
Phase 1
BOOM!

Set up breakpoints

(gdb) break [location]

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ gdb bomb
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 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 bomb...done.
(gdb) break *phase_1
Breakpoint 1 at 0x400edd
(gdb) break *explode_bomb
Breakpoint 2 at 0x401606
```

Now lets run

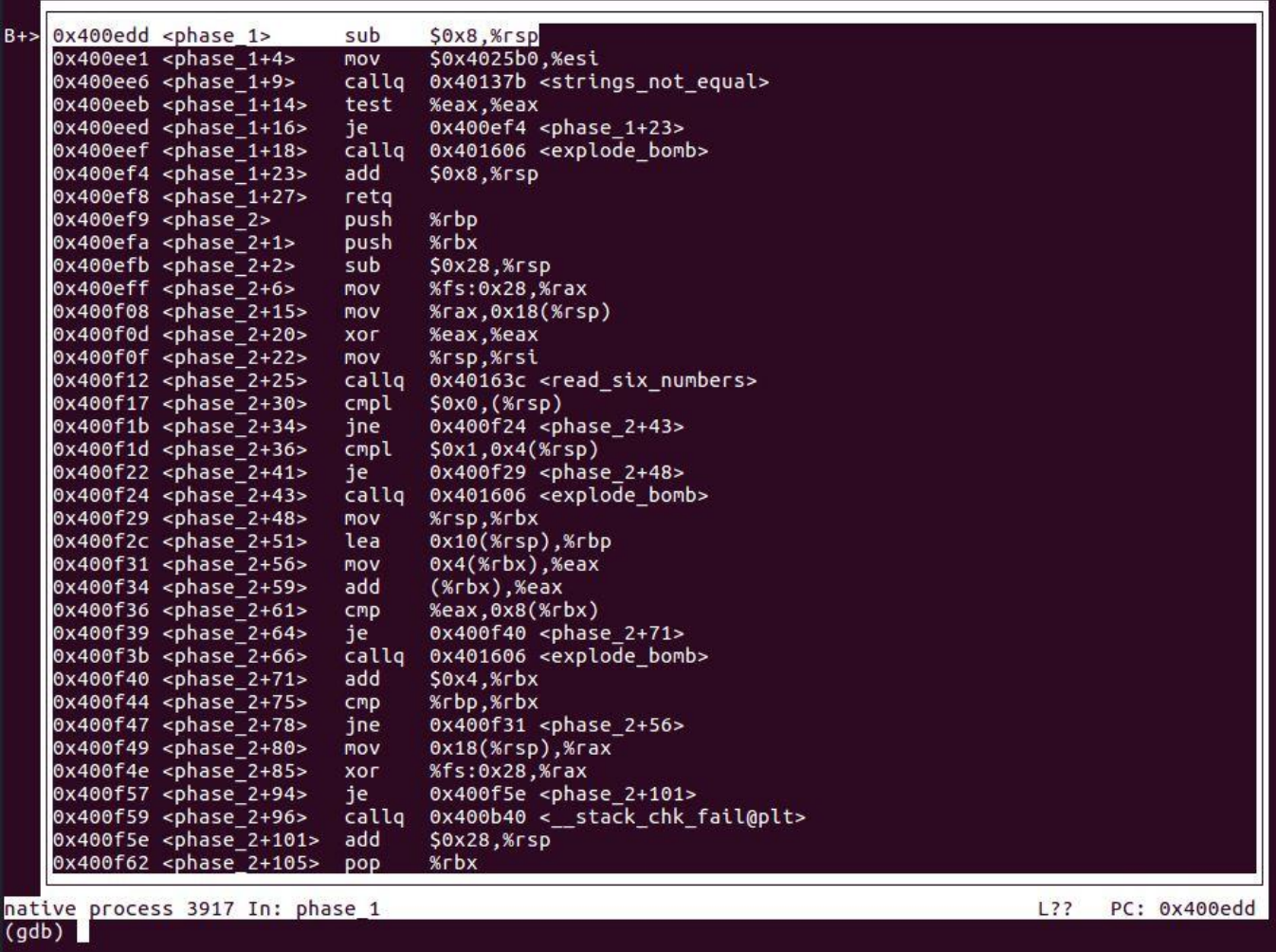
(gdb) run

```
(gdb) r
Starting program: /home/philip_nelson/Documents/Architecture/bomb3/bomb
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
```

Lets give the bomb a “test string”

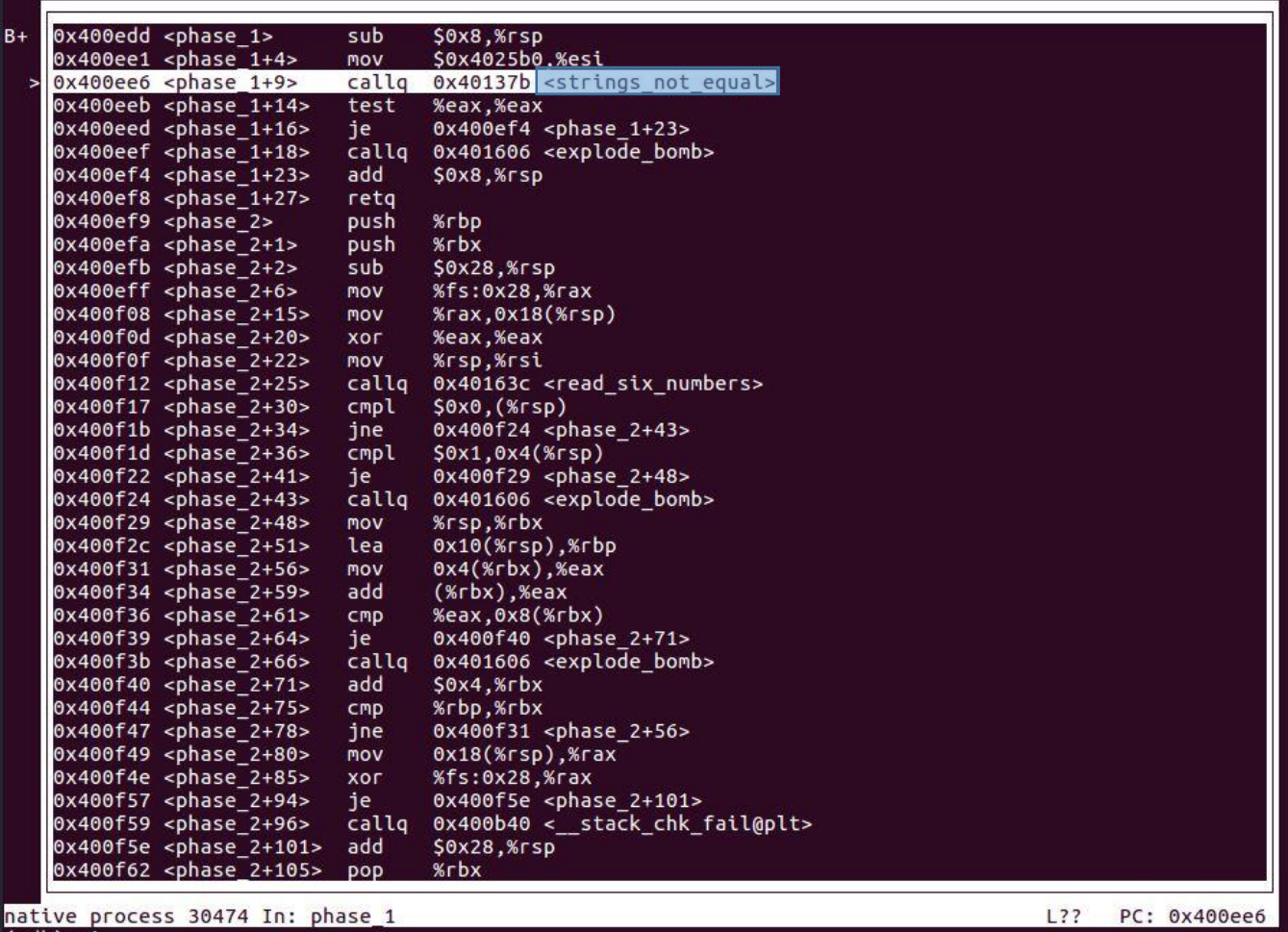
```
(gdb) r
Starting program: /home/philip_nelson/Documents/Architecture/bomb3/bomb
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
test string

Breakpoint 1, 0x0000000000400edd in phase_1 ()
(gdb)
```

And we can see GDB broke on our first breakpoint: phase_1



To step through the assembly, let's enter assembly mode

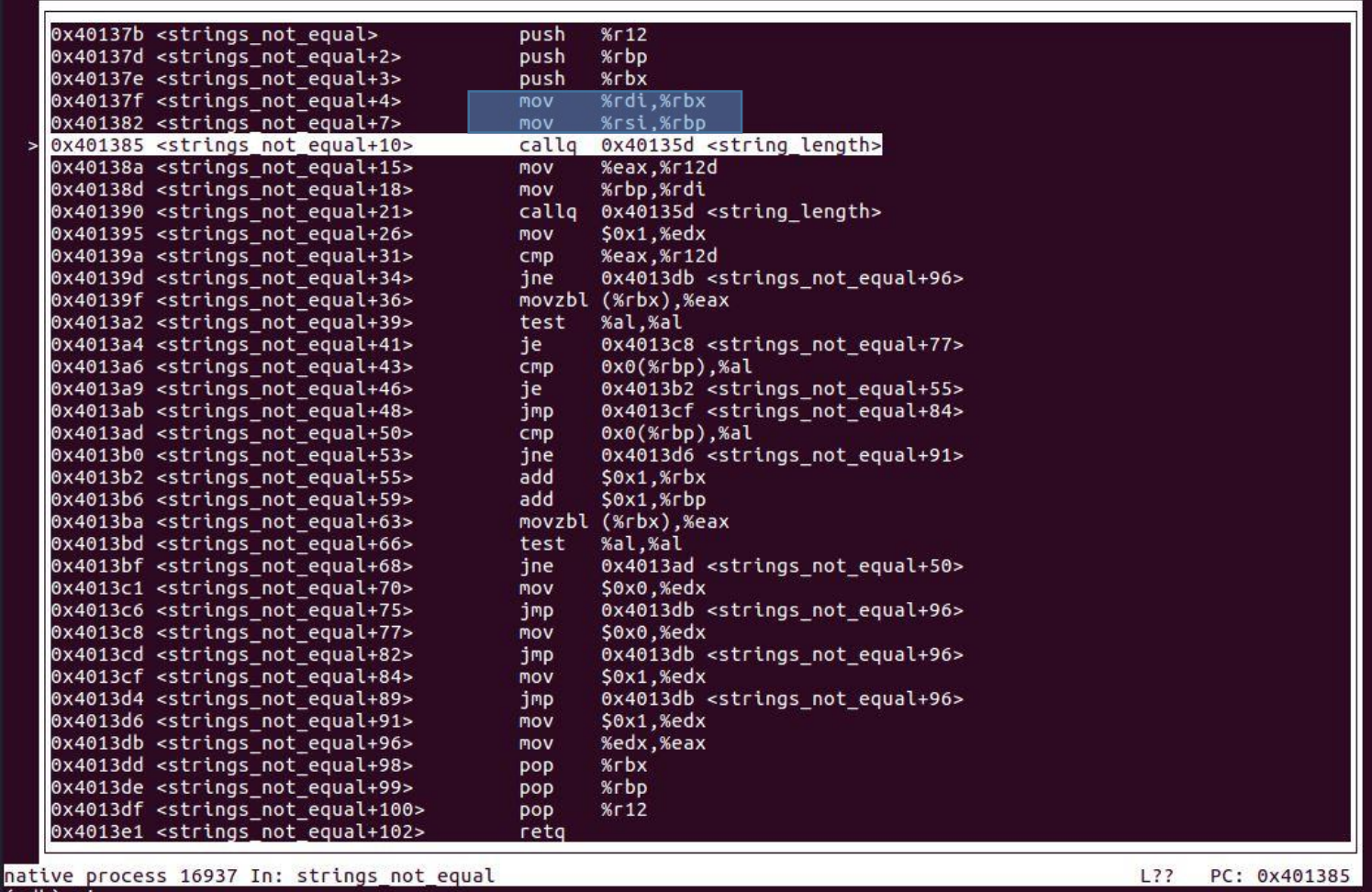
(gdb) layout asm



Now let’s step through some instructions

(gdb) stepi

look, a function called strings_not_equal



Some more steps later, look, a function called string_length!

It might compare the length of our “test string” to the answer

Let’s look at what is in those registers that were set right before the function call

The first is $rbx

(gdb) p/x $rbx

This gives us the memory address stored there

```
(gdb) p/x $rbx
$2 = 0x604bc0
(gdb) x /25c 0x604bc0
0x604bc0 <input_strings>:       116 't' 101 'e' 115 's' 116 't' 32 ' '  115 's' 116 't' 114 'r'
0x604bc8 <input_strings+8>:     105 'i' 110 'n' 103 'g' 0 '\000'        0 '\000'        0 '\000'
0x604bd0 <input_strings+16>:    0 '\000'        0 '\000'        0 '\000'        0 '\000'
0x604bd8 <input_strings+24>:    0 '\000'
(gdb)
```

(gdb) x /25c 0x604bc0

This prints the first 25 bytes starting at the specified address as chars

‘t’’e’’s’’t’’ “s’’t’’r’’i’’n’’g’

The second is $rbp

(gdb) p/x $rbp

This gives us the memory address stored there

```
(gdb) p/x $rbp
$3 = 0x4025b0
(gdb) x /25c 0x4025b0
0x4025b0:       89 'Y'  111 'o' 117 'u' 32 ' '  99 'c'  97 'a'  110 'n' 32 ' '
0x4025b8:       82 'R'  117 'u' 115 's' 115 's' 105 'i' 97 'a'  32 ' '  102 'f'
0x4025c0:       114 'r' 111 'o' 109 'm' 32 ' '  108 'l' 97 'a'  110 'n' 100 'd'
0x4025c8:       32 ' '
(gdb)
```

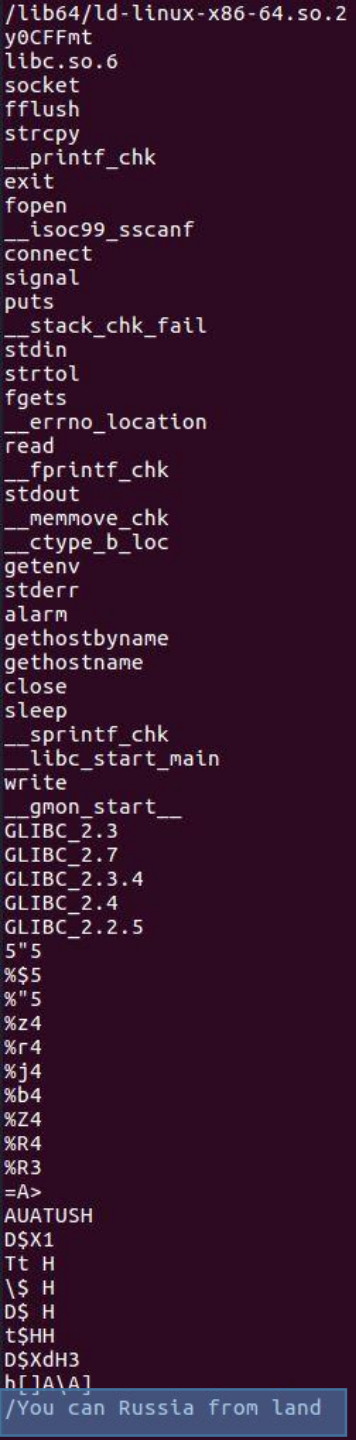
(gdb) x /25c 4025b0

This prints the first 25 bytes starting at the specified address as chars

‘Y’’o’’u’’ ‘’c’’a’’n’’ ‘’R’’u’’s’’s’’i’’a’’ ‘’f’’r’’o’’m’’ ‘’l’’a’’n’’d’

Lets see if we can find this in the strings

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ strings bomb | less
```



With ‘/’ we can search for our partial string

/You can Russia from land

```
You can Russia from land here in Alaska.
Wow! You've defused the secret stage!
So you think you can stop the bomb with ctrl-c, do you?
Initialization error: Running on an illegal host [1]
ERROR: Input string is too large.
Your instructor has been notified.
Curses, you've found the secret phase!
But finding it and solving it are quite different...
Congratulations! You've defused the bomb!
Your instructor has been notified and will verify your solution.
Well...
OK. :-)
Invalid phase%s
Initialization error:
defused
exploded
%d:%s:%d:%s
BOOM!!!
The bomb has blown up.
%d %d %d %d %d %d
Error: Premature EOF on stdin
GRADE_BOMB
Error: Input line too long
%d %d %s
DrEvil
greatwhite.ics.cs.cmu.edu
angelshark.ics.cs.cmu.edu
makoshark.ics.cs.cmu.edu
Program timed out after %d seconds
Error: HTTP request failed with error %d: %s
GET /%s/submitr.pl/?userid=%s&lab=%s&result=%s&submit=submit HTTP/1.0
Error: Unable to connect to server %s
%%%02X
%s %d %[a-zA-z ]
pine.cs.usu.edu
AUTORESULT_STRING=%s
USUspr17
;*3$"
philip_nelson
GCC: (Ubuntu 5.4.0-6ubuntu1~16.04.4) 5.4.0 20160609
<Welcome to my fiendish little bomb. You have 6 phases with
-which to blow yourself up. Have a nice day!
*Phase 1 defused. How about the next one?
That's number 2.  Keep going!
Halfway there!
%So you got that one.  Try this one.
Good work!  On to the next...
/usr/include/x86_64-linux-gnu/bits
/usr/lib/gcc/x86_64-linux-gnu/5/include
/usr/include
bomb.c
stdio2.h
stddef.h
types.h
stdio.h
libio.h
stdlib.h
support.h
:
```

Check that out, we found it!

The whole string is “You can Russia from land here in Alaska.”

Back to gdb
Let’s source our breakpoints

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ gdb bomb
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 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 bomb...done.
(gdb) source source.txt
Breakpoint 1 at 0x401606
Breakpoint 2 at 0x400edd
(gdb)
```

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ gdb bomb
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 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 bomb...done.
(gdb) source source.txt
Breakpoint 1 at 0x401606
Breakpoint 2 at 0x400edd
(gdb) r
Starting program: /home/philip_nelson/Documents/Architecture/bomb3/bomb
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
You can Russia from land in Alaska
```

And check out that string

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ gdb
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/
This is free software: you are free to change and redistrib
There is NO WARRANTY, to the extent permitted by law.  Type
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 onlin
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word
Reading symbols from bomb...done.
(gdb) source source.txt
Breakpoint 1 at 0x401606
Breakpoint 2 at 0x400edd
(gdb) r
Starting program: /home/philip_nelson/Documents/Architectur
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
You can Russia from land here in Alaska

Breakpoint 2, 0x0000000000400edd in phase_1 ()
(gdb) c
Continuing.

Breakpoint 1, 0x0000000000401606 in explode_bomb ()
(gdb)
```

WOAH! Looks like we typed in the string incorrectly.

Good thing we have a breakpoint set before the bomb explodes

```
philip_nelson@virtualbox:~/Documents/Architecture/bomb3$ gd
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/
This is free software: you are free to change and redistrib
There is NO WARRANTY, to the extent permitted by law.  Type
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 onlin
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word
Reading symbols from bomb...done.
(gdb) source source.txt
Breakpoint 1 at 0x401606
Breakpoint 2 at 0x400edd
(gdb) r
Starting program: /home/philip_nelson/Documents/Architectur
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
You can Russia from land here in Alaska.

Breakpoint 2, 0x0000000000400edd in phase_1 ()
(gdb) c
Continuing.
Phase 1 defused. How about the next one?
```

With the string typed in correctly this time we have the first phase defused