WinTar-Remote tut! 24/08/97

=-=

Program: WinTar-Remote
Version: 2.2.1
URL:http://www.spiralcomm.com
Description: i know shit about this program i picked up cause of the
size
Operating System: Windows
Cracker: nlabl [Me'97]
Level: Intermediate
Tools: SoftICE, W32Dasm, a Hex Editor.
Protection Type: 30 day trial
Encrypted/DLL: No
Method: Dissasemble

0.-index:

1.- Intro:

Hello, ok here again on another tut for C4N, this time i am goin to talk about Time Trials Even tough they are easy a lot of ppl still don't get it so this is why this tut is gone (hopefully) teach you, also i will teach some of nag remove and bmp (splash) screens :-)

ok, the program had to be a time trial (of course) but we need it a not to big program but one
that had some potencial in it or i could have used Rhino 3d wich is not small and does not have any teaching potential (u changed one byte and it's cracked) so ok with the help of a friend Griml0ck we decited to get this program is called WinTAR-Remote by SpiralCom Communications Inc. what this program does is not important to us we wil crack it and delted it for educational purpose ONLY :-).

In this tut i will asume u know how to use all of the tools i will use here else please get other
tuts that do explain how to use them (TKC's, Edison's, josephCo's and others)

2.- What We need (tools):

W32dasm (used mostly)
SoftIce
Any Hexeditor
a patch maker (if we want to release our crack), i recomen Gpatch by jes and patchit by Qapla
gpatch i like better cause of ease of use and does some good patches on the other hand patchit
gives u the source of the patch in C :-), other wiseuse Pascal or C and do ur own patch (not
explained in this tut sorry).

3.- Let's Crack the splash screen:

ok once d/l the program u run it add se a nasty splash that says Thanks for trying WinTar blah
blah,blah after some secs it shows u a license aggrement (ewww), now we don't like those 2 things
so let's start by taking them away we enter softice and set a bpx on LoadBitmapA once we do this
we run the program again and boom u in Softice cause of one of the bpx u seted b4 now we can see
this (from the w32dasm dissaemble) :

```
* Reference To: USER32.SetTimer, Ord:01FEh   ; set time the splash screen is goin to show
                                  |
:0040F5F4 FF15F0C64200            Call dword ptr [0042C6F0]
:0040F5FA E92D010000              jmp 0040F72C

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040F6FF(C)
|
:0040F5FF 6A67                    push 00000067    ; hmm nice push here (does nothing good)
:0040F601 A124A54200              mov eax, dword ptr [0042A524]
:0040F606 50                      push eax

* Reference To: USER32.LoadBitmapA, Ord:0165h      ; this is where u land
                                  |
:0040F607 FF15D0C64200            Call dword ptr [0042C6D0]
:0040F60D 8945DC                  mov dword ptr [ebp-24], eax
:0040F610 8D859CFEFFFF            lea eax, dword ptr [ebp+FFFFFE9C]
:0040F616 50                      push eax
:0040F617 8B4508                  mov eax, dword ptr [ebp+08]
:0040F61A 50                      push eax
```

```
* Reference To: USER32.BeginPaint, Ord:0009h  ; begin the painting of the splash
                                  |
:0040F61B FF1574C64200            Call dword ptr [0042C674]
:0040F621 8945F8                  mov dword ptr [ebp-08], eax
:0040F624 8B45F8                  mov eax, dword ptr [ebp-08]
:0040F627 50                      push eax

* Reference To: GDI32.CreateCompatibleDC, Ord:001Fh
                                  |
:0040F628 FF1590C44200            Call dword ptr [0042C490]
:0040F62E 8945FC                  mov dword ptr [ebp-04], eax
:0040F631 8B45DC                  mov eax, dword ptr [ebp-24]
:0040F634 50                      push eax
:0040F635 8B45FC                  mov eax, dword ptr [ebp-04]
:0040F638 50                      push eax

* Reference To: GDI32.SelectObject, Ord:013Ch
                                  |
:0040F639 FF15B0C44200            Call dword ptr [0042C4B0]
:0040F63F 8D45E0                  lea eax, dword ptr [ebp-20]
:0040F642 50                      push eax
:0040F643 6A18                    push 00000018
:0040F645 8B45DC                  mov eax, dword ptr [ebp-24]
:0040F648 50                      push eax

* Reference To: GDI32.GetObjectA, Ord:00DEh
                                  |
:0040F649 FF1598C44200            Call dword ptr [0042C498]
:0040F64F 682000CC00              push 00CC0020
:0040F654 6A00                    push 00000000
:0040F656 6A00                    push 00000000
:0040F658 8B45FC                  mov eax, dword ptr [ebp-04]
:0040F65B 50                      push eax
:0040F65C 8B45E8                  mov eax, dword ptr [ebp-18]
:0040F65F 50                      push eax
:0040F660 8B45E4                  mov eax, dword ptr [ebp-1C]
:0040F663 50                      push eax
:0040F664 6A00                    push 00000000
:0040F666 6A00                    push 00000000
:0040F668 8B45F8                  mov eax, dword ptr [ebp-08]
:0040F66B 50                      push eax

* Reference To: GDI32.BitBlt, Ord:000Ah
                                  |
:0040F66C FF1588C44200            Call dword ptr [0042C488]
:0040F672 8B45FC                  mov eax, dword ptr [ebp-04]
:0040F675 50                      push eax
```

```
* Reference To: GDI32.DeleteDC, Ord:0043h
                                  |
:0040F676 FF1584C44200           Call dword ptr [0042C484]
:0040F67C 8B45DC                mov eax, dword ptr [ebp-24]
:0040F67F 50                   push eax

* Reference To: GDI32.DeleteObject, Ord:0046h
                                  |
:0040F680 FF158CC44200           Call dword ptr [0042C48C]
:0040F686 8D859CFEFFFF           lea eax, dword ptr [ebp+FFFFFE9C]
:0040F68C 50                   push eax
:0040F68D 8B4508                 mov eax, dword ptr [ebp+08]
:0040F690 50                   push eax

* Reference To: USER32.EndPaint, Ord:00AFh
                                  |
:0040F691 FF1570C64200           Call dword ptr [0042C670]
:0040F697 B801000000            mov eax, 00000001
:0040F69C E992000000            jmp 0040F733

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040F721(C)
|
:0040F6A1 8B4510                 mov eax, dword ptr [ebp+10]
:0040F6A4 50                   push eax
:0040F6A5 8B4508                 mov eax, dword ptr [ebp+08]
:0040F6A8 50                   push eax

* Reference To: USER32.KillTimer, Ord:0162h ; kiil the timer set b4 to show the splash
                                  |
:0040F6A9 FF15F4C64200           Call dword ptr [0042C6F4]
```

ok u can see here one thing the line that contains push 00000067 in 40f5ff does nothing so to crack the splash screen we chage this

```
:0040F5FF 6A67                  push 00000067    ; hmm nice push here (does nothing good)
to this
:0040F5FF E9A5000000            JMP  0040F6A9    ; Nice jump, kills the timer and the
splash
```

so here the splash screen is disabled and we can continue cracking.

4.- Lic. screen removal:

ok this par needs some zen cracking :-) this is part of the disssemble in w32dasm :

```
:004094DD 813D3C5A420000010000    cmp dword ptr [00425A3C], 00000100
:004094E7 0F8533000000            jne 00409520
:004094ED 8B4508                  mov eax, dword ptr [ebp+08]
:004094F0 50                      push eax
:004094F1 E80AEFFFFF              call 00408400    ; call the lic screen(how did i got here ?
                                                   ; like i said zen cracking :-)
:004094F6 83C404                  add esp, 00000004
:004094F9 85C0                    test eax, eax
:004094FB 0F851F000000            jne 00409520
:00409501 C705105C420001000000    mov dword ptr [00425C10], 00000001
:0040950B 6A00                    push 00000000
:0040950D 6A00                    push 00000000
:0040950F 6A10                    push 00000010
:00409511 8B4508                  mov eax, dword ptr [ebp+08]
:00409514 50                      push eax
```

this is what the call to the lic screen is :

```
* Referenced by a CALL at Address:
|:004094F1
|
:00408400 55                      push ebp      ; this code is only checking if the file is not
                                                ; delted or something like that
:00408401 8BEC                    mov ebp, esp
:00408403 83EC08                  sub esp, 00000008
:00408406 53                      push ebx
:00408407 56                      push esi
:00408408 57                      push edi
:00408409 C745F867844000          mov [ebp-08], 00408467
:00408410 6A00                    push 00000000
:00408412 8B45F8                  mov eax, dword ptr [ebp-08]
:00408415 50                      push eax
:00408416 8B4508                  mov eax, dword ptr [ebp+08]
:00408419 50                      push eax
:0040841A 6A66                    push 00000066
:0040841C A124A54200              mov eax, dword ptr [0042A524]
:00408421 50                      push eax

* Reference To: USER32.DialogBoxParamA, Ord:008Ah
                                  |
:00408422 FF15C8C64200            Call dword ptr [0042C6C8]
:00408428 8945FC                  mov dword ptr [ebp-04], eax
:0040842B 837DFC02                cmp dword ptr [ebp-04], 00000002
:0040842F 0F8512000000            jne 00408447

* Possible Reference to String Resource ID=03302: "The licence agreement file is missing
or
```

```
                                        corrupted.  Please "
                                        ; as u can see here if u delete the
                                        ; licence.txt u get this msg
```

ok what we can do here is this since none of the checking of calling is done AFTER the call once it finds a ret the program says ok this guy pushed the i agree button, continue, so what we can do here is give the program a ret, whe change this :

```
:00408400 55                    push ebp
to this
:00408400 C3                    ret
```

the program calls the screen but a ret(return from call) is there so it returns to the program.

5.- The 1s part of the time trial:

ok now once we dissabled all of the nag's and nasty stuff we need to take the 30 day trial we try and find something on the nag box in w32dasm what we find is just a lot of garbage in this nag (not gabage but dificult to follow) how about something else ? hmm the .ini ? ok let's try we search for it and land here :

```
* Possible StringData Ref from Data Obj ->"wintar.ini"
                                  |
:00409275 A1485A4200            mov eax, dword ptr [00425A48]
:0040927A 50                    push eax
:0040927B 6A00                  push 00000000

* Possible StringData Ref from Data Obj ->"Validate"
                                  |
:0040927D 68405C4200            push 00425C40

* Possible StringData Ref from Data Obj ->"UserOpt"
                                  |
:00409282 684C5C4200            push 00425C4C

* Reference To: KERNEL32.GetPrivateProfileIntA, Ord:00F9h
                                  |
:00409287 FF152CC54200          Call dword ptr [0042C52C]
:0040928D 8985F4FEFFFF          mov dword ptr [ebp+FFFFFEF4], eax
:00409293 E91A000000            jmp 004092B2

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:0040926F(C)
```

```
|
:00409298 6A00                  push 00000000

* Possible StringData Ref from Data Obj ->"Validate"
                                  |
:0040929A 68545C4200            push 00425C54

* Possible StringData Ref from Data Obj ->"UserOpt"
                                  |
:0040929F 68605C4200            push 00425C60
:004092A4 E896E2FFFF             call 0040753F              ; if you follow in SI here u will
                                          ; find that this call does
                                          ; does something strange so we
                                          ; go to the call
:004092A9 83C40C                add esp, 0000000C
:004092AC 8985F4FEFFFF           mov dword ptr [ebp+FFFFFEF4], eax

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00409293(U)
|
:004092B2 83BDF4FEFFFF00         cmp dword ptr [ebp+FFFFFEF4], 00000000
:004092B9 0F850D000000           jne 004092CC
:004092BF E89CE8FFFF             call 00407B60              ; take a deep look :-)
:004092C4 85C0                  test eax, eax
:004092C6 0F849B000000           je 00409367
```

this is what we get by the call at 4092A4

```
* Referenced by a CALL at Addresses:
|:004092A4   , :00410C4F   , :00410C7F   , :00410C98   , :00410CB1
|:00410CCA   , :00410CE3   , :00410CFC   , :00410D15   , :00410D2E
|:00410D47   , :00410D60   , :00410D80   , :00410D99   , :00410DB2
|:00410DCB   , :00410DE4   , :00410DFD   , :00410E16   , :00411304
|:0041131D   , :00416C74   , :00416C8F   , :00416CAA   , :00416F4F
|:00416F6A   , :00416F85   , :00417415   , :00417622   , :004177C1
|:004177E2   , :0041788D   , :00417961   , :00417982   , :004179A3
|
                                   ; WOW this part sure does get called !
:0040753F 55                    push ebp
:00407540 8BEC                   mov ebp, esp
:00407542 81EC14010000           sub esp, 00000114
:00407548 53                    push ebx
:00407549 56                    push esi
:0040754A 57                    push edi
:0040754B C745F404010000         mov [ebp-0C], 00000104
:00407552 833D3856420000         cmp dword ptr [00425638], 00000000 ; is the flag Zero
?
:00407559 0F8507000000           jne 00407566                   ; no then bug off
```

```
:0040755F 33C0                  xor eax, eax
:00407561 E9A0000000            jmp 00407606
```

what we can do here is simple we look at our Registers ans check is EAX is zero b4 it called this
part........ we check and see that it is zero so this is getting better :) what we do here is simple ok remeber the lic. removal part how the call only wanted a ret ? ok so this is equal change this:

```
:0040753F 55                    push ebp
```
to this
```
:0040753F C3                    RET
```
there now the MARKER (if you set the time ahead or b4 30 days) is removed.

6.- The 2nd part of the time trial:

ok now we need to remove the 30 day check this will ALSO require more zen (this is prolly a zen
tut and not a time trial :]) but not many zen if u are a good looker u can see this call after the check mark call :

```
:004092BF E89CE8FFFF            call 00407B60 ; this is our check our time call :-)
```

unlucky us u can't do the RET trick here :-(so we go deep inside the call and find this:

```
* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00407CA1(C)
|
:00407CB1 833DB457420000        cmp dword ptr [004257B4], 00000000 ; check the flag
to zero
:00407CB8 0F850A000000          jne 00407CC8            ; no? the bug off
:00407CBE B801000000            mov eax, 00000001       ; and move EAX to 1
                                                        ; wich 1 = bad time
:00407CC3 E902000000            jmp 00407CCA            ; jump to return

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00407CB8(C)
|
:00407CC8 33C0                  xor eax, eax

* Referenced by a (U)nconditional or (C)onditional Jump at Address:
|:00407CC3(U)
|
:00407CCA E900000000            jmp 00407CCF

* Referenced by a (U)nconditional or (C)onditional Jump at Addresses:
```

```
|:00407BA4(U), :00407BBA(U), :00407BE6(U), :00407C1A(U), :00407C65(U)
|:00407CCA(U)
|
:00407CCF 5F                 pop edi
:00407CD0 5E                 pop esi
:00407CD1 5B                 pop ebx
:00407CD2 C9                 leave
:00407CD3 C3                 ret
```

ok now here the program is looking for something, what could it be ?..........
ok if we continue with eax in 1 we get the sorry screen and a help file opens and our program
terminates, we don't like this so we go back here and check again, ok i got it it checks if eax
is ZERO if it is then the guy is still on the 30 day limit, so we change this :

```
:00407CBE B801000000         mov eax, 00000001            ; and move EAX to 1
```

to this

```
:00407CBE B800000000         mov eax, 00000000            ; and move EAX to 0
```

now the program even if you are on the 30 day limit it will let you use it for the rest of your life :-).

7.-Last Notes:
ok now to finally do our crack we enter a hexeditor and search for the opcodes and change them
(like,i said at the beggining i assume you allready know this).

8.-Notes:

You could search for the text UNREGISTERED and changed to anything u like like CrackedVer.
ans search for the string Days left and change it to anything as well i will not explain this
because i think AT least the programmers deserve that since u cracking the software :-).

9.-Thak you's:

Ok thaks go to the follwing persons:
JosephCo: keep up the good work d00d
mpbaer: ha Rebirth ROX !!!!!! :)))
Razzi: ur tuts rule !!!
^pain^: cause u cool :)
tHATDUDE: he isnpired me to become a cracker :-)
Fant0m : damm ur coding is good
GThorne: haha this guy rox the world !
Tgunner: 10x for everything

lgb: 10q as well for all the help and support :)
blorght: the only female i seen (err on irc) that can do a lot of stuff ! u rule babe :-)
Griml0ck: he inspired me and asked me to this tut :-) ok d00d for you here it goes.
TeRaphY: this guy is kewl as well :)
Krazy_N: he is not crazy but he is kewl :)
all the regulars of #cracking4newbies thanks that shows us that we growing ! :-)
#cracking all of the guys in it aswell retf in especial :-P
#revolt bring up the warez ! :)
cat|man: thanks for those sites :)
if i forgot anyone please let me know i will respond ahh ok 10q :)
oh and also all of the ppl that shows some cracking teaching or explaining !!

nIabI[ME'97]