

Ghiribizzo's Cracking Tutorial

Lazy Cracking : UltraEdit-32 Version 5.0 (Part 1)

UltraEdit-32 Version 5.0 (Part 1)

Well, Aesculapius has beaten me to writing this tutorial so I won't go over the serial scheme here (see Aesculapius' homepage at Monolith for more details). Instead I shall go over the techniques I used to crack this target quickly.

PGP and Signed Tutorials

My tutorials and programs should be signed electronically using PGP. PGP 5 supports DSS/Diffie-Hellman keys. These keys are not supported by previous versions of PGP.

You should check the signature to make sure that the tutorial and especially its program files have not been tampered with. All cracks, tutorials and zip files I release will be signed. This will prevent tampering and will hopefully reduce the chances of viral infection.

My signature will also be the only way you can identify me as my email address will often change.

My Web Site: <http://Ghiribizzo.home.ml.org>

My Email: Ghiribizzo@geocities.com

My Backup Email: Ghiribizzo@hotmail.com

This document is Copyright © 1997 by Ghiribizzo. This document may be distributed non-commercially, provided that is it not modified in any way (including change of format). This publication may not be sold or packaged, in whole or in part, as a service, or with a product for sale in any form without the prior written permission of the author. This document is presented with no warranties or guarantees of any kind including fitness for any particular purpose. If you use the information contained herein, you do so at your own risk.

Scan Strings and Rawlisting with Hiew

I have been preparing this tutorial for the past few weeks, but have noticed that Aesculapius has beaten me to it! Although I haven't read his tutorial in any great detail, I know he is a good cracker and his coverage of the serial generation scheme should be very good. Therefore, I will save you from having to read through an identical tutorial and instead talk you through some of the tricks I used to crack it.

Firstly, I have cracked a previous version UltraEdit and as the author seems to like to change the serial scheme often, I took some steps to minimise future work on newer versions - this has paid off. Those of you who write viruses and release them into the wild will know the battle to reduce the length of scan strings. Well, having cracked the earlier version, I dumped critical parts of the protection routine from Hiew. Then simply using Hiew again to look for the strings, I could easily locate and crack the target in Hiew. Rawlisting with Hiew, my initial search strings are indicated in red. As, the entered password location is loaded several times and the suspicious call at 442A0 is made after pushing the passwords, I decided rather to alter the lea instructions to go to the 442A0 call itself and alter the passwords it retrieved from the stack:

```
Comparing files UEDIT32.EXE and UEKEYGEN.EXE
000442A3: 04 08

0000AB89: 8D45C0          lea     eax,[ebp][-0040]
0000AB8C: 50             push    eax
0000AB8D: 8D4580          lea     eax,[ebp][-0080]
0000AB90: 50             push    eax
0000AB91: E80A970300     call   0000442A0 ----- (8)
0000AB96: 59             pop     ecx
0000AB97: 85C0           test    eax,eax
0000AB99: 59             pop     ecx
0000AB9A: 741E           je      00000ABBA ----- (9)
0000AB9C: 8D8540FFFFFF    lea     eax,[ebp][0FFFFFFF40]
0000ABA2: 50             push    eax
0000ABA3: 8D4580          lea     eax,[ebp][-0080]
0000ABA6: 50             push    eax
0000ABA7: E8F4960300     call   0000442A0 ----- (A)
0000ABAC: 59             pop     ecx
0000ABAD: 85C0           test    eax,eax
0000ABAF: 59             pop     ecx
0000ABB0: 7408           je      00000ABBA ----- (B)

000442A0: 8B542404       mov     edx,[esp][00004]
000442A4: 8B4C2408       mov     ecx,[esp][00008]
000442A8: F7C203000000   test    edx,000000003
```

This was all done within Hiew. However, the crack seems fine superficially - the nags disappear and the about box shows that you are registered, however, the REG file produced is NOT a valid REG file. If you undo the crack, the nags appear again. I haven't investigated why this is so yet, but in Ghirc11a.zip you will find the above crack with a program to reverse it. It may be quite interesting to look into this itself.

Scan Strings and Embedded Breakpoints with SoftICE

I had not actually the above fault until the distribution package of the alpha version crack was being made. However, I do not usually like modifying the executables at all so in fact, the first way I actually cracked it was to again search for the above strings and then place a breakpoint at AB90. I did this by replacing 50 with CC in Hiew and then I ran SoftICE and turned on the embedded breakpoints feature. You do this by typing: 'SET I3HERE ON' in the SoftICE command window. Load the program in symbol loader and then let it run. It should break at the correct location (will liberal scattering of CCh in code make a come back?). You can now dump the contents of ebp-40 to get your serial number. Then to restore the code, what I did was simply rip to AB8C and run the push eax command and then rip back to AB91. However, I'm sure there should be a more elegant way of doing this.

Another SoftICE trick

I then decided to use the live technique to get a feeling for the code in preparation to see if there were any other little tricks about and to take a closer look at the serial scheme - though this was a secondary concern as the serial scheme varies so often.

If you use filemon, you will notice that the program tries to open UEDIT32.REG - the key file. I used this to break in at the correct place. However, this is not the only file that UltraEdit opens so it pays to do a little more work when setting breakpoints to make our lives a little easier. Instead of `bpx createfilea` (programs usually check for the existence of files before trying to open them by using this function) a `bpx createfilea do "d esp+14"` will be much better as when we break, the dump window will show the name of the file being dumped. Breakpoints can be enhanced by some API research and judicious use of the DO command. Remember also that SoftICE has a quite powerful macro ability which can be very useful (especially when tracing though self-tracing code). Unless, you use all the function keys, it may be worth re-mapping them to something different. I also recommend moving program step 'P' to 'F8' and trace 'T' to 'F7'. This will bring SoftICE in line with Turbo Debugger and then swapping between debuggers is much less hassle - in any case, I find it insane that these two keys should be so far apart on the keyboard by default.

One last word on SoftICE - although I've heard many good things about version 3.2x of SoftICE for 95, I still haven't tried it yet. I hear that the graphics driver is good (no more annoying text mode change). I recommend that you buy a second computer and use the 2 monitor debugging facility - you won't ever go back! It's simply amazing. The cost of hardware is going down all the time and to be honest it works perfectly fine on a 286 1mb system (2nd hand systems can be found at dirt cheap prices). It may take a while before you get the hang of typing on the correct keyboard though! :-)