 18000000 00000000  ................
 601f68 fefffff6f 00000000 30054000 00000000  ...o....0.@.....
 601f78 ffffff6f 00000000 02000000 00000000  ...o............
 601f88 f0ffff6f 00000000 14054000 00000000  ...o......@.....
 601f98 00000000 00000000 00000000 00000000  ................
 601fa8 00000000 00000000 00000000 00000000  ................
 601fb8 00000000 00000000 00000000 00000000  ................
 601fc8 00000000 00000000 00000000 00000000  ................
 601fd8 00000000 00000000 00000000 00000000  ................
 601fe8 00000000 00000000 00000000 00000000  ................
Contents of section .got:
 601ff8 00000000 00000000                    ........
Contents of section .got.plt:
 602000 181e6000 00000000 00000000 00000000  ..`.............
 602010 00000000 00000000 36064000 00000000  ........6.@.....
 602020 46064000 00000000 56064000 00000000  F.@.....V.@.....
 602030 66064000 00000000 76064000 00000000  f.@.....v.@.....
Contents of section .data:
 602040 00000000 00000000 00000000 00000000  ................
Contents of section .comment:
 0000 4743433a 20285562 756e7475 20342e38  GCC: (Ubuntu 4.8
 0010 2e342d32 7562756e 7475317e 31342e30  .4-2ubuntu1~14.0
 0020 34292034 2e382e34 00474343 3a202855  4) 4.8.4.GCC: (U
 0030 62756e74 7520342e 382e322d 31397562  buntu 4.8.2-19ub
 0040 756e7475 31292034 2e382e32 00        untu1) 4.8.2.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump 2
Usage: objdump <option(s)> <file(s)>
 Display information from object <file(s)>.
 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -P, --private=OPT,OPT... Display object format specific contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -e, --debugging-tags     Display debug information using ctags style
  -G, --stabs              Display (in raw form) any STABS info in the file
  -W[lLiaprmfFsoRt] or
  --dwarf[=rawline,=decodedline,=info,=abbrev,=pubnames,=aranges,=macro,=frames,
          =frames-interp,=str,=loc,=Ranges,=pubtypes,
          =gdb_index,=trace_info,=trace_abbrev,=trace_aranges,
          =addr,=cu_index]
                           Display DWARF info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  @<file>                  Read options from <file>
  -v, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -S 2

2:     file format elf64-x86-64


Disassembly of section .init:

0000000000400600 <_init>:
  400600:       48 83 ec 08             sub    $0x8,%rsp
  400604:       48 8b 05 ed 19 20 00    mov    0x2019ed(%rip),%rax        # 601ff8 <_fini+0x201704>
  40060b:       48 85 c0                test   %rax,%rax
  40060e:       74 05                   je     400615 <_init+0x15>
  400610:       e8 5b 00 00 00          callq  400670 <__gmon_start__@plt>
  400615:       48 83 c4 08             add    $0x8,%rsp
  400619:       c3                      retq

Disassembly of section .plt:

0000000000400620 <curl_easy_init@plt-0x10>:
  400620:       ff 35 e2 19 20 00       pushq  0x2019e2(%rip)        # 602008 <_fini+0x201714>
  400626:       ff 25 e4 19 20 00       jmpq   *0x2019e4(%rip)        # 602010 <_fini+0x20171c>
  40062c:       0f 1f 40 00             nopl   0x0(%rax)

0000000000400630 <curl_easy_init@plt>:
  400630:       ff 25 e2 19 20 00       jmpq   *0x2019e2(%rip)        # 602018 <_fini+0x201724>
  400636:       68 00 00 00 00          pushq  $0x0
  40063b:       e9 e0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400640 <curl_easy_perform@plt>:
  400640:       ff 25 da 19 20 00       jmpq   *0x2019da(%rip)        # 602020 <_fini+0x20172c>
  400646:       68 01 00 00 00          pushq  $0x1
  40064b:       e9 d0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400650 <curl_easy_setopt@plt>:
  400650:       ff 25 d2 19 20 00       jmpq   *0x2019d2(%rip)        # 602028 <_fini+0x201734>
  400656:       68 02 00 00 00          pushq  $0x2
  40065b:       e9 c0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400660 <__libc_start_main@plt>:
  400660:       ff 25 ca 19 20 00       jmpq   *0x2019ca(%rip)        # 602030 <_fini+0x20173c>
  400666:       68 03 00 00 00          pushq  $0x3
  40066b:       e9 b0 ff ff ff          jmpq   400620 <_init+0x20>

0000000000400670 <__gmon_start__@plt>:
  400670:       ff 25 c2 19 20 00       jmpq   *0x2019c2(%rip)        # 602038 <_fini+0x201744>
  400676:       68 04 00 00 00          pushq  $0x4
  40067b:       e9 a0 ff ff ff          jmpq   400620 <_init+0x20>

Disassembly of section .text:

0000000000400680 <.text>:
  400680:       31 ed                   xor    %ebp,%ebp
  400682:       49 89 d1                mov    %rdx,%r9
  400685:       5e                      pop    %rsi
  400686:       48 89 e2                mov    %rsp,%rdx
  400689:       48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40068d:       50                      push   %rax
  40068e:       54                      push   %rsp
  40068f:       49 c7 c0 f0 08 40 00    mov    $0x4008f0,%r8
  400696:       48 c7 c1 80 08 40 00    mov    $0x400880,%rcx
  40069d:       48 c7 c7 fd 07 40 00    mov    $0x4007fd,%rdi
  4006a4:       e8 b7 ff ff ff          callq  400660 <__libc_start_main@plt>
  4006a9:       f4                      hlt
  4006aa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  4006b0:       b8 57 20 60 00          mov    $0x602057,%eax
  4006b5:       55                      push   %rbp
  4006b6:       48 2d 50 20 60 00       sub    $0x602050,%rax
  4006bc:       48 83 f8 0e             cmp    $0xe,%rax
  4006c0:       48 89 e5                mov    %rsp,%rbp
  4006c3:       77 02                   ja     4006c7 <__gmon_start__@plt+0x57>
```

```
4006c5:       5d                      pop    %rbp
4006c6:       c3                      retq
4006c7:       b8 00 00 00 00          mov    $0x0,%eax
4006cc:       48 85 c0                test   %rax,%rax
4006cf:       74 f4                   je     4006c5 <__gmon_start__@plt+0x55>
4006d1:       5d                      pop    %rbp
4006d2:       bf 50 20 60 00          mov    $0x602050,%edi
4006d7:       ff e0                   jmpq   *%rax
4006d9:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
4006e0:       b8 50 20 60 00          mov    $0x602050,%eax
4006e5:       55                      push   %rbp
4006e6:       48 2d 50 20 60 00       sub    $0x602050,%rax
4006ec:       48 c1 f8 03             sar    $0x3,%rax
4006f0:       48 89 e5                mov    %rsp,%rbp
4006f3:       48 89 c2                mov    %rax,%rdx
4006f6:       48 c1 ea 3f             shr    $0x3f,%rdx
4006fa:       48 01 d0                add    %rdx,%rax
4006fd:       48 d1 f8                sar    %rax
400700:       75 02                   jne    400704 <__gmon_start__@plt+0x94>
400702:       5d                      pop    %rbp
400703:       c3                      retq
400704:       ba 00 00 00 00          mov    $0x0,%edx
400709:       48 85 d2                test   %rdx,%rdx
40070c:       74 f4                   je     400702 <__gmon_start__@plt+0x92>
40070e:       5d                      pop    %rbp
40070f:       48 89 c6                mov    %rax,%rsi
400712:       bf 50 20 60 00          mov    $0x602050,%edi
400717:       ff e2                   jmpq   *%rdx
400719:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
400720:       80 3d 29 19 20 00 00    cmpb   $0x0,0x201929(%rip)        # 602050 <_edata>
400727:       75 11                   jne    40073a <__gmon_start__@plt+0xca>
400729:       55                      push   %rbp
40072a:       48 89 e5                mov    %rsp,%rbp
40072d:       e8 7e ff ff ff          callq  4006b0 <__gmon_start__@plt+0x40>
400732:       5d                      pop    %rbp
400733:       c6 05 16 19 20 00 01    movb   $0x1,0x201916(%rip)        # 602050 <_edata>
40073a:       f3 c3                   repz retq
40073c:       0f 1f 40 00             nopl   0x0(%rax)
400740:       48 83 3d c8 16 20 00    cmpq   $0x0,0x2016c8(%rip)        # 601e10 <_fini+0x20151c>
400747:       00
400748:       74 1e                   je     400768 <__gmon_start__@plt+0xf8>
40074a:       b8 00 00 00 00          mov    $0x0,%eax
40074f:       48 85 c0                test   %rax,%rax
400752:       74 14                   je     400768 <__gmon_start__@plt+0xf8>
400754:       55                      push   %rbp
400755:       bf 10 1e 60 00          mov    $0x601e10,%edi
40075a:       48 89 e5                mov    %rsp,%rbp
40075d:       ff d0                   callq  *%rax
40075f:       5d                      pop    %rbp
400760:       e9 7b ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
400765:       0f 1f 00                nopl   (%rax)
400768:       e9 73 ff ff ff          jmpq   4006e0 <__gmon_start__@plt+0x70>
40076d:       55                      push   %rbp
40076e:       48 89 e5                mov    %rsp,%rbp
400771:       5d                      pop    %rbp
400772:       c3                      retq
400773:       55                      push   %rbp
400774:       48 89 e5                mov    %rsp,%rbp
400777:       5d                      pop    %rbp
400778:       c3                      retq
400779:       55                      push   %rbp
40077a:       48 89 e5                mov    %rsp,%rbp
40077d:       5d                      pop    %rbp
40077e:       c3                      retq
40077f:       55                      push   %rbp
400780:       48 89 e5                mov    %rsp,%rbp
400783:       5d                      pop    %rbp
400784:       c3                      retq
400785:       55                      push   %rbp
400786:       48 89 e5                mov    %rsp,%rbp
400789:       5d                      pop    %rbp
40078a:       c3                      retq
40078b:       55                      push   %rbp
40078c:       48 89 e5                mov    %rsp,%rbp
40078f:       5d                      pop    %rbp
400790:       c3                      retq
400791:       55                      push   %rbp
400792:       48 89 e5                mov    %rsp,%rbp
400795:       5d                      pop    %rbp
400796:       c3                      retq
400797:       55                      push   %rbp
400798:       48 89 e5                mov    %rsp,%rbp
40079b:       5d                      pop    %rbp
40079c:       c3                      retq
40079d:       55                      push   %rbp
40079e:       48 89 e5                mov    %rsp,%rbp
4007a1:       5d                      pop    %rbp
4007a2:       c3                      retq
4007a3:       55                      push   %rbp
4007a4:       48 89 e5                mov    %rsp,%rbp
4007a7:       5d                      pop    %rbp
4007a8:       c3                      retq
4007a9:       55                      push   %rbp
4007aa:       48 89 e5                mov    %rsp,%rbp
4007ad:       5d                      pop    %rbp
4007ae:       c3                      retq
4007af:       55                      push   %rbp
4007b0:       48 89 e5                mov    %rsp,%rbp
4007b3:       5d                      pop    %rbp
4007b4:       c3                      retq
4007b5:       55                      push   %rbp
4007b6:       48 89 e5                mov    %rsp,%rbp
4007b9:       5d                      pop    %rbp
4007ba:       c3                      retq
4007bb:       55                      push   %rbp
4007bc:       48 89 e5                mov    %rsp,%rbp
4007bf:       5d                      pop    %rbp
4007c0:       c3                      retq
4007c1:       55                      push   %rbp
4007c2:       48 89 e5                mov    %rsp,%rbp
4007c5:       5d                      pop    %rbp
4007c6:       c3                      retq
4007c7:       55                      push   %rbp
4007c8:       48 89 e5                mov    %rsp,%rbp
4007cb:       5d                      pop    %rbp
4007cc:       c3                      retq
4007cd:       55                      push   %rbp
4007ce:       48 89 e5                mov    %rsp,%rbp
4007d1:       5d                      pop    %rbp
4007d2:       c3                      retq
4007d3:       55                      push   %rbp
4007d4:       48 89 e5                mov    %rsp,%rbp
4007d7:       5d                      pop    %rbp
4007d8:       c3                      retq
4007d9:       55                      push   %rbp
4007da:       48 89 e5                mov    %rsp,%rbp
4007dd:       5d                      pop    %rbp
4007de:       c3                      retq
4007df:       55                      push   %rbp
4007e0:       48 89 e5                mov    %rsp,%rbp
4007e3:       5d                      pop    %rbp
4007e4:       c3                      retq
4007e5:       55                      push   %rbp
4007e6:       48 89 e5                mov    %rsp,%rbp
4007e9:       5d                      pop    %rbp
```

```
  4007ea:       c3                      retq
  4007eb:       55                      push   %rbp
  4007ec:       48 89 e5                mov    %rsp,%rbp
  4007ef:       5d                      pop    %rbp
  4007f0:       c3                      retq
  4007f1:       55                      push   %rbp
  4007f2:       48 89 e5                mov    %rsp,%rbp
  4007f5:       5d                      pop    %rbp
  4007f6:       c3                      retq
  4007f7:       55                      push   %rbp
  4007f8:       48 89 e5                mov    %rsp,%rbp
  4007fb:       5d                      pop    %rbp
  4007fc:       c3                      retq
  4007fd:       55                      push   %rbp
  4007fe:       48 89 e5                mov    %rsp,%rbp
  400801:       48 83 ec 20             sub    $0x20,%rsp
  400805:       e8 26 fe ff ff          callq  400630 <curl_easy_init@plt>
  40080a:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  40080e:       48 83 7d f8 00          cmpq   $0x0,-0x8(%rbp)
  400813:       74 5a                   je     40086f <__gmon_start__@plt+0x1ff>
  400815:       c7 45 ec 12 27 00 00    movl   $0x2712,-0x14(%rbp)
  40081c:       8b 4d ec                mov    -0x14(%rbp),%ecx
  40081f:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400823:       ba 08 09 40 00          mov    $0x400908,%edx
  400828:       89 ce                   mov    %ecx,%esi
  40082a:       48 89 c7                mov    %rax,%rdi
  40082d:       b8 00 00 00 00          mov    $0x0,%eax
  400832:       e8 19 fe ff ff          callq  400650 <curl_easy_setopt@plt>
  400837:       c7 45 f0 34 00 00 00    movl   $0x34,-0x10(%rbp)
  40083e:       8b 4d f0                mov    -0x10(%rbp),%ecx
  400841:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400845:       ba 01 00 00 00          mov    $0x1,%edx
  40084a:       89 ce                   mov    %ecx,%esi
  40084c:       48 89 c7                mov    %rax,%rdi
  40084f:       b8 00 00 00 00          mov    $0x0,%eax
  400854:       e8 f7 fd ff ff          callq  400650 <curl_easy_setopt@plt>
  400859:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  40085d:       48 89 c7                mov    %rax,%rdi
  400860:       e8 db fd ff ff          callq  400640 <curl_easy_perform@plt>
  400865:       89 45 f4                mov    %eax,-0xc(%rbp)
  400868:       b8 00 00 00 00          mov    $0x0,%eax
  40086d:       eb 05                   jmp    400874 <__gmon_start__@plt+0x204>
  40086f:       b8 01 00 00 00          mov    $0x1,%eax
  400874:       c9                      leaveq
  400875:       c3                      retq
  400876:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  40087d:       00 00 00
  400880:       41 57                   push   %r15
  400882:       41 89 ff                mov    %edi,%r15d
  400885:       41 56                   push   %r14
  400887:       49 89 f6                mov    %rsi,%r14
  40088a:       41 55                   push   %r13
  40088c:       49 89 d5                mov    %rdx,%r13
  40088f:       41 54                   push   %r12
  400891:       4c 8d 25 68 15 20 00    lea    0x201568(%rip),%r12        # 601e00 <_fini+0x20150c>
  400898:       55                      push   %rbp
  400899:       48 8d 2d 68 15 20 00    lea    0x201568(%rip),%rbp        # 601e08 <_fini+0x201514>
  4008a0:       53                      push   %rbx
  4008a1:       4c 29 e5                sub    %r12,%rbp
  4008a4:       31 db                   xor    %ebx,%ebx
  4008a6:       48 c1 fd 03             sar    $0x3,%rbp
  4008aa:       48 83 ec 08             sub    $0x8,%rsp
  4008ae:       e8 4d fd ff ff          callq  400600 <_init>
  4008b3:       48 85 ed                test   %rbp,%rbp
  4008b6:       74 1e                   je     4008d6 <__gmon_start__@plt+0x266>
  4008b8:       0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  4008bf:       00
  4008c0:       4c 89 ea                mov    %r13,%rdx
  4008c3:       4c 89 f6                mov    %r14,%rsi
  4008c6:       44 89 ff                mov    %r15d,%edi
  4008c9:       41 ff 14 dc             callq  *(%r12,%rbx,8)
  4008cd:       48 83 c3 01             add    $0x1,%rbx
  4008d1:       48 39 eb                cmp    %rbp,%rbx
  4008d4:       75 ea                   jne    4008c0 <__gmon_start__@plt+0x250>
  4008d6:       48 83 c4 08             add    $0x8,%rsp
  4008da:       5b                      pop    %rbx
  4008db:       5d                      pop    %rbp
  4008dc:       41 5c                   pop    %r12
  4008de:       41 5d                   pop    %r13
  4008e0:       41 5e                   pop    %r14
  4008e2:       41 5f                   pop    %r15
  4008e4:       c3                      retq
  4008e5:       66 66 2e 0f 1f 84 00    data32 nopw %cs:0x0(%rax,%rax,1)
  4008ec:       00 00 00 00
  4008f0:       f3 c3                   repz retq

Disassembly of section .fini:

00000000004008f4 <_fini>:
  4008f4:       48 83 ec 08             sub    $0x8,%rsp
  4008f8:       48 83 c4 08             add    $0x8,%rsp
  4008fc:       c3                      retq
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./2
The password is "yes" without quotes if the call to curl_easy_perform was successful; the password is "no" without quotes if the call to curl_easy_perform as unsuccessful.
You may not use gdb to answer this question.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf -s 3

Symbol table '.dynsym' contains 11 entries:
   Num:    Value          Size Type    Bind   Vis      Ndx Name
     0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
     1: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND puts@GLIBC_2.2.5 (2)
     2: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND strlen@GLIBC_2.2.5 (2)
     3: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __stack_chk_fail@GLIBC_2.4 (3)
     4: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND printf@GLIBC_2.2.5 (2)
     5: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.2.5 (2)
     6: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND calloc@GLIBC_2.2.5 (2)
     7: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND __gmon_start__
     8: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __xstat@GLIBC_2.2.5 (2)
     9: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND getlogin_r@GLIBC_2.2.5 (2)
    10: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND access@GLIBC_2.2.5 (2)
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf -s 3

Symbol table '.dynsym' contains 11 entries:
   Num:    Value          Size Type    Bind   Vis      Ndx Name
     0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
     1: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND puts@GLIBC_2.2.5 (2)
     2: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND strlen@GLIBC_2.2.5 (2)
     3: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __stack_chk_fail@GLIBC_2.4 (3)
     4: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND printf@GLIBC_2.2.5 (2)
     5: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.2.5 (2)
     6: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND calloc@GLIBC_2.2.5 (2)
     7: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND __gmon_start__
     8: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __xstat@GLIBC_2.2.5 (2)
     9: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND getlogin_r@GLIBC_2.2.5 (2)
    10: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND access@GLIBC_2.2.5 (2)
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
```

```
0  1  2  3  4  howto.txt  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ gdb 3
GNU gdb (Ubuntu 7.7.1-0ubuntu5~14.04.2) 7.7.1
Copyright (C) 2014 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:
<http://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 3...(no debugging symbols found)...done.
(gdb) b 1
No symbol table is loaded.  Use the "file" command.
(gdb) file 3
Reading symbols from 3...(no debugging symbols found)...done.
(gdb) disas
No frame selected.
(gdb) Quit
(gdb) quit
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ clear

Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ strings 3 | grep '\.txt'
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ strings 3
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
__stack_chk_fail
printf
calloc
strlen
getlogin_r
access
__libc_start_main
__xstat
__gmon_start__
GLIBC_2.4
GLIBC_2.2.5
dH34%(
[]A\A]A^A_
iamspecial
234567890123456789O1
%s%s
behold! I will only tell you the secret password if you enter the random number I just generated!
you win! the secret is:
beelzebub
checking for the existence of a file with a special name...
correct file found.
check failed.
checking for appropriate access rights...
correct rights found.
;*3$"
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
GCC: (Ubuntu 4.8.4-2ubuntu1~14.04) 4.8.4
GCC: (Ubuntu 4.8.2-19ubuntu1) 4.8.2
.shstrtab
.interp
.note.ABI-tag
.note.gnu.build-id
.gnu.hash
.dynsym
.dynstr
.gnu.version
.gnu.version_r
.rela.dyn
.rela.plt
.init
.text
.fini
.rodata
.eh_frame_hdr
.eh_frame
.init_array
.fini_array
.jcr
.dynamic
.got
.got.plt
.data
.bss
.comment
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi beelzebub
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  beelzebub  howto.txt  iamspecial  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ rm beelzebub
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  iamspecial  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./imspecial
-bash: ./imspecial: No such file or directory
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ clear

Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ strings 3
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
__stack_chk_fail
printf
calloc
strlen
getlogin_r
access
__libc_start_main
__xstat
__gmon_start__
GLIBC_2.4
GLIBC_2.2.5
dH34%(
[]A\A]A^A_
```

```
iamspecial
23456789012345678901
%s%s
behold! I will only tell you the secret password if you enter the random number I just generated!
you win! the secret is:
beelzebub
checking for the existence of a file with a special name...
correct file found.
check failed.
checking for appropriate access rights...
correct rights found.
;*3$"
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
GCC: (Ubuntu 4.8.4-2ubuntu1~14.04) 4.8.4
GCC: (Ubuntu 4.8.2-19ubuntu1) 4.8.2
.shstrtab
.interp
.note.ABI-tag
.note.gnu.build-id
.gnu.hash
.dynsym
.dynstr
.gnu.version
.gnu.version_r
.rela.dyn
.rela.plt
.init
.text
.fini
.rodata
.eh_frame_hdr
.eh_frame
.init_array
.fini_array
.jcr
.dynamic
.got
.got.plt
.data
.bss
.comment
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ strings 3
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
__stack_chk_fail
printf
calloc
strlen
getlogin_r
access
__libc_start_main
__xstat
__gmon_start__
GLIBC_2.4
GLIBC_2.2.5
dH34%(
[]A\A]A^A_
iamspecial
23456789012345678901
%s%s
behold! I will only tell you the secret password if you enter the random number I just generated!
you win! the secret is:
beelzebub
checking for the existence of a file with a special name...
correct file found.
check failed.
checking for appropriate access rights...
correct rights found.
;*3$"
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
GCC: (Ubuntu 4.8.4-2ubuntu1~14.04) 4.8.4
GCC: (Ubuntu 4.8.2-19ubuntu1) 4.8.2
.shstrtab
.interp
.note.ABI-tag
.note.gnu.build-id
.gnu.hash
.dynsym
.dynstr
.gnu.version
.gnu.version_r
.rela.dyn
.rela.plt
.init
.text
.fini
.rodata
.eh_frame_hdr
.eh_frame
.init_array
.fini_array
.jcr
.dynamic
.got
.got.plt
.data
.bss
.comment
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
```

```
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ readelf -s 3

Symbol table '.dynsym' contains 11 entries:
   Num:    Value          Size Type    Bind   Vis      Ndx Name
     0: 0000000000000000     0 NOTYPE  LOCAL  DEFAULT  UND
     1: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND puts@GLIBC_2.2.5 (2)
     2: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND strlen@GLIBC_2.2.5 (2)
     3: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __stack_chk_fail@GLIBC_2.4 (3)
     4: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND printf@GLIBC_2.2.5 (2)
     5: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __libc_start_main@GLIBC_2.2.5 (2)
     6: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND calloc@GLIBC_2.2.5 (2)
     7: 0000000000000000     0 NOTYPE  WEAK   DEFAULT  UND __gmon_start__
     8: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND __xstat@GLIBC_2.2.5 (2)
     9: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND getlogin_r@GLIBC_2.2.5 (2)
    10: 0000000000000000     0 FUNC    GLOBAL DEFAULT  UND access@GLIBC_2.2.5 (2)
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ objdump -S 3

3:     file format elf64-x86-64


Disassembly of section .init:

0000000000400598 <.init>:
  400598:       48 83 ec 08             sub    $0x8,%rsp
  40059c:       48 8b 05 55 1a 20 00    mov    0x201a55(%rip),%rax        # 601ff8 <access@plt+0x201998>
  4005a3:       48 85 c0                test   %rax,%rax
  4005a6:       74 05                   je     4005ad <puts@plt-0x23>
  4005a8:       e8 83 00 00 00          callq  400630 <__gmon_start__@plt>
  4005ad:       48 83 c4 08             add    $0x8,%rsp
  4005b1:       c3                      retq

Disassembly of section .plt:

00000000004005c0 <puts@plt-0x10>:
  4005c0:       ff 35 42 1a 20 00       pushq  0x201a42(%rip)        # 602008 <access@plt+0x2019a8>
  4005c6:       ff 25 44 1a 20 00       jmpq   *0x201a44(%rip)        # 602010 <access@plt+0x2019b0>
  4005cc:       0f 1f 40 00             nopl   0x0(%rax)

00000000004005d0 <puts@plt>:
  4005d0:       ff 25 42 1a 20 00       jmpq   *0x201a42(%rip)        # 602018 <access@plt+0x2019b8>
  4005d6:       68 00 00 00 00          pushq  $0x0
  4005db:       e9 e0 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

00000000004005e0 <strlen@plt>:
  4005e0:       ff 25 3a 1a 20 00       jmpq   *0x201a3a(%rip)        # 602020 <access@plt+0x2019c0>
  4005e6:       68 01 00 00 00          pushq  $0x1
  4005eb:       e9 d0 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

00000000004005f0 <__stack_chk_fail@plt>:
  4005f0:       ff 25 32 1a 20 00       jmpq   *0x201a32(%rip)        # 602028 <access@plt+0x2019c8>
  4005f6:       68 02 00 00 00          pushq  $0x2
  4005fb:       e9 c0 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400600 <printf@plt>:
  400600:       ff 25 2a 1a 20 00       jmpq   *0x201a2a(%rip)        # 602030 <access@plt+0x2019d0>
  400606:       68 03 00 00 00          pushq  $0x3
  40060b:       e9 b0 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400610 <__libc_start_main@plt>:
  400610:       ff 25 22 1a 20 00       jmpq   *0x201a22(%rip)        # 602038 <access@plt+0x2019d8>
  400616:       68 04 00 00 00          pushq  $0x4
  40061b:       e9 a0 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400620 <calloc@plt>:
  400620:       ff 25 1a 1a 20 00       jmpq   *0x201a1a(%rip)        # 602040 <access@plt+0x2019e0>
  400626:       68 05 00 00 00          pushq  $0x5
  40062b:       e9 90 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400630 <__gmon_start__@plt>:
  400630:       ff 25 12 1a 20 00       jmpq   *0x201a12(%rip)        # 602048 <access@plt+0x2019e8>
  400636:       68 06 00 00 00          pushq  $0x6
  40063b:       e9 80 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400640 <__xstat@plt>:
  400640:       ff 25 0a 1a 20 00       jmpq   *0x201a0a(%rip)        # 602050 <access@plt+0x2019f0>
  400646:       68 07 00 00 00          pushq  $0x7
  40064b:       e9 70 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400650 <getlogin_r@plt>:
  400650:       ff 25 02 1a 20 00       jmpq   *0x201a02(%rip)        # 602058 <access@plt+0x2019f8>
  400656:       68 08 00 00 00          pushq  $0x8
  40065b:       e9 60 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

0000000000400660 <access@plt>:
  400660:       ff 25 fa 19 20 00       jmpq   *0x2019fa(%rip)        # 602060 <access@plt+0x201a00>
  400666:       68 09 00 00 00          pushq  $0x9
  40066b:       e9 50 ff ff ff          jmpq   4005c0 <puts@plt-0x10>

Disassembly of section .text:

0000000000400670 <.text>:
  400670:       31 ed                   xor    %ebp,%ebp
  400672:       49 89 d1                mov    %rdx,%r9
  400675:       5e                      pop    %rsi
  400676:       48 89 e2                mov    %rsp,%rdx
  400679:       48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40067d:       50                      push   %rax
  40067e:       54                      push   %rsp
  40067f:       49 c7 c0 a0 0b 40 00    mov    $0x400ba0,%r8
  400686:       48 c7 c1 30 0b 40 00    mov    $0x400b30,%rcx
  40068d:       48 c7 c7 5d 07 40 00    mov    $0x40075d,%rdi
  400694:       e8 77 ff ff ff          callq  400610 <__libc_start_main@plt>
  400699:       f4                      hlt
  40069a:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  4006a0:       b8 17 21 60 00          mov    $0x602117,%eax
  4006a5:       55                      push   %rbp
  4006a6:       48 2d 10 21 60 00       sub    $0x602110,%rax
  4006ac:       48 83 f8 0e             cmp    $0xe,%rax
  4006b0:       48 89 e5                mov    %rsp,%rbp
```

```
4006b3:       77 02                   ja     4006b7 <access@plt+0x57>
4006b5:       5d                      pop    %rbp
4006b6:       c3                      retq
4006b7:       b8 00 00 00 00          mov    $0x0,%eax
4006bc:       48 85 c0                test   %rax,%rax
4006bf:       74 f4                   je     4006b5 <access@plt+0x55>
4006c1:       5d                      pop    %rbp
4006c2:       bf 10 21 60 00          mov    $0x602110,%edi
4006c7:       ff e0                   jmpq   *%rax
4006c9:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
4006d0:       b8 10 21 60 00          mov    $0x602110,%eax
4006d5:       55                      push   %rbp
4006d6:       48 2d 10 21 60 00       sub    $0x602110,%rax
4006dc:       48 c1 f8 03             sar    $0x3,%rax
4006e0:       48 89 e5                mov    %rsp,%rbp
4006e3:       48 89 c2                mov    %rax,%rdx
4006e6:       48 c1 ea 3f             shr    $0x3f,%rdx
4006ea:       48 01 d0                add    %rdx,%rax
4006ed:       48 d1 f8                sar    %rax
4006f0:       75 02                   jne    4006f4 <access@plt+0x94>
4006f2:       5d                      pop    %rbp
4006f3:       c3                      retq
4006f4:       ba 00 00 00 00          mov    $0x0,%edx
4006f9:       48 85 d2                test   %rdx,%rdx
4006fc:       74 f4                   je     4006f2 <access@plt+0x92>
4006fe:       5d                      pop    %rbp
4006ff:       48 89 c6                mov    %rax,%rsi
400702:       bf 10 21 60 00          mov    $0x602110,%edi
400707:       ff e2                   jmpq   *%rdx
400709:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
400710:       80 3d f5 19 20 00 00    cmpb   $0x0,0x2019f5(%rip)        # 60210c <access@plt+0x201aac>
400717:       75 11                   jne    40072a <access@plt+0xca>
400719:       55                      push   %rbp
40071a:       48 89 e5                mov    %rsp,%rbp
40071d:       e8 7e ff ff ff          callq  4006a0 <access@plt+0x40>
400722:       5d                      pop    %rbp
400723:       c6 05 e2 19 20 00 01    movb   $0x1,0x2019e2(%rip)        # 60210c <access@plt+0x201aac>
40072a:       f3 c3                   repz retq
40072c:       0f 1f 40 00             nopl   0x0(%rax)
400730:       48 83 3d e8 16 20 00    cmpq   $0x0,0x2016e8(%rip)        # 601e20 <access@plt+0x2017c0>
400737:       00
400738:       74 1e                   je     400758 <access@plt+0xf8>
40073a:       b8 00 00 00 00          mov    $0x0,%eax
40073f:       48 85 c0                test   %rax,%rax
400742:       74 14                   je     400758 <access@plt+0xf8>
400744:       55                      push   %rbp
400745:       bf 20 1e 60 00          mov    $0x601e20,%edi
40074a:       48 89 e5                mov    %rsp,%rbp
40074d:       ff d0                   callq  *%rax
40074f:       5d                      pop    %rbp
400750:       e9 7b ff ff ff          jmpq   4006d0 <access@plt+0x70>
400755:       0f 1f 00                nopl   (%rax)
400758:       e9 73 ff ff ff          jmpq   4006d0 <access@plt+0x70>
40075d:       55                      push   %rbp
40075e:       48 89 e5                mov    %rsp,%rbp
400761:       48 81 ec 90 00 00 00    sub    $0x90,%rsp
400768:       bf 98 0c 40 00          mov    $0x400c98,%edi
40076d:       e8 5e fe ff ff          callq  4005d0 <puts@plt>
400772:       48 8b 05 17 19 20 00    mov    0x201917(%rip),%rax        # 602090 <access@plt+0x201a30>
400779:       48 8d 95 70 ff ff ff    lea    -0x90(%rbp),%rdx
400780:       48 89 d6                mov    %rdx,%rsi
400783:       48 89 c7                mov    %rax,%rdi
400786:       e8 25 04 00 00          callq  400bb0 <access@plt+0x550>
40078b:       85 c0                   test   %eax,%eax
40078d:       75 2f                   jne    4007be <access@plt+0x15e>
40078f:       bf d4 0c 40 00          mov    $0x400cd4,%edi
400794:       e8 37 fe ff ff          callq  4005d0 <puts@plt>
400799:       bf f8 0c 40 00          mov    $0x400cf8,%edi
40079e:       e8 2d fe ff ff          callq  4005d0 <puts@plt>
4007a3:       48 8b 05 e6 18 20 00    mov    0x2018e6(%rip),%rax        # 602090 <access@plt+0x201a30>
4007aa:       be 01 00 00 00          mov    $0x1,%esi
4007af:       48 89 c7                mov    %rax,%rdi
4007b2:       e8 a9 fe ff ff          callq  400660 <access@plt>
4007b7:       83 f8 ff                cmp    $0xffffffff,%eax
4007ba:       74 54                   je     400810 <access@plt+0x1b0>
4007bc:       eb 11                   jmp    4007cf <access@plt+0x16f>
4007be:       bf e8 0c 40 00          mov    $0x400ce8,%edi
4007c3:       e8 08 fe ff ff          callq  4005d0 <puts@plt>
4007c8:       b8 01 00 00 00          mov    $0x1,%eax
4007cd:       eb 50                   jmp    40081f <access@plt+0x1bf>
4007cf:       bf 22 0d 40 00          mov    $0x400d22,%edi
4007d4:       e8 f7 fd ff ff          callq  4005d0 <puts@plt>
4007d9:       48 8b 05 b8 18 20 00    mov    0x2018b8(%rip),%rax        # 602098 <access@plt+0x201a38>
4007e0:       48 89 c7                mov    %rax,%rdi
4007e3:       e8 39 00 00 00          callq  400821 <access@plt+0x1c1>
4007e8:       48 89 c2                mov    %rax,%rdx
4007eb:       48 8b 0d c6 18 20 00    mov    0x2018c6(%rip),%rcx        # 6020b8 <access@plt+0x201a58>
4007f2:       48 8b 05 a7 18 20 00    mov    0x2018a7(%rip),%rax        # 6020a0 <access@plt+0x201a40>
4007f9:       48 89 ce                mov    %rcx,%rsi
4007fc:       48 89 c7                mov    %rax,%rdi
4007ff:       b8 00 00 00 00          mov    $0x0,%eax
400804:       e8 f7 fd ff ff          callq  400600 <printf@plt>
400809:       b8 00 00 00 00          mov    $0x0,%eax
40080e:       eb 0f                   jmp    40081f <access@plt+0x1bf>
400810:       bf e8 0c 40 00          mov    $0x400ce8,%edi
400815:       e8 b6 fd ff ff          callq  4005d0 <puts@plt>
40081a:       b8 01 00 00 00          mov    $0x1,%eax
40081f:       c9                      leaveq
400820:       c3                      retq
400821:       55                      push   %rbp
400822:       48 89 e5                mov    %rsp,%rbp
400825:       53                      push   %rbx
400826:       48 81 ec 38 04 00 00    sub    $0x438,%rsp
40082d:       48 89 bd c8 fb ff ff    mov    %rdi,-0x438(%rbp)
400834:       64 48 8b 04 25 28 00    mov    %fs:0x28,%rax
40083b:       00 00
40083d:       48 89 45 e8             mov    %rax,-0x18(%rbp)
400841:       31 c0                   xor    %eax,%eax
400843:       48 8d 85 e0 fb ff ff    lea    -0x420(%rbp),%rax
40084a:       be 00 04 00 00          mov    $0x400,%esi
40084f:       48 89 c7                mov    %rax,%rdi
400852:       e8 f9 fd ff ff          callq  400650 <getlogin_r@plt>
400857:       c7 85 d4 fb ff ff 00    movl   $0x0,-0x42c(%rbp)
40085e:       00 00 00
400861:       eb 3e                   jmp    4008a1 <access@plt+0x241>
400863:       8b 85 d4 fb ff ff       mov    -0x42c(%rbp),%eax
400869:       48 98                   cltq
40086b:       0f b6 94 05 e0 fb ff    movzbl -0x420(%rbp,%rax,1),%edx
400872:       ff
400873:       8b 85 d4 fb ff ff       mov    -0x42c(%rbp),%eax
400879:       48 63 c8                movslq %eax,%rcx
40087c:       48 8b 85 c8 fb ff ff    mov    -0x438(%rbp),%rax
400883:       48 01 c8                add    %rcx,%rax
400886:       0f b6 00                movzbl (%rax),%eax
400889:       31 c2                   xor    %eax,%edx
40088b:       8b 85 d4 fb ff ff       mov    -0x42c(%rbp),%eax
400891:       48 98                   cltq
400893:       88 94 05 e0 fb ff ff    mov    %dl,-0x420(%rbp,%rax,1)
40089a:       83 85 d4 fb ff ff 01    addl   $0x1,-0x42c(%rbp)
4008a1:       8b 85 d4 fb ff ff       mov    -0x42c(%rbp),%eax
4008a7:       48 63 d8                movslq %eax,%rbx
4008aa:       48 8d 85 e0 fb ff ff    lea    -0x420(%rbp),%rax
4008b1:       48 89 c7                mov    %rax,%rdi
4008b4:       e8 27 fd ff ff          callq  4005e0 <strlen@plt>
4008b9:       48 39 c3                cmp    %rax,%rbx
```

```
4008bc:       72 a5                   jb     400863 <access@plt+0x203>
4008be:       8b 85 d4 fb ff ff       mov    -0x42c(%rbp),%eax
4008c4:       48 63 c8                movslq %eax,%rcx
4008c7:       48 8d 95 d8 fb ff ff    lea    -0x428(%rbp),%rdx
4008ce:       48 8d 85 e0 fb ff ff    lea    -0x420(%rbp),%rax
4008d5:       48 89 ce                mov    %rcx,%rsi
4008d8:       48 89 c7                mov    %rax,%rdi
4008db:       e8 1e 00 00 00          callq  4008fe <access@plt+0x29e>
4008e0:       48 8b 75 e8             mov    -0x18(%rbp),%rsi
4008e4:       64 48 33 34 25 28 00    xor    %fs:0x28,%rsi
4008eb:       00 00
4008ed:       74 05                   je     4008f4 <access@plt+0x294>
4008ef:       e8 fc fc ff ff          callq  4005f0 <__stack_chk_fail@plt>
4008f4:       48 81 c4 38 04 00 00    add    $0x438,%rsp
4008fb:       5b                      pop    %rbx
4008fc:       5d                      pop    %rbp
4008fd:       c3                      retq
4008fe:       55                      push   %rbp
4008ff:       48 89 e5                mov    %rsp,%rbp
400902:       48 83 ec 40             sub    $0x40,%rsp
400906:       48 89 7d d8             mov    %rdi,-0x28(%rbp)
40090a:       48 89 75 d0             mov    %rsi,-0x30(%rbp)
40090e:       48 89 55 c8             mov    %rdx,-0x38(%rbp)
400912:       48 8b 45 d0             mov    -0x30(%rbp),%rax
400916:       48 83 c0 02             add    $0x2,%rax
40091a:       48 ba ab aa aa aa aa    movabs $0xaaaaaaaaaaaaaaab,%rdx
400921:       aa aa aa
400924:       48 f7 e2                mul    %rdx
400927:       48 89 d0                mov    %rdx,%rax
40092a:       48 d1 e8                shr    %rax
40092d:       48 8d 14 85 00 00 00    lea    0x0(,%rax,4),%rdx
400934:       00
400935:       48 8b 45 c8             mov    -0x38(%rbp),%rax
400939:       48 89 10                mov    %rdx,(%rax)
40093c:       48 8b 45 c8             mov    -0x38(%rbp),%rax
400940:       48 8b 00                mov    (%rax),%rax
400943:       be 01 00 00 00          mov    $0x1,%esi
400948:       48 89 c7                mov    %rax,%rdi
40094b:       e8 d0 fc ff ff          callq  400620 <calloc@plt>
400950:       48 89 45 f8             mov    %rax,-0x8(%rbp)
400954:       48 83 7d f8 00          cmpq   $0x0,-0x8(%rbp)
400959:       75 0a                   jne    400965 <access@plt+0x305>
40095b:       b8 00 00 00 00          mov    $0x0,%eax
400960:       e9 c0 01 00 00          jmpq   400b25 <access@plt+0x4c5>
400965:       c7 45 e0 00 00 00 00    movl   $0x0,-0x20(%rbp)
40096c:       c7 45 e4 00 00 00 00    movl   $0x0,-0x1c(%rbp)
400973:       e9 3b 01 00 00          jmpq   400ab3 <access@plt+0x453>
400978:       8b 45 e0                mov    -0x20(%rbp),%eax
40097b:       48 98                   cltq
40097d:       48 3b 45 d0             cmp    -0x30(%rbp),%rax
400981:       73 1b                   jae    40099e <access@plt+0x33e>
400983:       8b 45 e0                mov    -0x20(%rbp),%eax
400986:       8d 50 01                lea    0x1(%rax),%edx
400989:       89 55 e0                mov    %edx,-0x20(%rbp)
40098c:       48 63 d0                movslq %eax,%rdx
40098f:       48 8b 45 d8             mov    -0x28(%rbp),%rax
400993:       48 01 d0                add    %rdx,%rax
400996:       0f b6 00                movzbl (%rax),%eax
400999:       0f b6 c0                movzbl %al,%eax
40099c:       eb 05                   jmp    4009a3 <access@plt+0x343>
40099e:       b8 00 00 00 00          mov    $0x0,%eax
4009a3:       89 45 e8                mov    %eax,-0x18(%rbp)
4009a6:       8b 45 e0                mov    -0x20(%rbp),%eax
4009a9:       48 98                   cltq
4009ab:       48 3b 45 d0             cmp    -0x30(%rbp),%rax
4009af:       73 1b                   jae    4009cc <access@plt+0x36c>
4009b1:       8b 45 e0                mov    -0x20(%rbp),%eax
4009b4:       8d 50 01                lea    0x1(%rax),%edx
4009b7:       89 55 e0                mov    %edx,-0x20(%rbp)
4009ba:       48 63 d0                movslq %eax,%rdx
4009bd:       48 8b 45 d8             mov    -0x28(%rbp),%rax
4009c1:       48 01 d0                add    %rdx,%rax
4009c4:       0f b6 00                movzbl (%rax),%eax
4009c7:       0f b6 c0                movzbl %al,%eax
4009ca:       eb 05                   jmp    4009d1 <access@plt+0x371>
4009cc:       b8 00 00 00 00          mov    $0x0,%eax
4009d1:       89 45 ec                mov    %eax,-0x14(%rbp)
4009d4:       8b 45 e0                mov    -0x20(%rbp),%eax
4009d7:       48 98                   cltq
4009d9:       48 3b 45 d0             cmp    -0x30(%rbp),%rax
4009dd:       73 1b                   jae    4009fa <access@plt+0x39a>
4009df:       8b 45 e0                mov    -0x20(%rbp),%eax
4009e2:       8d 50 01                lea    0x1(%rax),%edx
4009e5:       89 55 e0                mov    %edx,-0x20(%rbp)
4009e8:       48 63 d0                movslq %eax,%rdx
4009eb:       48 8b 45 d8             mov    -0x28(%rbp),%rax
4009ef:       48 01 d0                add    %rdx,%rax
4009f2:       0f b6 00                movzbl (%rax),%eax
4009f5:       0f b6 c0                movzbl %al,%eax
4009f8:       eb 05                   jmp    4009ff <access@plt+0x39f>
4009fa:       b8 00 00 00 00          mov    $0x0,%eax
4009ff:       89 45 f0                mov    %eax,-0x10(%rbp)
400a02:       8b 45 e8                mov    -0x18(%rbp),%eax
400a05:       c1 e0 10                shl    $0x10,%eax
400a08:       89 c2                   mov    %eax,%edx
400a0a:       8b 45 ec                mov    -0x14(%rbp),%eax
400a0d:       c1 e0 08                shl    $0x8,%eax
400a10:       01 c2                   add    %eax,%edx
400a12:       8b 45 f0                mov    -0x10(%rbp),%eax
400a15:       01 d0                   add    %edx,%eax
400a17:       89 45 f4                mov    %eax,-0xc(%rbp)
400a1a:       8b 45 e4                mov    -0x1c(%rbp),%eax
400a1d:       8d 50 01                lea    0x1(%rax),%edx
400a20:       89 55 e4                mov    %edx,-0x1c(%rbp)
400a23:       48 63 d0                movslq %eax,%rdx
400a26:       48 8b 45 f8             mov    -0x8(%rbp),%rax
400a2a:       48 01 c2                add    %rax,%rdx
400a2d:       8b 45 f4                mov    -0xc(%rbp),%eax
400a30:       c1 e8 12                shr    $0x12,%eax
400a33:       83 e0 3f                and    $0x3f,%eax
400a36:       89 c0                   mov    %eax,%eax
400a38:       0f b6 80 c0 20 60 00    movzbl 0x6020c0(%rax),%eax
400a3f:       88 02                   mov    %al,(%rdx)
400a41:       8b 45 e4                mov    -0x1c(%rbp),%eax
400a44:       8d 50 01                lea    0x1(%rax),%edx
400a47:       89 55 e4                mov    %edx,-0x1c(%rbp)
400a4a:       48 63 d0                movslq %eax,%rdx
400a4d:       48 8b 45 f8             mov    -0x8(%rbp),%rax
400a51:       48 01 c2                add    %rax,%rdx
400a54:       8b 45 f4                mov    -0xc(%rbp),%eax
400a57:       c1 e8 0c                shr    $0xc,%eax
400a5a:       83 e0 3f                and    $0x3f,%eax
400a5d:       89 c0                   mov    %eax,%eax
400a5f:       0f b6 80 c0 20 60 00    movzbl 0x6020c0(%rax),%eax
400a66:       88 02                   mov    %al,(%rdx)
400a68:       8b 45 e4                mov    -0x1c(%rbp),%eax
400a6b:       8d 50 01                lea    0x1(%rax),%edx
400a6e:       89 55 e4                mov    %edx,-0x1c(%rbp)
400a71:       48 63 d0                movslq %eax,%rdx
400a74:       48 8b 45 f8             mov    -0x8(%rbp),%rax
400a78:       48 01 c2                add    %rax,%rdx
400a7b:       8b 45 f4                mov    -0xc(%rbp),%eax
400a7e:       c1 e8 06                shr    $0x6,%eax
400a81:       83 e0 3f                and    $0x3f,%eax
400a84:       89 c0                   mov    %eax,%eax
```

```
  400a86:       0f b6 80 c0 20 60 00    movzbl 0x6020c0(%rax),%eax
  400a8d:       88 02                   mov    %al,(%rdx)
  400a8f:       8b 45 e4                mov    -0x1c(%rbp),%eax
  400a92:       8d 50 01                lea    0x1(%rax),%edx
  400a95:       89 55 e4                mov    %edx,-0x1c(%rbp)
  400a98:       48 63 d0                movslq %eax,%rdx
  400a9b:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400a9f:       48 01 c2                add    %rax,%rdx
  400aa2:       8b 45 f4                mov    -0xc(%rbp),%eax
  400aa5:       83 e0 3f                and    $0x3f,%eax
  400aa8:       89 c0                   mov    %eax,%eax
  400aaa:       0f b6 80 c0 20 60 00    movzbl 0x6020c0(%rax),%eax
  400ab1:       88 02                   mov    %al,(%rdx)
  400ab3:       8b 45 e0                mov    -0x20(%rbp),%eax
  400ab6:       48 98                   cltq
  400ab8:       48 3b 45 d0             cmp    -0x30(%rbp),%rax
  400abc:       0f 82 b6 fe ff ff       jb     400978 <access@plt+0x318>
  400ac2:       c7 45 e0 00 00 00 00    movl   $0x0,-0x20(%rbp)
  400ac9:       eb 24                   jmp    400aef <access@plt+0x48f>
  400acb:       48 8b 45 c8             mov    -0x38(%rbp),%rax
  400acf:       48 8b 10                mov    (%rax),%rdx
  400ad2:       8b 45 e0                mov    -0x20(%rbp),%eax
  400ad5:       48 98                   cltq
  400ad7:       48 29 c2                sub    %rax,%rdx
  400ada:       48 89 d0                mov    %rdx,%rax
  400add:       48 8d 50 ff             lea    -0x1(%rax),%rdx
  400ae1:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400ae5:       48 01 d0                add    %rdx,%rax
  400ae8:       c6 00 3d                movb   $0x3d,(%rax)
  400aeb:       83 45 e0 01             addl   $0x1,-0x20(%rbp)
  400aef:       48 8b 4d d0             mov    -0x30(%rbp),%rcx
  400af3:       48 ba ab aa aa aa aa aa  movabs $0xaaaaaaaaaaaaaaab,%rdx
  400afa:       aa aa aa
  400afd:       48 89 c8                mov    %rcx,%rax
  400b00:       48 f7 e2                mul    %rdx
  400b03:       48 d1 ea                shr    %rdx
  400b06:       48 89 d0                mov    %rdx,%rax
  400b09:       48 01 c0                add    %rax,%rax
  400b0c:       48 01 d0                add    %rdx,%rax
  400b0f:       48 29 c1                sub    %rax,%rcx
  400b12:       48 89 ca                mov    %rcx,%rdx
  400b15:       8b 04 95 00 21 60 00    mov    0x602100(,%rdx,4),%eax
  400b1c:       3b 45 e0                cmp    -0x20(%rbp),%eax
  400b1f:       7f aa                   jg     400acb <access@plt+0x46b>
  400b21:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400b25:       c9                      leaveq
  400b26:       c3                      retq
  400b27:       66 0f 1f 84 00 00 00    nopw   0x0(%rax,%rax,1)
  400b2e:       00 00
  400b30:       41 57                   push   %r15
  400b32:       41 89 ff                mov    %edi,%r15d
  400b35:       41 56                   push   %r14
  400b37:       49 89 f6                mov    %rsi,%r14
  400b3a:       41 55                   push   %r13
  400b3c:       49 89 d5                mov    %rdx,%r13
  400b3f:       41 54                   push   %r12
  400b41:       4c 8d 25 c8 12 20 00    lea    0x2012c8(%rip),%r12        # 601e10 <access@plt+0x2017b0>
  400b48:       55                      push   %rbp
  400b49:       48 8d 2d c8 12 20 00    lea    0x2012c8(%rip),%rbp        # 601e18 <access@plt+0x2017b8>
  400b50:       53                      push   %rbx
  400b51:       4c 29 e5                sub    %r12,%rbp
  400b54:       31 db                   xor    %ebx,%ebx
  400b56:       48 c1 fd 03             sar    $0x3,%rbp
  400b5a:       48 83 ec 08             sub    $0x8,%rsp
  400b5e:       e8 35 fa ff ff          callq  400598 <puts@plt-0x38>
  400b63:       48 85 ed                test   %rbp,%rbp
  400b66:       74 1e                   je     400b86 <access@plt+0x526>
  400b68:       0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  400b6f:       00
  400b70:       4c 89 ea                mov    %r13,%rdx
  400b73:       4c 89 f6                mov    %r14,%rsi
  400b76:       44 89 ff                mov    %r15d,%edi
  400b79:       41 ff 14 dc             callq  *(%r12,%rbx,8)
  400b7d:       48 83 c3 01             add    $0x1,%rbx
  400b81:       48 39 eb                cmp    %rbp,%rbx
  400b84:       75 ea                   jne    400b70 <access@plt+0x510>
  400b86:       48 83 c4 08             add    $0x8,%rsp
  400b8a:       5b                      pop    %rbx
  400b8b:       5d                      pop    %rbp
  400b8c:       41 5c                   pop    %r12
  400b8e:       41 5d                   pop    %r13
  400b90:       41 5e                   pop    %r14
  400b92:       41 5f                   pop    %r15
  400b94:       c3                      retq
  400b95:       66 66 2e 0f 1f 84 00    data32 nopw %cs:0x0(%rax,%rax,1)
  400b9c:       00 00 00 00
  400ba0:       f3 c3                   repz retq
  400ba2:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  400ba9:       00 00 00
  400bac:       0f 1f 40 00             nopl   0x0(%rax)
  400bb0:       48 89 f2                mov    %rsi,%rdx
  400bb3:       48 89 fe                mov    %rdi,%rsi
  400bb6:       bf 01 00 00 00          mov    $0x1,%edi
  400bbb:       e9 80 fa ff ff          jmpq   400640 <__xstat@plt>

Disassembly of section .fini:

0000000000400bc0 <.fini>:
  400bc0:       48 83 ec 08             sub    $0x8,%rsp
  400bc4:       48 83 c4 08             add    $0x8,%rsp
  400bc8:       c3                      retq
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
check failed.
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ chmod 700 iamspecial
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./3
checking for the existence of a file with a special name...
correct file found.
checking for appropriate access rights...
correct rights found.
you win! the secret is:
dkNVQVMPDQ==
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ls
0  1  2  3  4  howto.txt  iamspecial  secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi howto.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ ./4
./4: error while loading shared libraries: lib361.so: cannot open shared object file: No such file or directory
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$ vi secrets.txt
Dpate85@Dhrumil:~/dpate85/hw2/puzzles$
```