

## Re: Reverse engineering != piracy (was Re: RosAsm disassembler output vs. IDA Pro)

*Source:* <http://coding.derkeiler.com/Archive/Assembler/alt.lang.asm/2004-01/1354.html>

---

*From:* Gerhard W. Gruber (*sparhawk\_at\_gmx.at*)

*Date:* 01/28/04

Date: Wed, 28 Jan 2004 21:48:36 +0100

On 27 Jan 2004 23:30:11 GMT wrote Betov <betov@free.fr> in alt.lang.asm with <XnF947E8C2944BBbetovfreefr@213.228.0.138>

>*I have always said that the first purpose of this  
>Tool is for \_Translation\_. In that area, Assembly  
>Language has a \_huge\_ advantage on all other ones:*

I always understood you that way, because it would be really ridiculous if you would believe otherwise.

>*Assembly is the universal nodal of all languages.*

Yes. :) We agree on that.

>*So, facing the lack of Asm Demos and the pain the  
>guys wanting to switch to Asm may have with not  
>recovering their previous works, all i can say is  
>that i am rather estonished, that nobody did it  
>before.*

It's not really astonishing to me. I learned first a tiny little BASIC (back in those days :) ) on the C64, but because it was so lacking in speed, and it couldn't do some of the cool things like really smooth scrolling or rasterline interrupt and other stuff, I soon learned assembly. Actually by accident, because I read some BASIC programming book, which had a few lines assembler included to speed up a certain part. Fortunately the code was so good documented, that I immediately understood the mnemonics and how to use them, so I played around with them and got hooked. From then on I exclusively programmed in ASM, directly into memory usually, without so fancy things like symbols. Back in those days I knew every address of my computer by its forename. :) Of course, when I started to learn C I also used assembly to debug my code and see what it actually was doing and of course in those days there were no such things like sourcelevel debugger. This was a great leap forward, and even after the first compilers were available with sourcelevel debugging, I still used mainly assembly for debugging (on the Amiga then).

>*From this I learned quite a lot of how the compiler translates code into ASM,*

alt.lang.asm: Re: Reverse engineering != piracy (was Re: RosAsm disassembler output vs. IDA Pro)

without ever reading a manual on compiler technology. So with this background I think it is understandable that I could embrace your approach. But I don't believe that anybody, who starts learning assembly coming from a HLL first, will benefit from a disassembler. If he disassembles he previous HLL code into RosAsm Syntax then he has HLL code in RosAsm syntax. Not really helpfull for a beginner. If he just translates his exsiting ASM code into RosAsm syntax with this, then he has gained some speed on the translation process, but because of some inherent problems with diassembling, I question the usefullnuss of such a code.

As I already stated: Even if you disassembler will work perfectly, and could disassmble each and every piece of executable or DLL or library, you throw at it, and it could determine what is code and what is data, there is still an inherent flaw with this. You can NOT disassemble equates, which will make the listing readable and usable for a human reader, not to mention maintainable. AND you can NOT disassemble structures properly. It doesn't matter wether you support structures by syntax or not. Even if you consider a structure simply as a set of equates with specifying the offset for each field (basically a structure does just this) then you can NOT determine this structure by diassembling.

*>[Please, Gerhard, there are other interesting  
>things to discuss about, without falling into  
>Master Pdf usual Tips&Tricks&Traps... Discussing  
>about the possibility of any legal problem does  
>nothing else than suggesting that there is really  
>some possibility of a legal problem. Enough.]*

Well. I don't mind. I don't buy this that you intend to do this just for enhancing piracy anyway, so there is no point in discussin this particular point endlessly. :)

--

Gerhard Gruber  
Maintainer of  
SoftICE for Linux - <http://pice.sourceforge.net/>  
Fast application launcher - <http://sourceforge.net/projects/launchmenu>