Cours de Cracking

(6ième Partie)

Mon objectif : vous apprendre a vous servir de Softice et meme à cracker avec (tant ka faire ;).

1/ Les logiciels utiles pour ce cours

-> Le programme à craquer : **Start Clean v1.2**
-> Le débugger : **Softice 4.05**

2/ Qu'est ce que SoftIce ?

SoftIce est un debuggeur. Il vous permet d'executer un programme pas a pas pour en trouver les erreurs... enfin, ca c'est la mainiere officiel de s'en servir :). Parceque nous, les erreurs, on s'en fout ! Ce qu'on veut, c'est de faire tourner un programme pas a paspour trouver l'endroit d'ou on appelle un nagscreen ou mieux encore, trouver l'endroit ou on compare un serial avec le bon (et donc chopper le bon!)

Vu comme ca, ca a l'air facile non ? En fait, les problemes viennent plutot de l'utilisation du debuggueur et de l'analyse du code Assembleur qu'il nous affiche ! Ah Ah !! Oui, ici vous pouvez oublier les beau "string data ref" et tout ce qui s'apparente de pres ou de loin a du texte !!! SoftIce, c'est du 100% ASM ! Avec des 'eax' 'ebx' '40000:02245' 'mov eax, [ebp-14]' etc....

C'est donc le but de se tutoriel de vous apprendre a décortiquer tout ca ! :) Et c'est parti !!

3/ Installer softice

Bon, ne mettons pas la charrue avant les boeufs, et commencons deja par installer SoftIce correctement.

-> lorsqu'on vous demande un serial...ben vous avez ka mettre **1900-0000DD-9B** (v3.25)
-> Installez-le dans le répertoire par defaut...
-> choisissez tous les composants...
-> si vous trouvez pas votre carte video, mettez "Standard VGA" et cochez "Universal Video driver"...
-> choisissez si votre souris est sur COM1, COM2 ou PS2...
-> laissez l'installation modifiez votre autoexec.bat (pour windows 95/98) ...
-> faite "Register Later" puis "Yes, restart my computer now".

Vala, maintenant SoftIce est installé !! ;)
Ok,maintenant il faut que vous modifiez quelque parametres...

-> éditez le fichier winice.dat qui se trouve dans le répertoire ou vous avez installé SoftIce
-> C:\WINDOWS\system32\drivers sous XP
-> *C:\program files\numega\softice95 par défaut*
L'édition de winice.dat se fait avec la commande "ouvrir" dans notepad.exe.
-> enlevez les **';'** devant les dll suivantes : **kernel32.dll, user32.dll, gdi32.dll**.
-> recherchez aussi une ligne commencant par ligne *INIT=* et remplacez là par ceci:
INIT="X;CODE ON;DATA;R;". => (pifoman : cette ligne va demander l'affichage de la zone de commande,de la zone de code, de la zone de données et de la zone de registres à chaque lancement de softice)
-> pour que les modifications prennent effet il faut redémarrer.

3/ Comment est-ce qu'on se sert de softice ?

Vous avez redemarré ?

Et maintenant, vous vous dites : bon, comment on lance SoftIce ?

-> réponse: on le lance pas, car vous etes deja dedans ! En fait, c'est comme dans Alien IV :)) Vous etes en apparence dans windows, mais au fond, ya SoftIce qui veille...Et dès que vous tapez Ctrl+D (ou F5), vous etes dans SoftIce !

Et là, vous allez me dire "mais a quoi ca sert le Symbol Loader qu'il m'on mis en raccourci ?

-> et ben ca, c'est si vous voulez tracer un programme depuis sa premiere instruction ! Autant dire que ca sert pas a grand chose...du moins pas souvent :) Passke, entre le debut d'un programme et son nagscreen, peut bien avoir des milliers d'instruction... imaginez si faisiez chacune de ces instruction pas a pas !

Le problème du debuggeur, c'est que quand vous entrez dedans, vous pouvez etre n'importe ou !!

Explications : windows, ca fait plein plein de truc en tache de fond...(vous imaginez meme pas...) et comme le debuggueur il sait pas que vous vous interressez qu'a tel ou tel programme, et ben il vous affiche les instructions en cours au moment ou vous l'appelez.

Oauis...vous avez l'air perdu... En gros, des que vous faites Ctrl+D, vous tombez n'importe ou dans windows !! (Meme si vous faites Ctrl+D alors que vous etes dans le programme a cracker..). Il va donc falloir trouver une ruse pour atterir ou on veut ! Et c'est ce qu'on appelle les "breakpoints" (= "point d'arret" = *bpx*). Ca consiste a dire a SoftIce "Arrete toi a tel endroit". Et alors je vous entend d'ici me crier :

4/ Comment est-ce qu'on sait à quel endroit il faut s'arrêter ?

Ahhh...En fait on va dire au debuggeur "arrête toi des qu'il y a une fenêtre de crée".C'est alors qu'intervient les fonctions usuelles de windows... Par exemple, quand un programme crée un fenêtre, il utilise souvent la fonction "CreateWindowExA" et "ShowWindow"...

Donc si on dit a SoftIce **"**Arretes toi des que la fonction CreateWindowExA intervient**"**, et bien il nous arrête dès que la création de la fenêtre est appelée. Donc on a l'appelant, et on peut empêcher qu'il appelle le nagscreen...vous suivez ??

Bon, en technque, pour poser un tel point d'arret, il suffit de rentrer dans SoftIce, puis de taper :

bpx CreateWindowExA

5/ Quelques fonctions souvent utilisées

Notes toutes les fonctions ayant un A la fin signifie que ce sont des fonctions 32 bits.
Pour le mode 16 bits enlever simplement le A...

Exemple : GetWindowTextA = GetWindowText

Lecture/Ecriture de fichier :	ReadFile WriteFile CreateFileA
Lecture de données d'un fichier ini :	GetPrivateProfileStringA GetPrivateProfileIntA WritePrivateProfileStringA WritePrivateProfileIntA
Accès à la base de registre:	RegCreateKeyA RegDeleteKeyA RegQueryValueA RegCloseKeyA RegOpenKeyA
Boîtes de dialogues:	GetWindowTextA GetWindowTextW GetDlgItemTextA GetDlgItemTextW GetDlgItemInt

Boite de messages:	MessageBox MessageBoxA MessageBoxExA MessageBeep
Date et heure :	GetLocalTime GetSystemTime GetFileTime
Creation d'une fenêtre :	CreateWindowExA ShowWindow
Fonctions utiles pour les programme en Visual Basic :	Hmempcy MultiByteToWideChar (comparaison de deux chaines)
Accès au CD-ROM (pour les jeux sur CD)	GetDriveType

Don't Panic !! Si j'enumere ces fonctions (liste non exhaustive..) c'est juste pour vous montrer quelque exemples...dans les autres parties du cours, vous apprendrez a vous en servir :)

6/ Commandes Principales de SoftIce

C'est pas tout, mais faut bien que vous sachiez utiliser un peu SoftIce...Donc voici une liste des principales commandes de SoftIce...

F8 = permet d'éxécuter le programme pas à pas tout en rentrant dans les CALL (c'est à dire que le programme appelle une fonction, ou une routine de vérification du serial par exemple...).
Exemple : CALL 000012345 => si ici on fait F8 on rentre dans la fonction.

F10 = la meme chose que F8 mais ne rentre pas dans les CALL : il les éxécutent, vous n'avez simplement pas le détail de la fonctions). Si on fait F10 on éxécute le CALL mais on ne rentre pas dedans on va directement a l'instruction suivante...

La nuance entre les deux est tres importantes: Imaginez qu'un call est une porte donnant sur une piece ayant elle meme d'autre porte, et ainsi de suite a n'en plus finir...Et ben, si vous rentrez dans une porte (un CALL) puis, a partir de cette porte, vous entrez dans une autre, et une autre, et encore une autre.... ben vous vous etes plus qu'eloigner de l'origine... :) c'est pour ca que F8 est a utiliser avec moderations, et il faut eviter de trop s'enfoncer de call en call...

F12 = permet de sortir d'un CALL et de reprendre l'éxécution juste après. C'est comme ca que vous retrouverez l'appelant d'une fonction.(vous etes a l'endroit X, appuyez sur F12 et vous arriverez juste apres l'endroit Y qui appelle X...)

"? nom_de_registre" = permet d'évaluer une valeur d'un registre *en décimal*
Exemple si eax = 00003039 faites *"? eax"* et vous obtiendrez : "12345"

? = aide, très utile, vous y trouverez toutes les fonction de SoftIce...

r = pour modifier la valeur d'un registre. Exemple si vous voulez que eax soit égal à 1 faites *"r eax=1"*
Attention : les valeurs contenues dans les registres sont des valeurs héxadécimale.

bpx nom_de_fonction = pour creer un breakpoint sur une fonction (ex: bpx showwindow).
bpm adresse_memoire = pour creer un breakpoint sur une addresse memoire (ex: bpm 0040660).

bc * = supprimmer tous les breakpoints, car quand vous en poser un, il reste jusqu'a ce que vous l'effaciez....

Exit = forcer SoftIce à quitter le programme (pratique en cas plantage).

Task = permet de savoir sous quel nom tourne un programme précis.

HWND nom_du_programme = Pour connaître le différents sous objets d'une application.(fenêtre, boite de dialogue...)

CTRL D ou F5 = Rentrer et sortir de SoftIce...une des touches les plus utile :)

Voila, vous savez le principal sur l'utilisation de SoftIce... N'hésitez pas a revenir sur cette partie du tutorial pour revoir les commandes et les fonctions utilisées dans SoftIce... Allez, en cadeau bonux, je vous offre un beau dessin d'une fenêtre SoftIce, histoire que vous voyez a quoi ca ressemble :) Ouais, je sais, ca parait austère vu comme ca, mais en fait c'est bien pratique :)