

Tutorial #6: CrackMe de Crudd

NIVEAU 1

Outils:

- Le [Crackme](#) de Crudd
- Un désassembleur/debugger: Win32asm
- Un éditeur hexadécimal: WinHex 10.2
- ResHacker

Observation du CrackMe:

- Level 1:
 - Un menu non-valable
 - Un CD-Check normal.
- Level 2:
 - Un menu non-valable un peu plus difficile
 - Un CD-Check plus difficile (pas de string data ref)
 - Un Anti-SI Serial number, ou FrogIce ne nous apprend rien.
- Level 3:
 - Plante sous Windows XP donc sans intérêt pour nous.

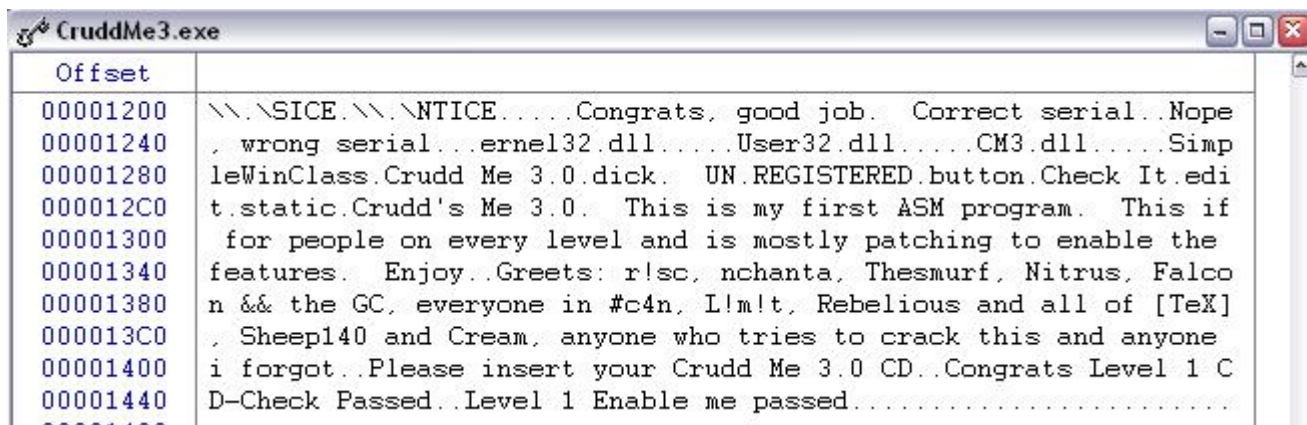
Objectifs:

Niveau 1

- Passer au statut "Registered" artificiellement
 - Rendre les menus bloqués accessibles
 - Virer un pop-up
 - Se débarrasser du CD-Check
-

Partie 1: Rendre le menu "Enable Me" accessible

On va commencer par s'attaquer a ce "unregistered" qui nous est pénible a voir. Rien de plus simple, il suffit d'ouvrir l'éditeur hexa, dans lequel on va modifier des octets. Dans WinHex, trouvez la ligne de texte Un Registered. Si vous ne la trouvez pas, cliquez simplement sur View, et tiquez la case *Text Display Only*, ou simplement sur F7. La, vous voyez en descendant la grosse portion de texte qui apparait a l'ouverture du CrackMe, comme ci-dessous:



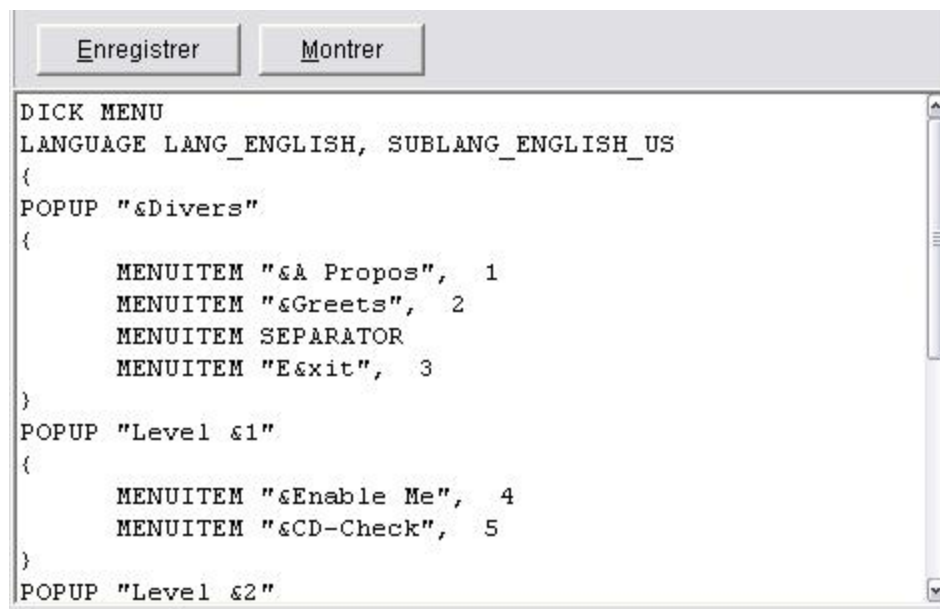
Offset	Text
00001200	\\.\SICE.\\.\NTICE.....Congrats, good job. Correct serial..Nope
00001240	, wrong serial...ernel32.dll.....User32.dll.....CM3.dll.....Simp
00001280	leWinClass.Crudd Me 3.0.dick. UN.REGISTERED.button.Check It.edi
000012C0	t.static.Crudd's Me 3.0. This is my first ASM program. This if
00001300	for people on every level and is mostly patching to enable the
00001340	features. Enjoy..Greetings: r!sc, nchanta, Thesmurf, Nitrus, Falco
00001380	n && the GC, everyone in #c4n, L!m!t, Rebelious and all of [TeX]
000013C0	, Sheep140 and Cream, anyone who tries to crack this and anyone
00001400	i forgot..Please insert your Crudd Me 3.0 CD..Congrats Level 1 C
00001440	D-Check Passed..Level 1 Enable me passed.....

Remettez la vue originale, positionnez-vous sur le "U" de "UN.REGISTERED", puis modifiez dans la fenêtre du milieu les octets 55 et 4E, qui correspondent bien sur en hexa aux lettres U et N. On va remplacer ces deux lettres par des 20, qui correspondent en hexa a un espace. Le tour est joue, et c'en est fait du message qu'on n'aime pas voir!

NB: au passage, on va mettre des espaces a la place de "SICE" et "NTICE", juste au-dessus, de manière a virer la protection anti-Softlce. Rien d'obligatoire cependant.

C'était vraiment bidon comme on l'a vu, donc on va maintenant faire une petite manip' avec ResHacker. Ouvrez le CrackMe avec et explorez le folder Menu (comme son nom l'indique il va nous montrer ce qui se trouve dans les menus du prog ouvert), puis DICK et enfin 1033.

On va se faire plaisir et changer le nom des menus, franciser tout ca un peu... et voir a quel point il est facile de changer le nom d'un menu dans un programme! Procédez comme suit sur l'image, le plus important étant bien sur de virer la partie qui dit "GRAYED", juste après le 4 de la ligne **MENUITEM "&Enable Me"**. Pensez bien a virer la virgule aussi (en virant "GRAYED", le menu ne sera plus grisé mais on pourra l'avoir en surbrillance):



On enregistre le tout avec *Fichier > Enregistrer*. On relance le CrackMe pour voir ce que ça donne, et voilà!

On a nos menus en Français, au lieu de "About" on a "A Propos", au lieu de "Shits" on a "Divers". Vous pouvez bien sûr mettre toutes les conneries que vous voulez.

De plus, lorsqu'on clique sur "Enable Me", un pop-up s'affiche et nous dit de patcher de manière à ce que le pop-up n'apparaisse plus. On remarque au passage que sur le titre de cette boîte de dialogue, on apparaît comme "REGISTERED".

Partie 2: Se débarrasser du pop-up dans "Enable Me"

On va faire comme on nous dit, et virer le pop-up. C'est parti pour une séance d'analyse classique avec notre bon vieux W32Dasm. Dans les SDR, on n'a rien de particulier qui nous intéresse, si ce n'est la string "*Level 1 Enable me passed*", qui veut dire qu'on a réussi à faire apparaître le menu. On double-clique dessus, et voici ce qu'on y trouve:

```
* Reference To: USER32.MessageBoxA,
Ord:01BBh
|
:004013FB E85A050000 Call 0040195A
:00401400 EB10 jmp 00401412

* Referenced by a (U)nconditional or
(C)onditional Jump at Address:
|:004013EB(C)
|
* Possible StringData Ref from Data Obj
->"Level 1 Enable me passed"
|
:00401402 6850324000 push 00403250
:00401407 FF3590324000 push dword ptr
```

Comme d'habitude, on va voir a l'adresse de référence en **004013EB** en faisant *Goto > Goto Code Location* puis on tape l'adresse.

```
* Referenced by a (U)nconditional or  
(C)onditional Jump at Address:  
|:004013C9(C)  
|  
|:004013EB 7415 je 00401402  
|:004013ED 6A00 push 00000000
```

On tombe la sur un saut de type JE. Procédure de vérification qu'il s'agit bien du saut de référence a notre pop-up:

1. Lancez le debugging avec CTRL+L
2. Allez a l'adresse du JE (004013EB)
3. Posez un BreakPoint sur le JE avec F2
4. Faites F9 pour lancer le process
5. Cliquez sur "Enable Me" a l'ouverture du CrackMe

Win32asm breake avant que le pop-up ne s'affiche, ce qui nous indique qu'on est au bon endroit. Quittez tout ça, et réfléchissez bien a ce qu'il va falloir faire pour ne plus avoir le pop-up (ceux qui ont suivi les cours précédents le savent déjà j'en suis sur)?

On sait que ce JE nous emmène vers un message qui dit qu'on a passe le Level 1, mais a condition que quelque chose... et nous on ne veut pas d'une condition pour non seulement aller vers le bon message mais ne plus avoir le pop-up. Il va donc suffire de transformer ce **JE** en **JMP**, pour toujours aller vers le bon message qui nous dit "**Level 1 Enable me passed**"...

A vos éditeurs hexa, rendez vous a l'adresse Offset 7EB (repérée dans Wdasm en bas dans la ligne d'info), puis changez le 74 en EB. Enregistrez puis lancez le CrackMe, et voila le travail!

Partie 3: Se débarrasser du CD-Check

On passe a l'étape suivante, jamais encore abordée pour nous: un **CD-Check**. Dans Wdasm, on trouve une string qui dit "*Please insert your Crudd Me 3.0 CD*" sur laquelle on clique, a défaut d'en voir une plus intéressante...

```
-----
* Reference To: KERNEL32.GetDriveTypeA,
Ord:00F0h
|
:00401424 E873050000 Call 0040199C
:00401429 83F805 cmp eax, 00000005
:0040142C 7412 je 00401440

* Possible StringData Ref from Data Obj ->"Please
insert your Crudd Me 3.0 "
->"CD."
|
:0040142E 680A324000 push 0040320A
:00401433 FF3590324000 push dword ptr [00403290]

* Reference To: USER32.SetWindowTextA, Ord:0259h
|
:00401439 E834050000 Call 00401972
:0040143E EB10 jmp 00401450

* Referenced by a (U)nconditional or (C)onditional
Jump at Address:
|:0040142C(C)
|

* Possible StringData Ref from Data Obj ->"Congrats
Level 1 CD-Check Passed."
|
:00401440 682E324000 push 0040322E
:00401445 FF3590324000 push dword ptr [00403290]
```

On voit tout de suite la référence a l'API [KERNEL32.GetDriveTypeA](#), qui demande un disque dans le lecteur (ou plutôt vérifie s'il y en a un...). On sait alors qu'on se trouve au bon endroit. Intéressons-nous aux lignes surlignées en rouge...

Sous le call en 00401424, le CMP compare a 5 le résultat de l'API (5 = présence d'un CD dans le lecteur), et si celui-ci est égal (c'est-a-dire s'il y a en effet un CD) le JE juste en dessous nous emmène en 0040142C. En regardant plus bas, on voit que l'on saute par-dessus le mauvais message, et c'est ensuite du message "**Congrats Level 1 CD-Check Passed.**" qu'il s'agit. Si cette comparaison n'est pas égale, alors on continue tout droit dans le code, directement vers le message "**Please insert your Crudd Me 3.0**".

Le but est de ne pas passer par-la... on va donc modifier ce saut JE de manière à ce qu'il nous emmène quoiqu'il arrive vers le message de félicitations.

La encore, il faut le transformer en JMP (donc on modifie dans l'éditeur hexadécimal le 74 en EB).

Voilà, après sauvegarde des modif', on essaie, et le message d'erreur n'apparaît plus. Il est pas si difficile de broyer du CrackMe quand on sait réfléchir, non?

Conclusion Niveau 1:

On a vu comment modifier du texte et des menus en utilisant ResHacker. Vous verrez qu'on l'utilise beaucoup pour faire des cracks dans un futur lointain, pour récupérer des icônes de programme notamment. Mais aussi pour beaucoup d'autres choses.

Ceci-dit, la partie la plus intéressante est de loin celle où on rend accessible en littéralement 10 secondes un menu qu'on pourrait croire perdu à jamais. J'essaie à travers mes tutos de vous donner une perspective enthousiaste, sachez que quand on veut, on peut. Ne jamais se laisser décourager par une apparente difficulté, ça doit devenir la règle No 1 pour vous: le cracking est un challenge constant. Alors...

Félicitations !