
Subject: howto see assembly code? [SOLVED]
Posted by [fudadmin](#) on Mon, 21 Nov 2005 17:27:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

howto see assembly code?

Subject: Re: howto see assembly code?
Posted by [mirek](#) on Mon, 21 Nov 2005 17:44:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

It is always showed when debugging in the right pane. Just place breakpoint and run the app.

BTW, there is a problem with viewing release code assembly (useful when doing optimizations). I tend to solve this by placing `__BREAK__`, which is essentially `*(int *)0 = 0`; to crash the code at the right place and invoke debugger

Subject: Re: howto see assembly code?
Posted by [mr_ped](#) on Tue, 22 Nov 2005 21:41:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Mon, 21 November 2005 18:44*`*(int *)0 = 0`; to crash the code at the right place and invoke debugger

`__asm int 3;`

(MS syntax)

will probably work a tad better in such case.

(legal "hard breakpoint" without testing the stability of system's core memory protection)

Subject: Re: howto see assembly code?
Posted by [mirek](#) on Tue, 22 Nov 2005 22:22:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Yes, in fact that is equivalent. Does not work on linux thought.

Subject: Re: howto see assembly code?
Posted by [mr_ped](#) on Wed, 23 Nov 2005 00:06:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

hm... calling breakpoint interrupt and rewriting memory at address 0 with 0 is really not the equivalent. INT 3 summons breakpoint interrupt in DOS (Win16/32), and after resume the thread

will continue, while `*(int *)0 = 0;` does rewrite protected memory, so the thread will crash. Maybe minor difference when you want to check the code, but can be helpful when you are just stepping through some code.

I wonder why linux does not implement (or does it?) "INT 3" as breakpoint, because AFAIK (at least in the age of 286/386/486 CPUs) int 3 is the only 1B long interrupt instruction, and the main purpose for this instruction was exactly ability to use breakpoints easily in debuggers. (at least the code should *not* crash at linux, as the INT 3 should be simple RET interrupt when no debugger is running)

Subject: Re: howto see assembly code?

Posted by [mirek](#) on Wed, 23 Nov 2005 06:12:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Well, actually I think Linux uses int 3 as well, I was rather referring to syntax (it would need `#ifdef` because of assembly).

Anyway, I do not think this is the most important thing in the world. Thankfully, all systems are able to deal with NULL assignment well as this is just a tool for experimenting with the code, nothing that should be left in production app...
