

Forum:[Hack Forums](#)

Thread: [Checking Values on Stack](#)

7_R3X - July 27, 2017, 12:00 pm

I want to print 1 words from the top of stack in the form of hexadecimal. To do so, I typed the following:

Code (gdb) x/1xw \$espbut GDB keeps popping up:

Code 0xffffffffffe030: Cannot access memory at address 0xffffffffffe030

The program I'm trying to debug has already pushed a value onto stack so just in case if you're wondering that I might be trying to access kernel variables at the very beginning of program, it's not so. Any idea?

Mischa - July 27, 2017, 2:08 pm

I have no idea what architecture you're on.

What OS you're on.

You didn't paste any code. Clearly the problem is the ESP register because GDB is accessing invalid memory when it tries to read ESP.

At the start of your program, check ESP and EBP. Then push onto the stack and check ESP again.

You're tampering with ESP somewhere in your code, there's no other reason why it would holding an invalid memory address.

7_R3X - July 28, 2017, 6:00 am

(Jul 27, 2017 02:08 PM)Mischa Wrote: [\[CITE\]](#) I have no idea what architecture you're on.

What OS you're on.

You didn't paste any code. Clearly the problem is the ESP register because GDB is accessing invalid memory when it tries to read ESP.

At the start of your program, check ESP and EBP. Then push onto the stack and check ESP again.

You're tampering with ESP somewhere in your code, there's no other reason why it would holding an invalid memory address.

I found the reason why I had problem accessing the address space. 0xffffffffffe030 is actually a 64 bit address and I tried to access it using esp which is a 32 bit register. I changed

Code (gdb) x/1xw \$espto

Code (gdb) x/1xw \$rspand every thing works fine now!!