Tutor6

In the previous tutorial I explained you how to crack a program by modifying the heart of the protection : Its serial validation routine. This kind of protections are used very often. In this tutorial I will demonstrate one :-)

The program we are going to crack here, is a website mirroring / web spidering program.
It is called Aeria Leech version 1.00. You can download it from : http://www.aeria.com/leech
I released a keygenerator for this program and it took me about 1 week to figure out how the key generation routine worked. But it is so easy to crack :-) We will do it together …
In just a few minutes.
Let's get going ;-)

Step 1: Run the program:
Run it. In the help menu you have the option to register.
Enter any number and press ok.
You get an error message : The key is not a valid license key for this product.
Ok. Let's go on to the next step :)

Step 2: Disassemble the program:
Disassemble, and look for the string you see above.
Found.
This string reference brings us to the next step.

Step 3: Analyzing the protection routine.... / Understanding the jumping Mechanism...
Let's analyze the protection routine.

```
////////////////////// Code snip ////////////////////////////
ADDRESS   MACHINE CODE        ASSEMBLER INSTRUCTIONS

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00405E11(C)
|
:00405F28 E86F8C0300              call 0043EB9C
:00405F2D 8B4004                  mov eax, dword ptr [eax+04]
:00405F30 6A10                    push 00000010

* Possible StringData Ref from Data Obj ->"Invalid License Key"
                                   |
:00405F32 6808844500              push 00458408

* Possible StringData Ref from Data Obj ->"The key is not a valid license "
                                        ->"key for this product."
                                   |
:00405F37 68D0834500              push 004583D0
////////////////////// Code snip ////////////////////////////
```

Well ... This piece of code sure looks familiar :-)
This code is reached after a jump at address : 405E11. The reference shows that.
Let's go back to address 405E11.

```
////////////////////// Code snip ////////////////////////////
ADDRESS   MACHINE CODE        ASSEMBLER INSTRUCTIONS

:00405E04 83C408                  add esp, 00000008
:00405E07 E864380000              call 00409670
:00405E0C 83C404                  add esp, 00000004
:00405E0F 84C0                    test al, al
:00405E11 0F8411010000            je 00405F28
////////////////////// Code snip ////////////////////////////
```

This jump at address 405E11 jumps over all the piece of code that tells us that we are registered. So nopping this jump away would make us display the nice registered dialog box :-)
But luckily we worked through the previous tutorial. and we first try to analyze this code.

```
////////////////////// Code snip ////////////////////////////
call 409670    ; isregistered ?
test al, al    ; result of the call is in eax :-)
               ; ( al is the lower part of eax, not important now)
je             ; if eax = 0  then jump ! if eax is 1 don't jump !
////////////////////// Code snip ////////////////////////////
```

Well ... this code is clear not ?
The call checks the entered serial. And if the serial is ok, it returns 1 in eax, else it returns a zero in eax.
Let's trace into the call at address 409670.

```
////////////////////// Code snip ////////////////////////////
ADDRESS   MACHINE CODE        ASSEMBLER INSTRUCTIONS

* Referenced by a CALL at Addresses:
|:004014BF   , :00405E07
|

* Possible Reference to Dialog: DialogID_00CB, CONTROL_ID:00FF, ""
                                |
:00409670 6AFF                  push FFFFFFFF
:00409672 68F8424400            push 004442F8
:00409677 64A100000000          mov eax, dword ptr fs:[00000000]
////////////////////// Code snip ////////////////////////////
```

And yes ... We were right :-)
This call is referenced twice. Once at startup and once while entering the serial :-)

Step 4: Changing the original program...

Since we analyzed the jumping mechanism carefully, we can change the original program.
Open the program in hiew after making a backup copy of it.
Get to the beginning of the call at address 409670.

Change the beginning of the call into the following :

```
////////////////////// Code snip ////////////////////////////

mov eax, 1     ; 1 means serial is ok
ret            ; return to the caller.
////////////////////// Code snip ////////////////////////////
```

It will look like the following:

```
////////////////////// Code snip ////////////////////////////
ADDRESS   MACHINE CODE        ASSEMBLER INSTRUCTIONS
00009670: B801000000                  mov       eax,000000001
00009675: C3                          retn
00009676: 0064A100                    add       [ecx][00000],ah
////////////////////// Code snip ////////////////////////////
```

Save your patched program and exit hiew.

Step 5: Testing your cracked program...
Run Leech. It will still tell you that you are unregistered in the about box.
Register it with any number, and it will happily accept your serial :-)

Do not forget to close the program and restart it. You will see that it is still happy with the provided serial :-)