

Re: Copying a kernel routine

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2008-01/msg00003.html>

- *From:* "Alexander Grigoriev" <alegr@xxxxxxxxxxxxxx>
 - *Date:* Tue, 1 Jan 2008 07:42:55 -0800
-

Again,

moving an arbitrary code to different location is BAD idea.

"Hummingbird" <Hummingbird@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message <news:DC1E26FF-936F-4147-B0EC-6DDC099FC312@xxxxxxxxxxxxxxxxxxxx>

That's a good idea. Thanks a lot.
I can create a dummy function like

```
_asm
{
nop;
nop;
nop;
...
}
```

and copy the kernel routine into this address.
I'll try it. Thanks again.

"Vetzak" wrote:

Bad idea. Stay off the CR0 register or any other processor state. The OS manages this state.

I would try this method: Extend the size of the .text segment in your driver .sys file. The .text segment is marked as executable by the OS. Once loaded into memory, you can use the extra, unused bytes and fill them up with your own subroutine(s).

You may want to write such routines in assembler. Relocations: you can extract them from the .obj file, or write assembler code that does not depend on fixed addresses.

On Dec 29, 11:41 pm, Hummingbird
<Hummingb...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

Re: Copying a kernel routine

Hi, Thanks for respond.

Well, i am writing a anti-malware software actually. As you know, they use hooks and modify the Windows kernel to hide and protect themselves.

You are right, Relocation is a problem. I can use something like LdrRelocateImageWithBias (I mean write another one myself since it's not exported), but that's means I have to copy the whole ntoskrnl.exe to the pool.

About the execute bit, I guess we can set it manually since we are in Ring 0, just like the CR0 register. I don't know if i am right. But maybe that's not recommended by Microsoft.

"Don Burn" wrote:

Bad idea in general, firs