Cracker des Nags-screens

Dans ce cours, nous allons voir comment supprimer des nags sur un Crackme. Ce petit tutoriel s'adresse tout particulièrement aux newbies car la solution est très simple. La méthode est semblable à celle utilisée dans le tuto sur StealthPE, lui aussi disponible sur le forum.
Il est fort probable qu'il existe d'autres solutions hormis celles que j'ai proposées, n'hésitez donc pas à poster les vôtres !

Je rappelle que je ne peux en AUCUN CAS être tenu pour responsable d'un dommage survenant sur votre PC lors de la mise en pratique de ce tuto.

Prérequis :

Vous aurez besoin de connaître le fonctionnement d'OllyDbg, et d'avoir des connnaissances en assembleur, sinon vous n'irez pas très loin.

Voici une introduction à OllyDbg par Crisanar :
http://deamonftp.free.fr/deamoncrack/Tuts/Crisanar/introOlly.htm

Pour l'assembleur, j'ai sélectionné deux cours très bien faits, accessibles aux débutants mais proposant tout de même une approche assez complète.
Cours de Deamon : http://deamonftp.free.fr/deamoncrack/index0.htm
Cours de Falcon : http://xtx.free.fr/liens/tut/Assembleur%20par%20Falcon/Assembleur.html

Normalement vous n'avez besoin de rien de plus.

Outils :
- Le crackme KillNag téléchargeable ici :
http://www.reversing.be/easyfile/file.php?show=20050522170128797
- Peid 0.95 ou RDG Packer Detector 0.6.6 (au choix)
- Un débuggeur/désassembleur : OllyDbg (1.10 ou 2.0)
- Un cerveau :)

Les anciennes versions des logiciels proposés (Peid / RDG) marchent également. Pour Olly, je l'ai fait avec la version 1.10. Tout ces logiciels sont trouvables rapidement dans Google.

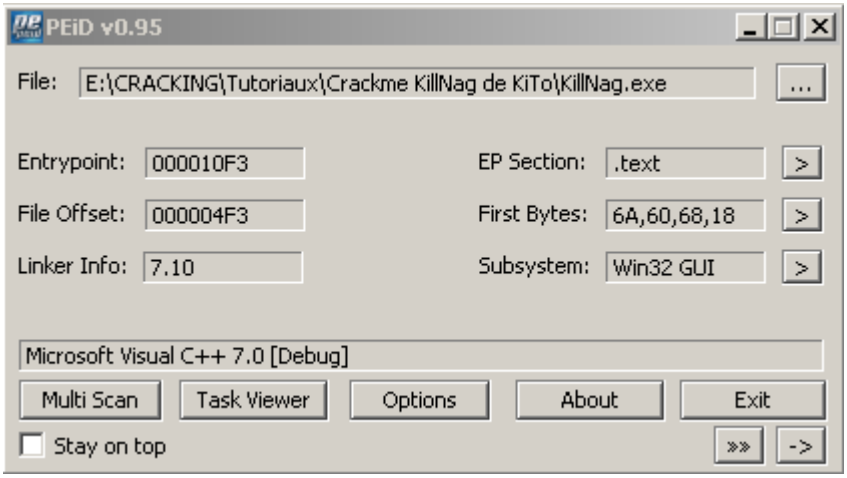
Introduction

Commençons d'abord par observer notre crackme pour repèrer les différentes limitations. Après un rapide coup d'oeil, il en ressort ceci :
- Nagscreen au démarrage
- Nagscreen lors de la sortie du crackme

Voilà, je crois qu'on a fait le tour :). J'espère que vous avez bien noté les messages qu'on a eu avec les messagesbox.

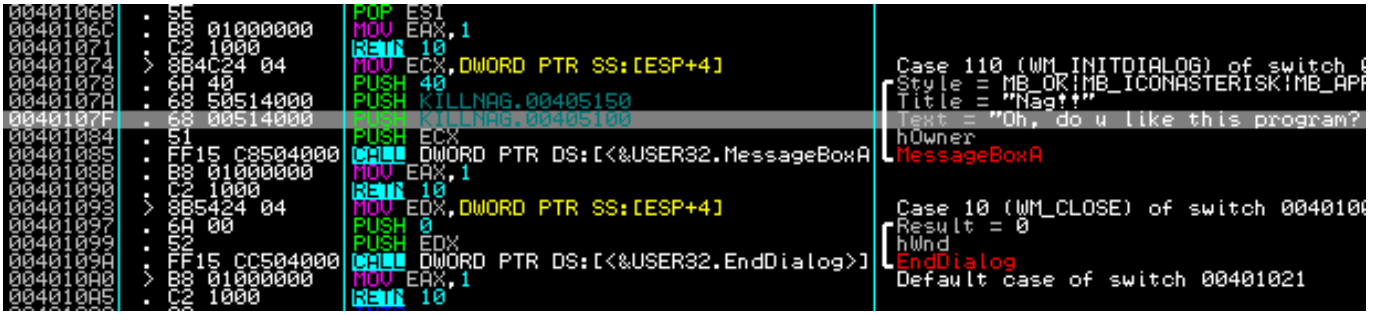
Suppression des Nags

On passe tout d'abord notre crackme dans Peid pour voir si il est packé.



C'est bon, le Crackme est clean. On peut donc continuer.

Chargez le Crackme dans Olly, faites un click droit, Search for, All referenced text strings pour partir à la recherche des messages contenus dans les nags. On a de la chance, ils sont juste au dessus de notre PUSH 60. Double-cliquez sur la string juste au dessus, celle du nag de départ, pour arriver à l'endroit ou se situe le message d'erreur.
Vous arrivez normalement ici :



Examinons voir le code : on observe un CALL à l'API MessageBoxA qui appelle notre Nag en 00401085. Ce CALL est précédé par les éléments nécessaires pour la déclaration de l'API, tel le texte à insérer dans celle ci (la phrase de notre nag). On a deux méthodes pour contourner ce nag. Placez tout d'abord un Bp (F2) sur le MOV ECX,DWORD PTR SS:[ESP+4] en 00401074.

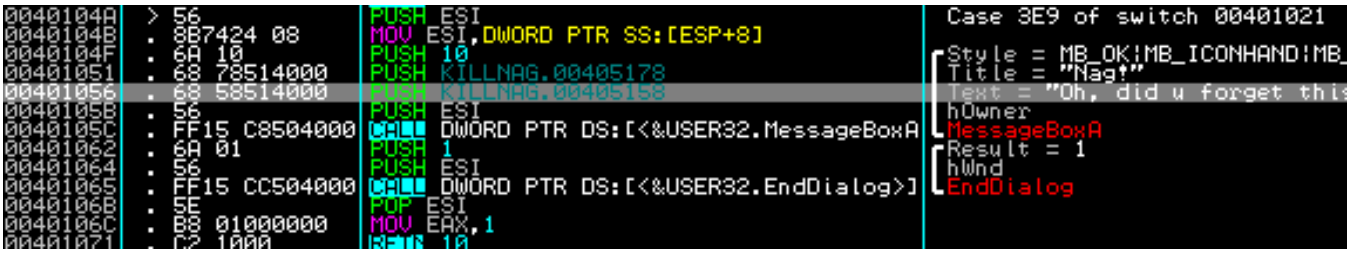
Voyons maintenant la première méthode :
Nous allons ici contourner l'affichage de la messagebox en sautant la procédure qui l'appelle. Faites espace sur le MOV et changez le en JMP 0040108B pour sauter jusqu'à l'instruction après la fin de la routine MessageBoxA.

Faites F9, Olly breake sur notre BP, faites F8 et vous verrez que l'on va esquiver l'affichage du nag et que le programme va se lancer normalement. Et un de fait :).

Voyons maintenant la deuxième méthode :
Nous allons ici contourner l'affichage de la messagebox en effectuant une sortie prématurée de la routine par le biais d'un RETN. On va donc devoir rajouter les instructions qui se trouvent après l'appel à l'API MessageBoxA, pour que le programme continue à tourner normalement.
Modifiez donc le MOV ECX,… en MOV EAX,1 puis rajoutez un RETN à la place du dernier NOP.
L'avantage de cette méthode est qu'elle est un peu plus propre, car je déteste utiliser les NOPs.
Vous n'avez plus qu'à tester, et ça marche encore une fois :)

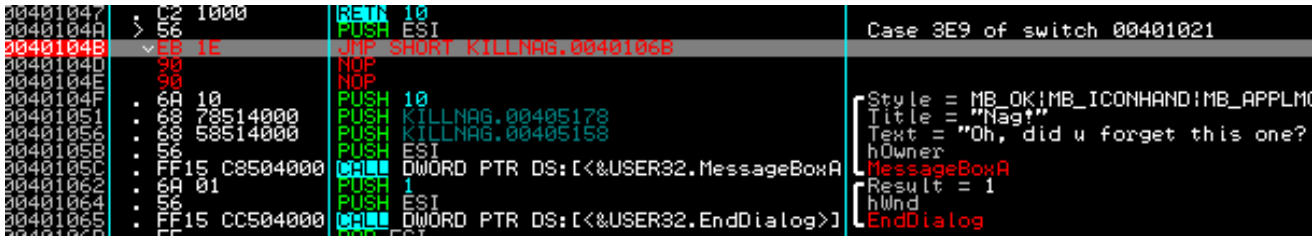
Passons maintenant au deuxième Nag, celui de la fin du programme.

Faites un click droit, Search for, All referenced text strings pour partir à la recherche du message contenu dans le deuxième nag. Il est juste au dessus de la string de notre premier nag.
Double-cliquez sur la string pour arriver à l'endroit ou se situe le message d'erreur.
Vous arrivez normalement ici :



Examinons voir le code : on observe un CALL à l'API MessageBoxA qui appelle notre Nag en 00401085. Ce CALL est précédé par les éléments nécessaires pour la déclaration de l'API, tel le texte à insérer dans celle ci (la phrase de notre nag).
Or, la procédure d'appel de notre nag est suivie de la procédure EndDialog qui met fin à la boîte de dialogue ouverte, c'est à dire à celle qui nous fait ici sortir définitivement du programme.
On a qu'une seule méthode (ici :) pour contourner ce nag. Placez tout d'abord un Bp (F2) sur le MOV ESI,DWORD PTR SS:[ESP+8] en 00401074.

Nous allons ici directement sauter sur la procédure de clotûre du Crackme. Faites espace sur le MOV et changez le en JMP 0040104B pour sauter jusqu'à l'instruction qui marque le début de la routine EndDialog.



Faites F9, Olly breake sur notre BP, faites F8 et vous verrez que l'on va esquiver l'affichage du nag et que le programme va se quitter normalement. Et deux de faits :).

Pour maintenant enregistrer les changements, faites Click droit => Copy to executable => All Modifications => Copy All. Dans la fenêtre qui s'ouvre, faites Click droit => Save File.
Votre fichier cracké est maintenant enregistré ;). C'est quand même plus rapide qu'avec un éditeur hexadécimal, non ?

Ce tutoriel est fini. J'espère qu'il a été clair, et que vous n'avez pas rencontré de difficultés.
Tuto finalisé par Horgh le 03/03/2010

Merci à KiTo pour son crackme

P.S. : J'ai remarqué après avoir fini le tuto que thorpe en avait déjà fait un en anglais, que l'on peut consulter ici :
http://www.reversing.be/article.php?story=20050526043821808&query=killnag

Il propose deux autres méthodes intéressantes car la manière de procéder est différente, mais que j'avoue ne pas trop apprécier à cause des NOPs.
Jettez y quand même un œil :)