# Cracking Amecisco's "Invisible Key-Logger 97" (available at www.amecisco.com, Ik97v12s.exe)

by: Red_Star (aka Xander)

(Written exclusively for Factor members)

All right kids, this program can be very useful tool for you hackers out there who have access to IBM stations that people telnet from (i.e. schools). This is the only Key-Logger to my knowledge that has no icons, either in the system tray or menu bar when running. It's only noticable in the "Close Program" dialog box (CTRL-ALT-DEL) as "Ik" so it even looks normal. But that's not it! It actually "encrypts" the log file, so only you (or anyone else with the program) can decode it! Amecisco wants $29.99 for the full product, which will start recording before windows asks for a username and password (shareware version wait's till after..) and it will only record the first 500 key strokes. I really don't care about the windows login, but the limit of 500 characters was a pain to me, so that's what this crack gets rid of. If you want the windows logon information recorded, feel free to crack that yourself ;) Ok! Well, lets get started shall we!

For this little "project" we'll need the following tools –

W32Dasm89
HexWorkshop 32 2.53 ( or any hex editor)

Now you may be saying, where's SoftIce? Why don't you use SoftIce? Well, no, and this program is difficult to use SoftIce with so it's 100 times much easier not to use it. This is why:
The program (ik.exe) once disassembled and when you look under Functions>Imports will reveal the following:

IK.InstallHook
Kernel32.GetModuleHandleA
Kernel32.GetStartupInfoA
MFC42.MFC42:NoName00
MFC42.MFC42:NoName01
MFC42.MFC42:NoName02
MFC42.MFC42:NoName03
……..
MFC42.MFC42:NoName82
MSVCRT.__CxxFrameHandler
… blah blah

Anyway, basically there's nothing that you can set a breakpoint on in SoftIce, mainly because it runs in the background. No "real" user input, no messages to the user, etc.. All it does is read a key, encrypt it and save it to the disk. So without any breakpoints, no SoftIce! So since you've got IK.exe disassembled, let's check out the code. Now we know IK has a max of 500 characters for shareware, so lets look for anything about 500 or subtracting to 1 or adding 1 499 times…. Well, 500 in hex is 01F4 so lets search for that! Search, search, search and what do you get? Nothing, god damnit! Ok, so lets look for and inc or dec instructions, or anything else that adds 1 or subtracts one. Well, we get a few cases, lets look at a few.

```
:00401707 46      inc esi
:00401708 89758C  mov dword ptr [ebp-74], esi
:0040170B 8A06    mov al, byte ptr [esi]
:0040170D 84C0    test al, al
:0040170F 7404    je 00401715
:00401711 3C22    cmp al, 22
:00401713 75F2    jne 00401707
```

What does this do? I really don't know, something to do with the little encryption process that it saves the file with I guess. But you can spend a few days (like I did) going crazy, NOPing these bad boys out till it hits you that MAYBE the damn check isn't in the .exe file!
So lets see the import list again. Look at the first one, Ik.Install.Hook. That's in Ik.dll (you can see that from quickviewing the file (right click on icon, choose quikview). So lets load up Ik.DLL! Disassemble, Disassemble…. Ok! Now, check out the Exports! What do we have? InstallHook and KeyboardHook! Damn! Could be close! Double click on Keyboard.Hook and it takes us to what we see below…

```
Exported fn(): KeyboardHook - Ord:0002h
:11F0 53                push ebx
:11F1 8B5C240C          mov ebx, dword ptr [esp+0C]
:11F5 56                push esi
:11F6 8B74240C          mov esi, dword ptr [esp+0C]
:11FA 85F6              test esi, esi
:11FC 7C79              jl 10001277           ;keep looking...
:11FE 83FE03            cmp esi, 00000003
:1201 7474              je 10001277
:1203 A108C00010        mov eax, dword ptr [1000C008]
:1208 8ACB              mov cl, bl
:120A 80F102            xor cl, 02
:120D C705B0C0001000000000    mov dword ptr [1000C0B0], 00000000
:1217 888810C00010      mov byte ptr [eax+1000C010], cl
:121D 8B0DB4C00010      mov ecx, dword ptr [1000C0B4]
:1223 40                inc eax    ;←-----
:1224 41                inc ecx    ;←----- Looky!
:1225 81F9E8030000      cmp ecx, 000003E8
```

Well I'll be damned. Check those puppies out. After some screwing around, you'll find out that the first counter (1223 inc eax), actually just counts the key strokes up to 75. If you read the readme file all the way through (always read your docs!), you'll know it reads 75 characters into a buffer and then writes those 75 all at once. So, here's a mistake that could throw you off. Don't NOP out the inc eax because the buffer will never reach 75, and it will never write anything to a file! Oops! So the next one is the one we want to get rid of, just NOP out the inc ecx and that's the "real" counter. Ok, for you real newbies, here's precise instructions..
Load up the hex editor, search for the string "10 40 41 81 F9" and replace it with "10 40 90 81 F9". Save the file and reload IK.exe and you set! And remember, when you don't have anything to set a BPX on, you're still ok, don't become too dependent on 1 tool. Learn all the aspects! You see the power of one byte! That does it, good luck and happy cracking!

Greets: Razzia, ED!SON, ACP, fravia+

Questions, comments, send 'em on over to: xander500@hotmail.com

P.S. I have included ikpatch.exe, which will patch the file for you (as long as it's in the same directory) in case you really have problems. If you want the .asm source for the patch, email me at the address above.