Cours de Cracking

(11ième Partie)

Mon objectif : craquer Unreal no-cd

1/ Les logiciels utiles pour ce cours

-> Un désassembleur : **W32dasm 8.93**
-> Un éditeur hexadécimal : **Winhex 10.2**

2/ Présentation

Unreal a été, en son temps, un jeux magnifique, super beau, super jouable, avec une bande-son superbe aussi. Malheuresement, pour en profiter pleinement, il faut l'installer complètement sinon ses temps de chargement CD sont beaucoup trops longs.

Une fois UNREAL installé complètement, à chaque fois que je démarre le jeux, il me demande ce CD. Comme je suis de nature assez bordélique, et que j'égare souvent mes CD dans le foutoir qu'est my bedroom, j'ai décidé de faire un crack pour jouer sans CD.

Dans ce tutorial, je vais donc vous expliquer mon raisonnement... Ca n'a pas été évident du fait que j'ai dû tout détailler alors que ce crack ne m'a pris que 5 minute. Hééé oui, c'est certainement un des cracks que j'ai fait le plus rapidement, c'est donc pas forcément parcequ'on s'attaque à une grosse boîte, que le crack sera dur.

2/ Le crack

Au début, on serait tenté de désassembler le fichier Unreal.exe, hophophop! Pas du tout! Quand on démarre UNREAL sans CD la boîte de dialogue qui réclament le CD vient du fichier Window.dll.

NdSmeita: de facon generale, quand vous trouvez rien d'interressant dans le fichier '.exe' et qu'il existe des Dlls qui accompagne le programme, il ne coute rien de desassembler ces Dlls, histoire de voir si c'est pas dedans que ca se passe :)

-> Il faut donc désassembler le fichier window.dll. (après avoir fait une copie de sauvegarde...).
-> Lancez une recherche sur la String Reference " Cd Required". On tombe alors sur cette partie:

```
:10B1C3D5 A114B7B310     mov eax,dword ptr [10B3B414]
:10B1C3DA 833800         cmp dword ptr [eax], 00000000    //Regarde si eax=0
:10B1C3DD 75E3           jne 10B1C41D
:10B1C3DF 8D8DE49FFFF    lea ecx, dword ptr[ebp+FFFFFF9E4]
:10B1C3E5 51             puch ecx
:10B1C3E6 FFD3           call ebx
:10B1C3E8 83C404         add esp, 00000004
:10B1C3EB 85C0           test eax, eax        //Test
:10B1C3ED 7F2E           jg 10B1C41D           //même saut qu'en 10B1C3DD...
:10B1C3EF 680120000      push 00002001


*Possible StringData Ref from Data Obj - >"Cd Required"
:10B1C3f4 68089DB310        push 10B39D08


*Possible StringData Ref from Data Obj - > "Please insert CD-Rom in drive"
```

-> En 10B1C3DD on effectue un saut. On remarque qu'il atterit juste après le "ExitProcess" (donc juste après les messages d'erreur) ce qui signifie que la procédure de fermeture n'est pas prise en compte si le registre eax vaut 1.

[Message Pifoman] En effet l'instruction cmp dans l'instruction cmp dword ptr [eax], 00000000 (comme dans toute instruction cmp) effectue le calcul [eax] - 00000000 en mettant l'indicateur ZF (Zero Flag) à 0 puisque la soustraction du contenu du registre EAX qui vaut 1 à 0 donne 1 (1-0=1). L'indicateur ZF est ensuite utilisé par le jne 10B1C41D. Si ZF=0 (ce qui est le cas ici) le jne (jump if not equal to zero) saute vers l'adresse qui suit le jne à savoir 10B1C41D.

-> Il suffit de remplacer le 'jne' par un 'jmp', saut qui s'effectue quoi qu'il arrive ! (*saut inconditionnel !!*)

-> C'est à ce moment là qui faut ouvrir window.dll avec votre éditeur hexadécimal préféré à savoir winhex.exe et de faire la modif' suivante ;-)). Faites CTRL ALT F dans winhex et

Cherchez -------> 75 E3 8D 8D
Remplacez par --> EB E3 8D 8D

Conclusion:

Les programmeurs de ce jeux sont pas cons du tout, car ils ont enfin compris qu'il était inutile de se faire ch??? en faisant des protection anti crack qui de toute façon se font largués les jours qui suivent.

PS : pour ceux que ça interresse, la valeur hexadécimale A114B7B30 (qu'on peut retrouver à l'addresse *10B1C3D5*) est en rapport avec le type de securité appelé Realloc@...