

Changing Interrupt Vector Table (Create your own interrupt!)

Author: InternetNightmare

Another thing I want to write tutorial is about changing interrupts. There are two ways you can do that using DOS interrupts and modifying interrupt vector table directly. Both ways are pretty simple, you need to know these DOS interrupts:

Function	What does it do?	Parameters
AH = 25h	Set interrupt vector	AL – interrupt number to change DS:DX – pointer to interrupt function
AH = 35h	Get interrupt vector. Gets address of currently set interrupt.	AL – interrupt number <u>Returns:</u> ES:BX – pointer to interrupt
AH = 4Ch	Exits DOS program ;)	AL – exit code (not sure what it does)

It's pretty simple, just take a look at the sample code [here](#).

The other way to make your own interrupt is to modify interrupt vector table directly. It's mapped from 0000:0000 to 0000:0400h in memory. The structure is very simple:

	Offset	Segment
Int 0	(Offset 0000)	(Offset 0002)
Int 1	(Offset 0004)	(Offset 0006)
Int 2	(Offset 0008)	(Offset 0010)

So getting interrupt offset is:

```
mov ax, [intnum*4]
```

And segment:

```
mov ax [intnum*4+2]
```

Setting:

```
mov ax, [intnum*4] ; offset  
mov ax [intnum*4+2] ; segment
```

Well and how to call the interrupt, I think we all know:

```
int intnum
```

Everything is pretty simple. NASM source code:

DOS interrupt version – [here](#)

Direct modifying of intvec table - [here](#)

And you can always e-mail me at: internetnightmare_x@yahoo.com