How to crack

Written by : +VipVop and Published by Splatter Industries,Inc.

Using W32Dasm and SoftIce 3.2 on Psplit97

Ok, First of all I tried using softice on this program, but since it was a Delphi program it didn't use any of the standard calls, so I decided to use w32dasm, however softice did play a later roll ,although it could have been cracked without it. So So anyway, open Psplit97.exe in W32dasm. Now click on "Refs" (to the left of help) and choose the string data references. You will see a list of all the strings contained in the string. Now we want to know what make the program put the unregistered in the title so we double click on the Psplit97 (unregistered). W32dasm will stop at the following code:

```
*Possible StringData Ref from Code Obj ->"Psplit97 (Unregistered)"

:0043FE57   BAE4FE4300          mov  edx,  0043FEE4
:0043FE5C  A16C514500           mov  edx,dword ptr [0045516C]
:0043FE61   E88A2CFDFF          call 00412AF0
:0043FE66   C6055651450000      mov byte ptr [00455156], 00
```

Now as soon as I seen this I knew I had the program cracked. See how on the line 0043FE66 it moves the 00 into memoey location [00455156]? Now most programs have a variable called registered or isreg, and its true or false (Boolean). Your computer stores true or false as either 1 (true usally) or 0 (false usally). The variable are always memory locations in the bracket like [00455156]. So if you ever see something comparing something in brackets to 1 or 0 thats usually part of the protection. Anyway look a little bit above the code we just stopped at :

```
 :0043FE13   8B45F8          mov   eax, dword ptr [ebp-08]
 :0043FE16   8B55F0          mov   edx, dword ptr [ebp-10]
:0043FE19   E8AA38FCFF      call 004036C8
 :0043FE1E   7537            jne  0043FE57

*Possible StringData Ref from Code Obj ->"Psplit97, Register to"

 :0043FE20   BAC0FE4300      mov  edx,  004EFC0
:0043FE25  8D85ECFEFFFF    lea  eax, dword ptr [ebp+FFFFFEEC]
 :0043FE2B   E8A836FCFF      call  004036C8
:0043FE30  8D85ECFEFFFF    lea  eax, dword ptr [epb+FFFFFEEC]
 :0043FE36   8D55FC          mov edx, dword ptr [ebp-04]
 :0043FE39   E88237FCFF      call 004035C0
:0043FE3E  8B95ECFEFFFF   mov  edx, dword ptr [ebp+FFFFFEEC]
 :0043FE44   A16C514500      mov  eax, dword ptr  [0045516C]
:0043FE49  E8A22CFDFF     call  00412AF0
 :0043FE4E  C6055651450001   mov  byte ptr  [00455156],01
:0043FE55  EB16           jmp  043FE6D
```

Ok, so now we know if we are a registered user this code will be executed and the title of the program will be "Psplit97", Registered. But look at the line 0043FE4E. Remember the variable [00455156] earlier that unreged users had set to 0, it looks like that if reg'ed users have that set to 1.
So now lets see how we can change it so we have that always set to 1 and the title always says registered to. Well right before we see the text "Psplit registered to" we see a jne (jump not equal) and where does it jump to? the text says unregistered.
So that is obviously the jump that determines wether or not we are regged, and right before it is a call, so it is safe to assume that call compares 2 things and says wether or not they are equal (wether or not we are regged basically). Well if you want to make an actual crack you could just change to jne 0043FE57 (hex bytes: 75 37) to 2 NOP's (NOP means "no operation, do nothing basically) (hex bytes: 90 90). We need 2 NOP's because the jump takes 2 bytes and a nop is only one byte. So we copy down a couple of bytes before the jne and a couple of bytes after, then open up Psplit97.exe and search for the following :

```
search:  8B55F0E8AA38FCFF7537
replace: 8B55F0E8AA38FCFF9090
```

Notice the NOP's instead of the jne? We save the changes and run it. What the hell?!?! Its still says unregistered! Thats not possible though, is it? Well maybe its because there is no user name for the program in our registery and one of those calls checks for it. So go into the Psplit, then look in the 97 directory. We see an empty key called User and one called RegKey. I entered "VipVop" for user. Now run the program again. Cool it says registered to +VipVop. So now we could distribute the crack and tell people to enter whatever they want in the registry, but that would be kind of sloppy to do.
Lets look at that call that determines if we are registered.

```
0043FE13  8B45F8          mov  eax, dword ptr [ebp-08]
0043FE16  8B55F0          mov  edx, dword ptr [ebp-10]
0043FE19  E8AA38FCFF           call  004036C8
0043FE1E  7537
                          jne  0043FE57
```

Well it looks like the call to see if we registered or not to take arguements to it.
In C code prob looks like this:
int isregged (char realnumber[],char usernumber[])
But wait a min, one arguement is ebp-08, the other is ebp-10. Thats only a 2 byte difference, not enough for a whole reg code. But there is enough room for the address of the string its its references. So now the code looks like this:
int isregged(*realnumber[],char *usernumber[])
So now we know that for ebp-08 and ebp-10 that one holds the address and the number in the registery, and the other holds our reg number. So if we could be at the line in Soft-Ice we could just enter the command. d (address) and see the real reg number for whatever name we want to enter into the registry. But how do we get there in Softice? Ok, remember how we said that [00451556] is prob the is_reg varible? Well the lines of code we want to see are only a little before the [00455156] variable is set, so if we could scroll up and set another breakpoint on the lines we actually want to be at. So lets try that.

You have to use the softice loader for this though. Load up Psplit97.exe, and you will pop into softice before any of the programs code is executed. Now Enter The following command:
>bpm 00455156 RW
That is the breakpointonMemoryaccess, for address 00455156, and the RW means if it is written to. Now press ctrl-d to get out of softice, the program will run for a split second then you will pop back into softice. Now press ctrl-(uparrow) to scroll up the code a little bit until you find the line that says:

0043FE13 8B45F8	mov eax, dword ptr [ebp-08]

Double click on the that line (all it does is automatically sets a breakpoint on the execution for that line, meaning next time that line is executed softice will pop-up) now close PSplit97 and run it again. You will pop up on that line . Press F10(step) so the value of ebp-08 is moved into eax, then type the following: >? eax
The question mark tells you the value of. You will see something like this:
008F002 0009371560 "(some weird characters)"
Ok, the number on the far left is the hex value of eax, the middle is the decimal (base10) value, and the one on the right is the ASCII value. So now type the following:
>d 008F002 (don't actually put 008F002, put whatever it said when you typed ?eax)

Now in the date window in softice you will see the reg code(if any) you entered into the registery from earlier. Now do the same thing for the edx register, and you will see that looks an awful lot like a reg code. (For +VipVop it was 1882-PS-2117) Write that number down, now go into the program and click the register(Note:this assumes you made a backup of Psplit97.exe before you used the hex-editor on it, and that you are running the back-up one, because otherwise it won't give you a chance to register). Enter the name you put into the registry and the reg number you just got and there you go, you are registered with a working serial number for your name and don't have to distribute a messy crack we made earlier (for Splatter its 1881-PS-1555)

One last thing, if you look a couple lines above the call to see if we are regged you will see the string reference to "-PS-". Remember how the reg code had that in the middle, or actually all reg codes have that in the middle? Well that means the code directly above is the code that actually determines what the number is from the user name entered. If we felt like it we could study that code and make a key-generator, but I feel why spend so much time on something , we already have it registered to us...so who cares.

How to Crack
Was written by : +Vip-Vop
And published by : Splatter Industries,Inc.

The End