

## TUTO N° 2

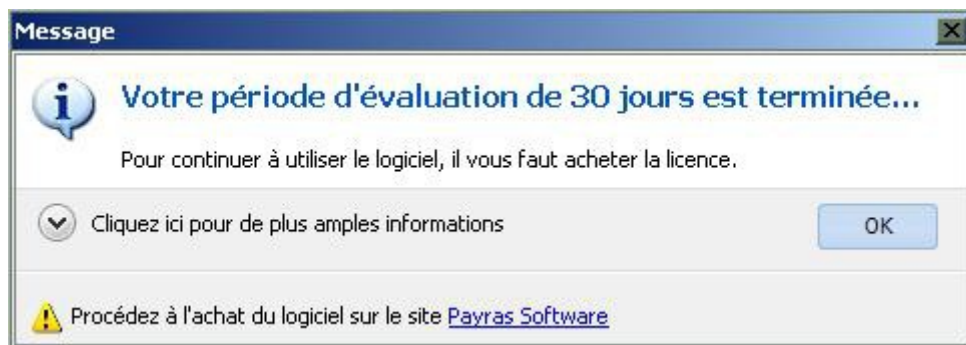
### Cracker KenoExpert Version 7.090.

Plusieurs versions de ce logiciel sont sorties depuis mon tutoriel ( lien [ICI](#).)qui est devenu obsolète pour les versions inférieures à la 7.0. Aussi ayant voulu voir ce que qui a changé, c'est pour moi l'occasion d'écrire une suite.

Le principe reste le même : basculer d'une version d'évaluation à une version enregistrée. Seules les méthodes utilisées pour y parvenir changent, celles du premier tuto ne fonctionnant plus sur cette version.

En effet si l'on cherche les strings "version d'évaluation" et "version pro enregistrée" dans IDA, nous les trouvons effectivement ainsi que les adresses correspondantes. Si nous patchons comme dans le premier tuto, cela ne marche tout simplement plus.

J'ai donc cherché un nouveau point de départ, et pour les besoins de la cause, j'ai avancé volontairement l'horloge de mon PC de quelques semaines, lancé KenoExpert, et suis revenu à la date d'aujourd'hui. C'est pour ça que désormais nous arrivons à ce message quand nous lançons le soft, et non pas parce que j'ai attendu trop longtemps pour faire ce cours :



Et nous allons partir de ce nag pour arriver à nos fins.

Ainsi nous utiliserons encore la méthode de la pile, mais la recherche de strings sera remplacée par des recherches en mémoire.

Mais avant, un petit détail à régler : le nag de lancement qui nous pourrit la vie en masquant une partie de l'écran de Olly.

A l'époque, c'est Kirjo qui m'avait fait penser à supprimer ce nag. Il me disait :

« Autre petit truc, si tu veux tracer tranquillement le début du soft sans le nag d'accueil en plein milieu, tu peux virer ce nag en remplaçant :

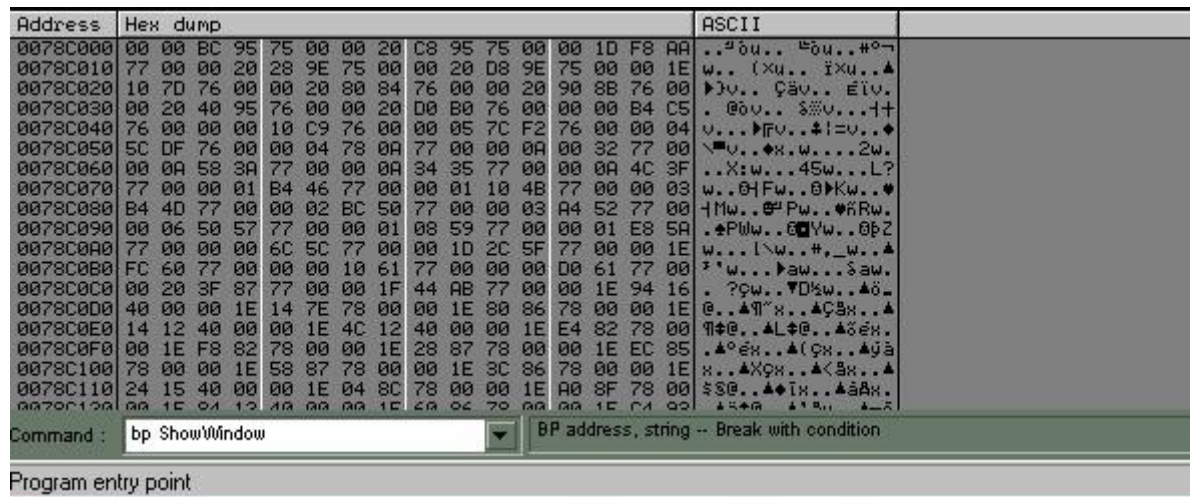
Code : 0040A1BB      MOV EBX, EAX      par :    JMP 0040A1D8 »



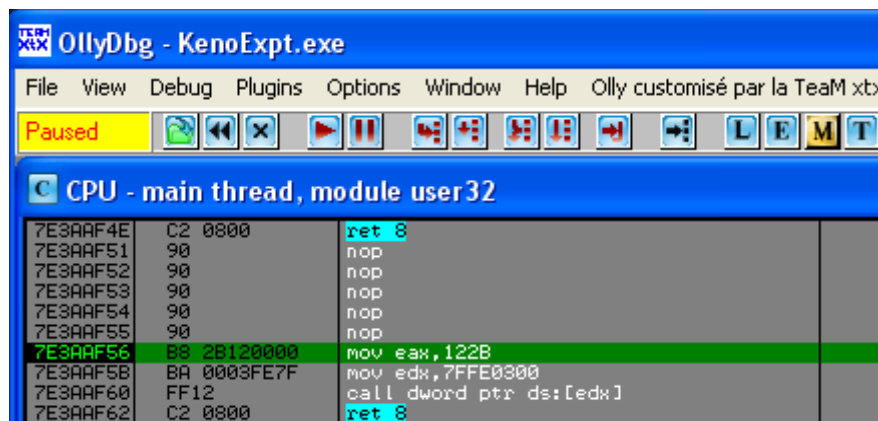
Il est bien évident que cette adresse n'est plus d'actualité, et je vous propose une solution inspirée par notre ami Burner pour la retrouver, en utilisant l'API ShowWindow.

Nous chargeons notre cible dans Olly (j'utilise encore ici Olly 1,10).

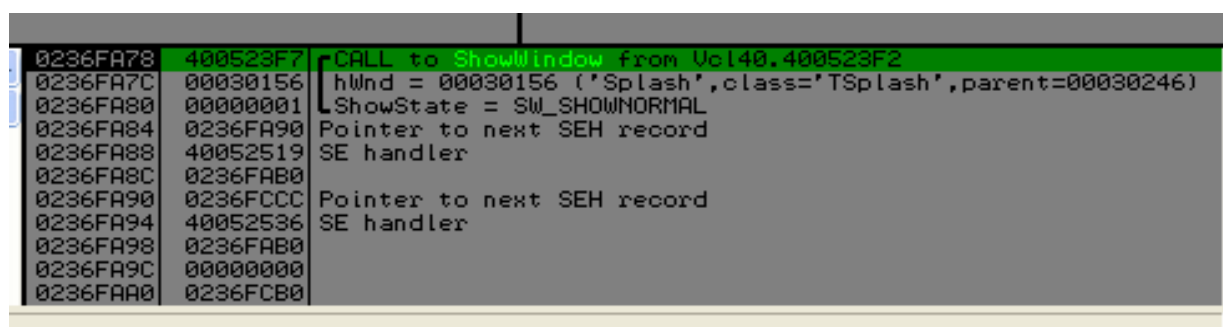
Il faut maintenant poser un BP sur l'API ShowWindow en saisissant la commande bp ShowWindow dans la fenêtre de commande en bas à gauche (en respectant la casse : majuscules/minuscules) et lancer la cible.



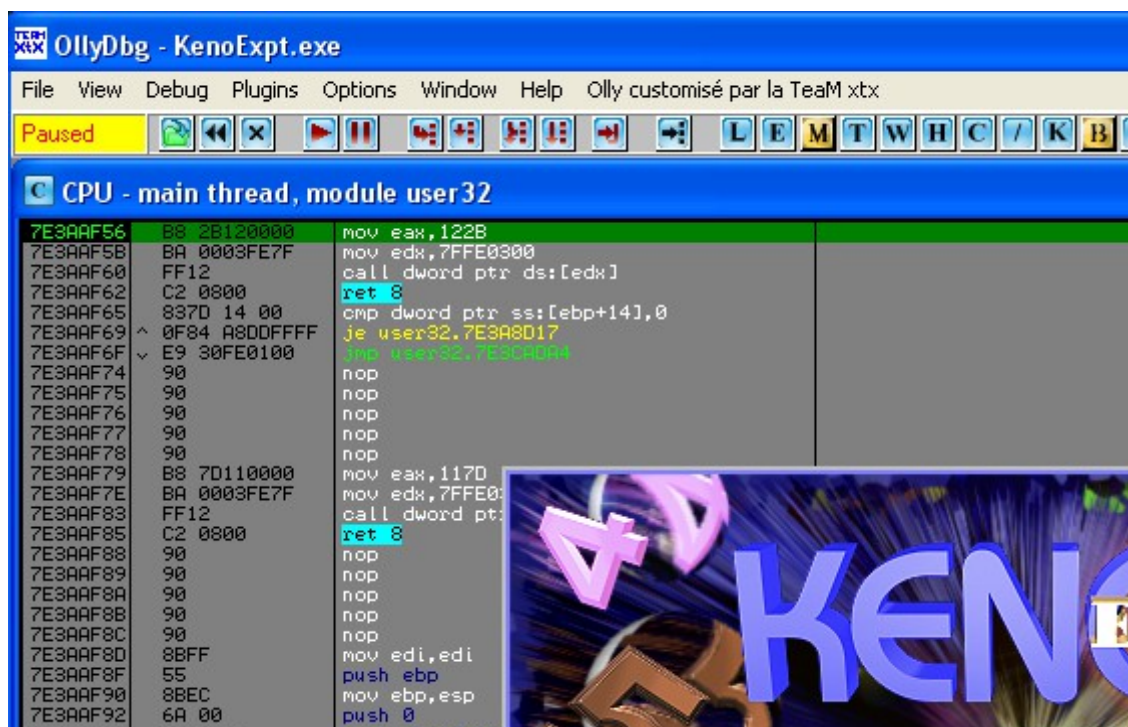
Olly breake aussitôt à cette adresse, mais dans le module user32 :



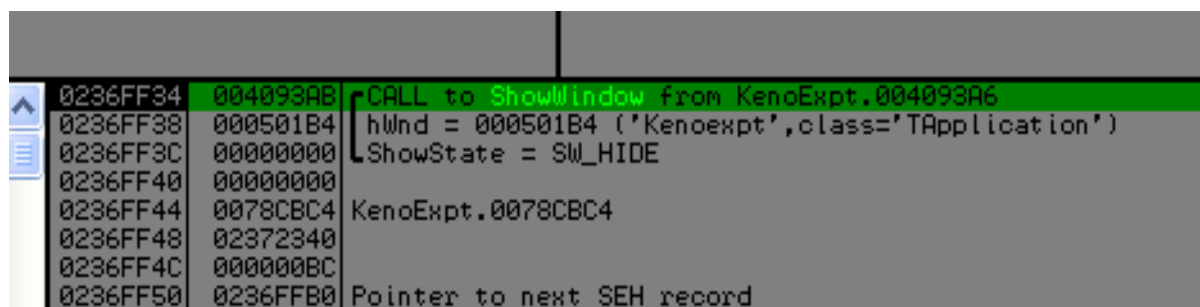
Et nous voyons ceci dans la pile :



Call to ShowWindow from Vcl40.400523F2 : Pas bon ! Car à première vue ça ne nous intéresse pas de patcher dans Vcl40, donc F9 pour voir si nous avons autre chose :

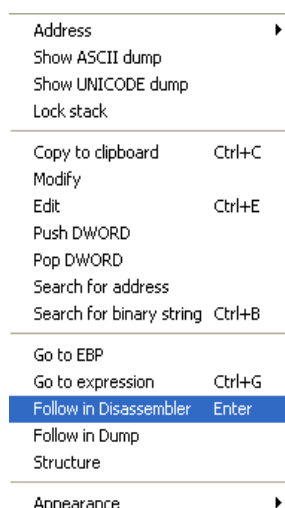


Olly breake dans le module user32 au moment ou apparaît notre nag. Pas de doute, nous sommes au bon endroit, et la pile nous le confirme :



Là nous sommes dans le module KenoExpt ! Y A BON !

Clic droit sur la ligne Call to ShowWindow from KenoExpt . 004093A6, puis Follow in Disassembler



Address	Disassembly	Comment
0040934F	33C0	xor eax,eax
00409351	8B55 C8	mov edx,dword ptr ss:[ebp-38]
00409354	64:8915 0000	mov dword ptr fs:[0],edx
0040935B	E9 0E020000	jmp KenoExpt.0040935E
00409360	8B0D 6CCAF20	mov ecx,dword ptr ds:[<&Ucl40.@Forms@Application>]
00409366	8B01	mov eax,dword ptr ds:[ecx]
00409368	E8 0F013800	call <jmp.&Ucl40.@Forms@TApplication@Initialize\$qqr>
0040936D	33C9	xor ecx,ecx
0040936F	B2 01	mov dl,1
00409371	A1 54607E00	mov eax,dword ptr ds:[7E6054]
00409376	E8 8DB61500	call KenoExpt.00564A08
0040937B	8B08	mov ebx,eax
0040937D	A1 20A07F00	mov eax,dword ptr ds:[7FA020]
00409382	8918	mov dword ptr ds:[eax],ebx
00409384	8BC3	mov eax,ebx
00409386	E8 5D013800	call <jmp.&Ucl40.@Forms@TCustomForm@Show\$qqr>
0040938B	8B15 20A07F00	mov edx,dword ptr ds:[7FA020]
00409391	8B02	mov eax,dword ptr ds:[edx]
00409393	8B10	mov edx,dword ptr ds:[eax]
00409395	FF52 7C	call dword ptr ds:[edx+7C]
00409398	6A 00	push 0
0040939A	8B0D 6CCAF20	mov ecx,dword ptr ds:[<&Ucl40.@Forms@Application>]
004093A0	8B01	mov eax,dword ptr ds:[ecx]
004093A2	8B50 24	mov edx,dword ptr ds:[eax+24]
004093A5	52	push edx
004093A6	E8 2B213800	call <jmp.&USER32.ShowWindow>
004093AB	66:C745 D8 20	mov word ptr ss:[ebp-20],20
004093B1	BA 44CE7800	mov edx,KenoExpt.0078CE44
004093B6	8D45 F8	lea eax,dword ptr ss:[ebp-8]

et nous arrivons sur cette portion de code très intéressante :

Si nous remontons un peu, nous avons un call en 00409376. Si nous y posons un BP, et que nous relançons le soft dans Olly, nous allons breaker dessus. Quelques F7 pour constater que le nag se fabrique ici, et nous arrivons à la ligne suivante, en 0040937B, le fameux mov ebx, eax évoqué par Kirjo, avec lequel le soft charge définitivement ce nag. C'est donc bien ici qu'il faut opérer. Vous pouvez toujours essayer de patcher ailleurs, bravo à vous si vous réussissez et merci de m'indiquer comment vous avez fait...

Bon, à présent, nous savons où il faut patcher, mais pour aller où ?

Nous redescendons de quelques lignes en 004093A6 ou nous apercevons un call ShowWindow. C'est là que nous passerions si nous ne patchions pas et le nag apparaîtrait.

Mais 2 lignes plus bas, en 004093B1, nous tombons sur un mov edx, KenoExpt qui pourrait bien nous emmener dans le soft après le nag (regardez le commentaire qui semble confirmer : keno expert 7. 0)

Donc on remonte à la ligne 0040937B, clic droit – Assemble - remplacer mov ebx,eax par JMP 004093B1- et bouton Assemble . Sauvegarder le fichier comme vous avez appris à le faire dans le premier tuto, et vous ne serez plus gêné par ce nag.

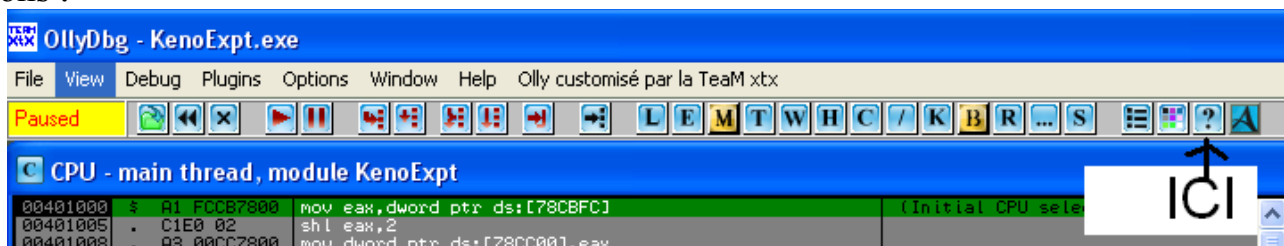
Il existe une autre manière de procéder pour supprimer ce nag , toujours en utilisant l'API ShowWindow.

Relancer la cible dans Olly, mais au lieu de se servir de la fenêtre de commande, comme dans la première méthode, nous allons chercher le nom de cet API dans la fenêtre des labels en tapant CTRL+N. Oui je sais, il n'y a pas de bouton de raccourci pour cette demande.

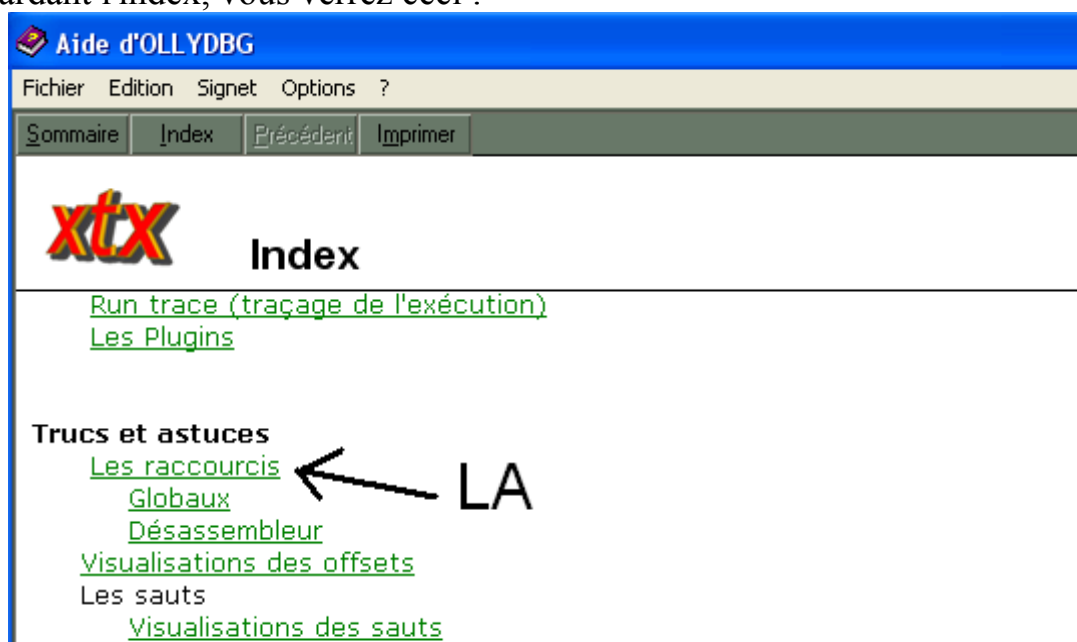
C'est l'occasion de rappeler la présence de l'aide d'Olly que chacun devrait penser à consulter avant et pendant l'utilisation de ce débogueur.

Les captures qui suivent vont faire sourire les plus aguerris des reversers, mais je souffre d'entendre des questions de la part de débutants qui bloquent sur le fonctionnement de Olly, alors qu'il leur suffirait d'appuyer sur un bouton.

Pour y accéder , vous trouverez donc ce bouton orné d'un point d'interrogation dans la barre des boutons :

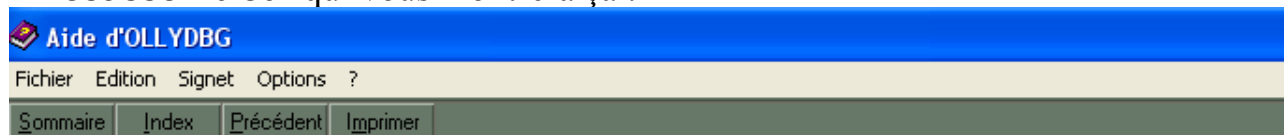


En regardant l'index, vous verrez ceci :





Choix Désassembleur qui vous montrera ça :



## Désassembleur

pouvoir utiliser ce dispositif.

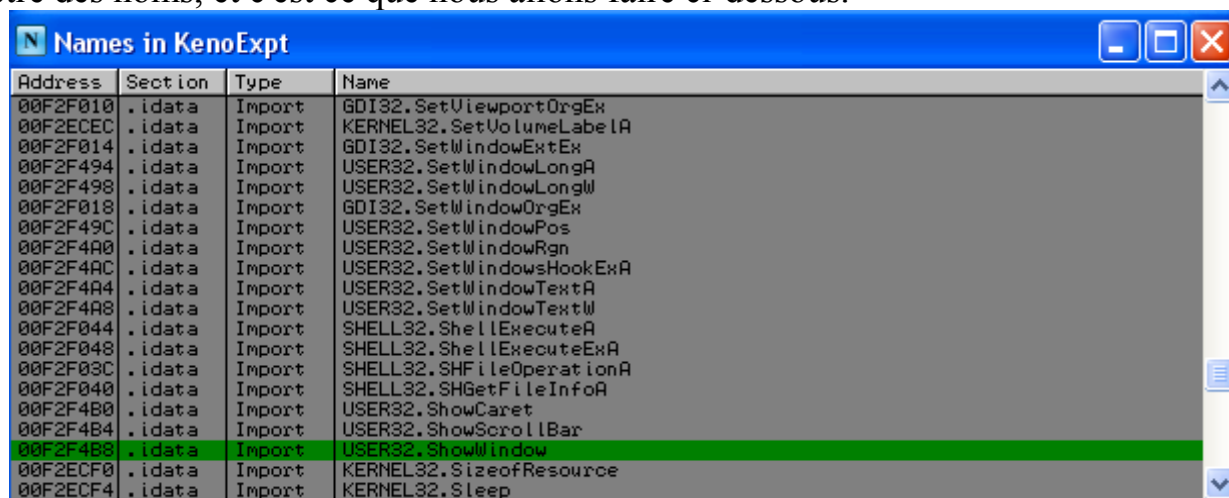
**Ctrl+K** - Afficher l'arborescence d'appel associés au processus actuel. Vous devez avoir analysé le co avant de pouvoir utiliser ce dispositif.

**Ctrl+L** - Recherche suivante, répète la dernière recherche.

**Ctrl+N** - Ouvrir la liste des noms (labels) dans le module actuel. ← **CE RACCOURCI LA**

**CTRL+O** - Scanner les fichiers objets. Cette commande affiche la "dialogue" de scan des fichiers obje vous pouvez les fichiers objets du module actuel et trouver toutes les références ou bibliothèques et scanner dans le but de trouver les modules d'objet dans la section de code actuel.

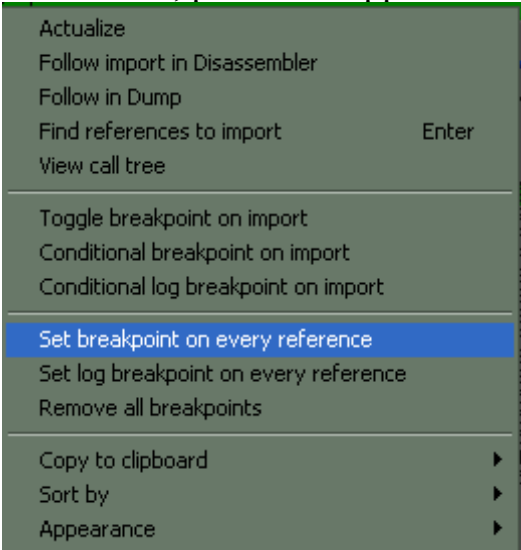
Maintenant que nous avons consulté l'aide, nous savons que le raccourci CTRL+N nous ouvre la fenêtre des noms, et c'est ce que nous allons faire ci-dessous.



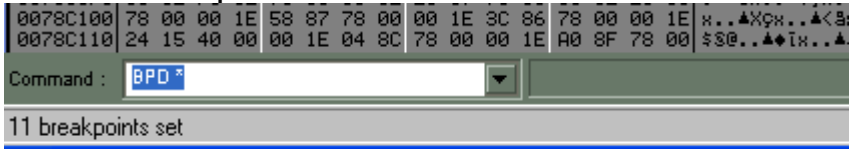
Les APIS sont classées par ordre alphabétique dans cette fenêtre, ce qui n'est pas le cas des modules auxquels elles appartiennent. C'est ainsi que voyons que ShowWindow appartient au module USER32, qui est classé avant KERNEL32 et après SHELL32.

Donc faire la recherche sur le nom de l'API plutôt que sur le nom de son module.

Mettre la ligne en surbrillance et clic droit, pour faire apparaître cette fenêtre :



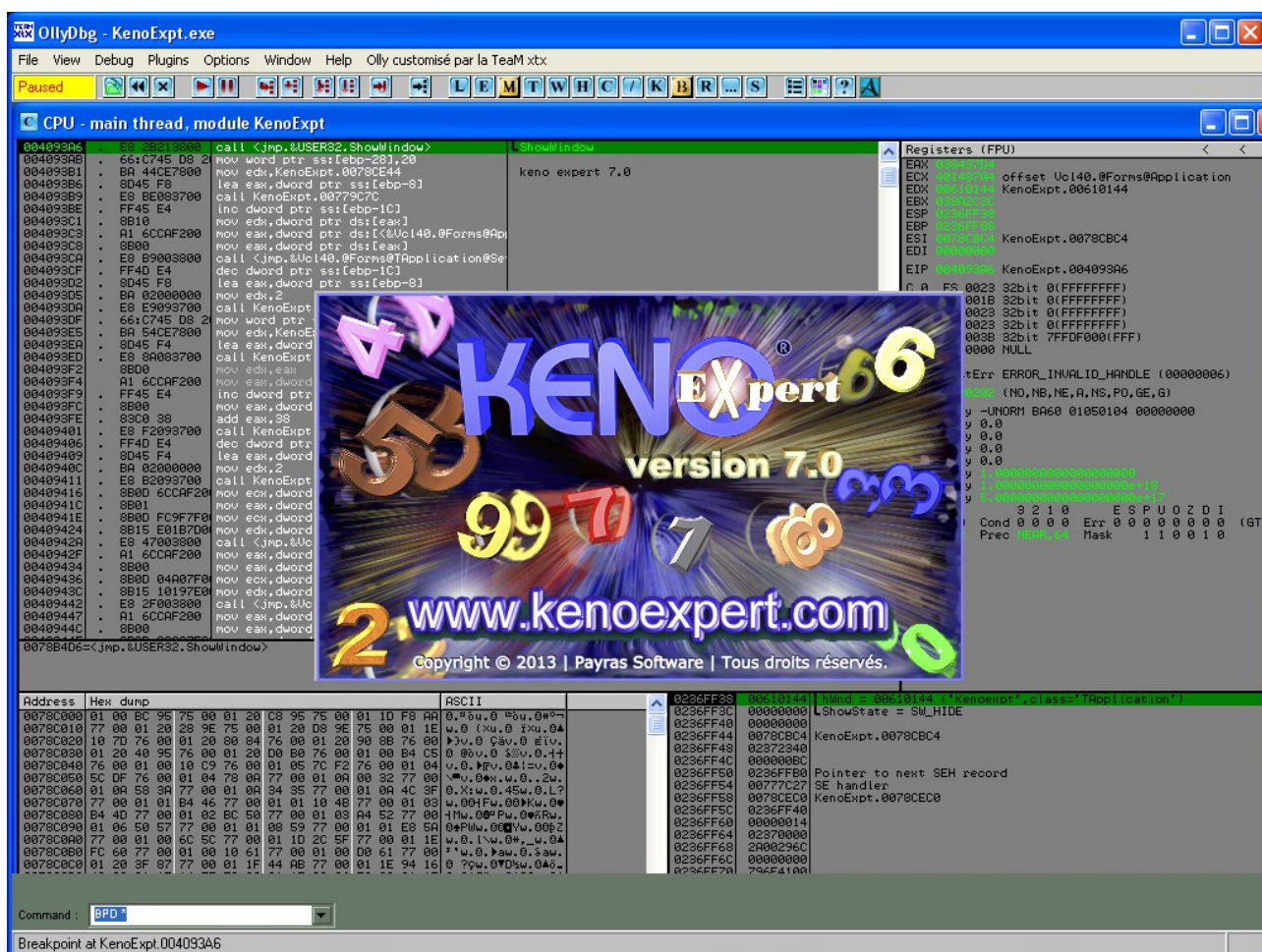
clic gauche sur cette ligne et l'information suivante apparaît en bas à gauche de Olly, sous la fenêtre de commandes : 11 breakpoints set



que nous retrouvons en détail dans la fenêtre des breakpoints en appuyant sur le bouton B.

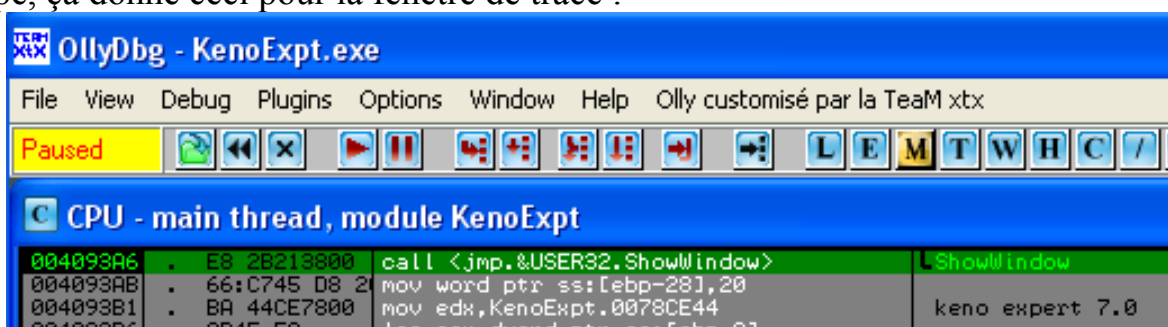
B Breakpoints				
Address	Module	Active	Disassembly	Comment
004093A6	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
004094E4	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
0055F9EC	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
0055FA42	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
0058EA92	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
005CBA2F	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
005CBE16	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
0060EC88	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
006E7104	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
007066BD	KenoExpt	Always	call < jmp. &USER32.ShowWindow>	
0078B4D6	KenoExpt	Always	jmp dword ptr ds:[&USER32.ShowWind	

Nous pouvons à présent lancer la cible dans Olly, qui breake aussitôt ici, en même temps que s'affiche notre nag :



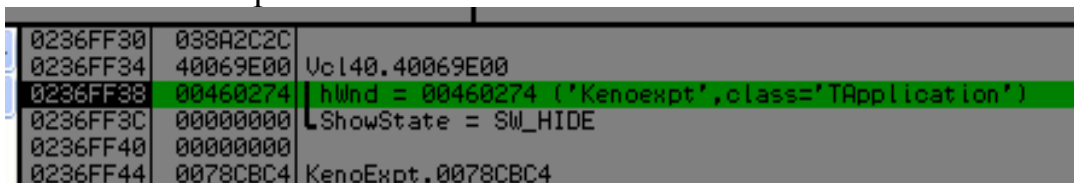
Nous allons faire un petit tour de ce que nous pouvons observer à partir de cette fenêtre :

A la loupe, ça donne ceci pour la fenêtre de trace :



et ceci pour la fenêtre de la pile (vous obtiendrez certainement quelque chose de différent chez vous, mais ça n'a pas d'incidence sur la suite) :

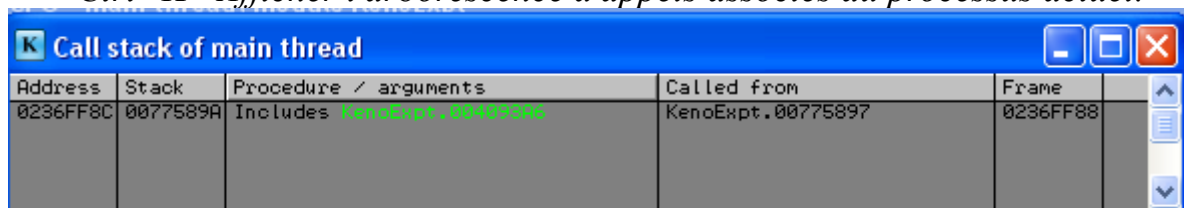
Ces informations ne sont pas vraiment intéressantes pour ce que nous avons à faire, mais nous faisons un petit tour n'est ce pas ?





En appuyant sur le bouton K , nous faisons apparaître cette fenêtre, qui d'après l'aide de Olly, nous montre :

*Ctrl+K - Afficher l'arborescence d'appels associés au processus actuel.*



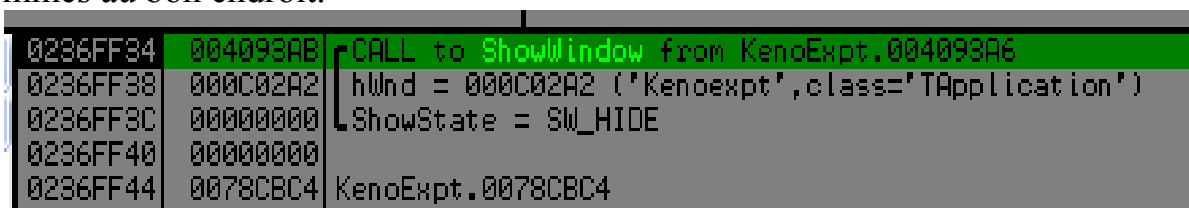
Address	Stack	Procedure / arguments	Called from	Frame
0236FF8C	0077589A	Includes KenoExpt.004093A6	KenoExpt.00775897	0236FF88

Nous retrouvons bien notre adresse de la fenêtre de trace 004093A6, avec son appel.

Voilà pour le petit tour.

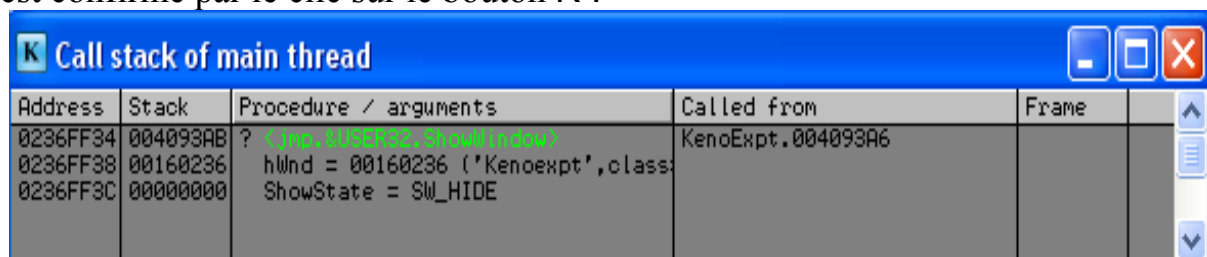
Alors, une deuxième fois F9 nous amène ici :

Nous sommes au bon endroit.



0236FF34	004093AB	CALL to ShowWindow from KenoExpt.004093A6
0236FF38	000C02A2	hWnd = 000C02A2 ('Kenoexpt',class='TApplication')
0236FF3C	00000000	ShowState = SW_HIDE
0236FF40	00000000	
0236FF44	0078CBC4	KenoExpt.0078CBC4

Ce qui est confirmé par le clic sur le bouton K :

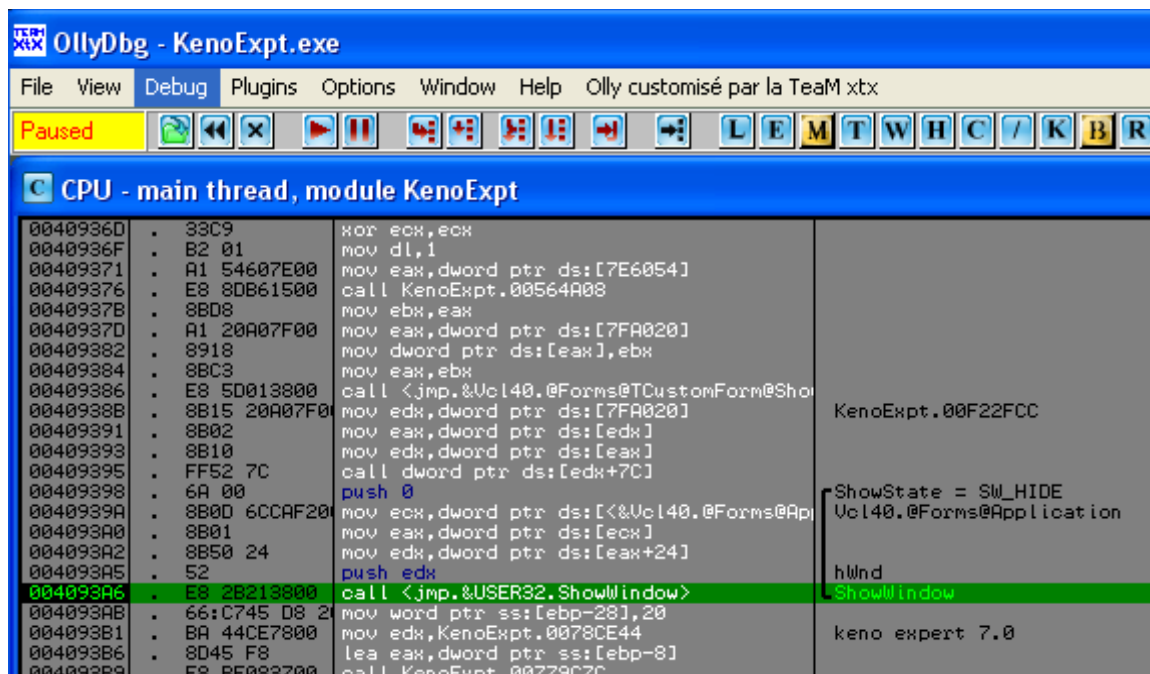


Address	Stack	Procedure / arguments	Called from	Frame
0236FF34	004093AB	? (<jmp.&USER32.ShowWindow>)	KenoExpt.004093A6	
0236FF38	00160236	hWnd = 00160236 ('Kenoexpt',class='TApplication')		
0236FF3C	00000000	ShowState = SW_HIDE		

Si nous cliquons droit sur la ligne CALL to ShowWindow de la pile, choisissons **Follow in Disassembler**, je suis sûr que vous reconnaissez maintenant le bout de code où Olly a breaké, que j'avais qualifié de très intéressant dans la première méthode :

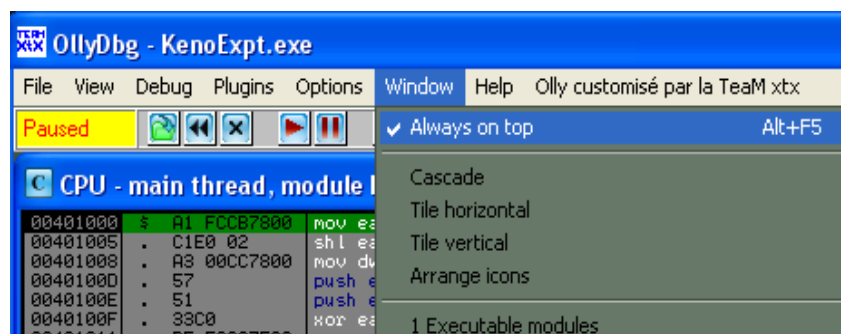
avec son `mov eax,ebx` en 0040937B

son `mov edx,KenoExpt.0078CE44` en 004093B1



A vous de patcher...

**BONUS :** Papa Bango m' a aussi donné une astuce quand la fenêtre de Olly est en partie masquée par ce genre de désagrément. Il s'agit d'une option de Olly :

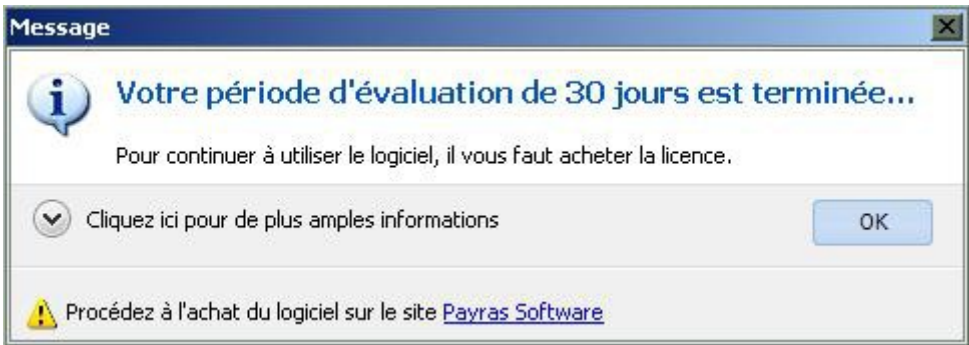


Si vous avez coché cette option pour l'essayer, décochez la pour la suite du cours, sans quoi vous risquez de ne plus voir passer les messages d'erreur du soft sans être obligé de réduire la fenêtre de Olly.

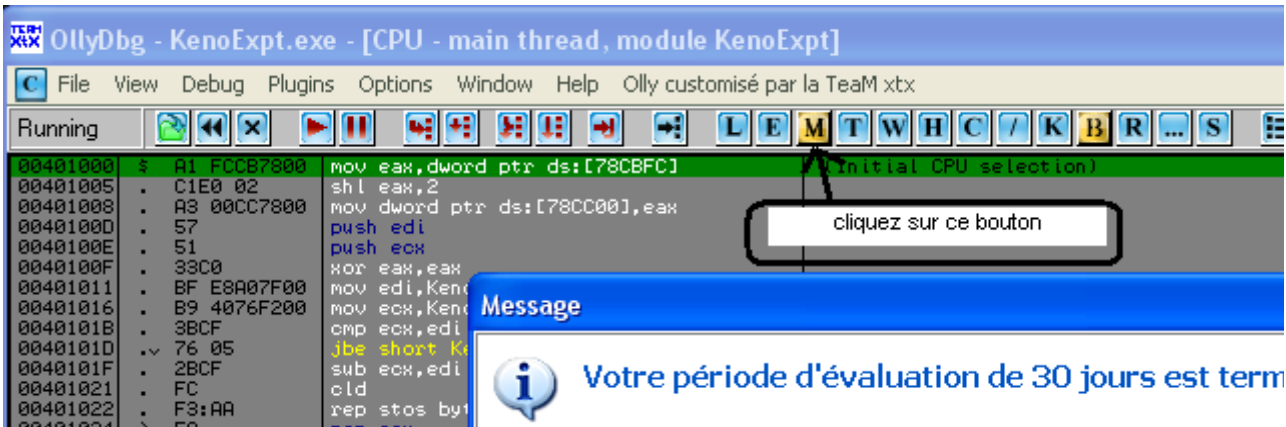
A présent que le problème du nag est résolu, si nous relançons le soft, nous arrivons à nouveau à la fenêtre qui informe que notre période d'évaluation de 30 jours est expirée.

Le moment est venu de traiter l'objet initial de ce tuto.

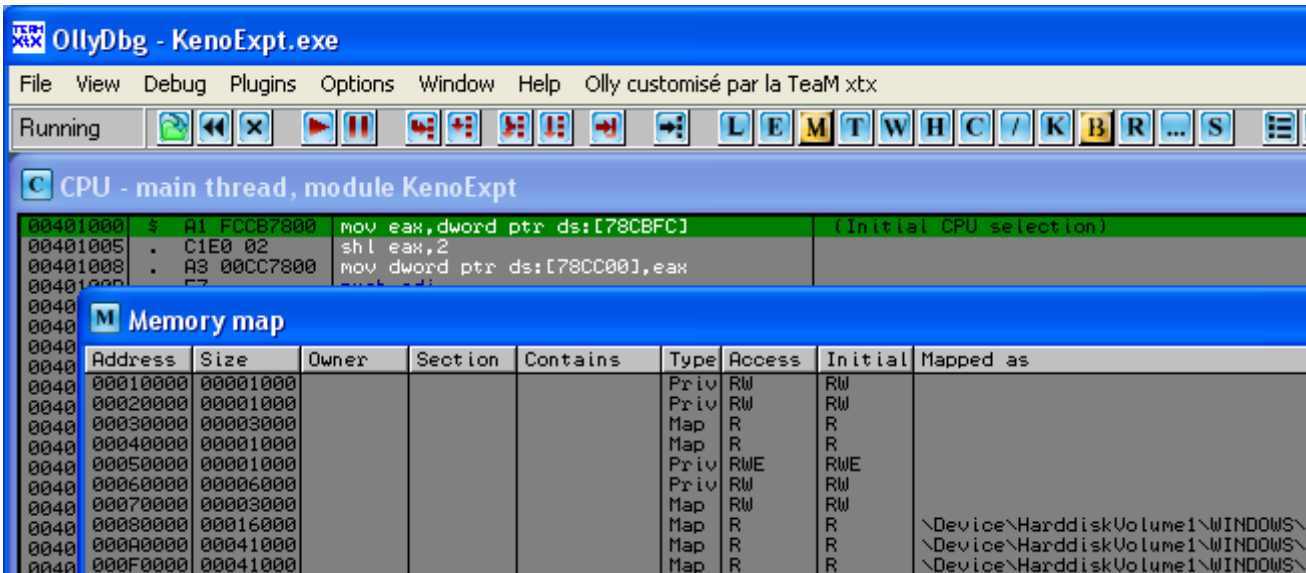
Rappelez vous que nous devons partir de cette fenêtre pour rendre notre cible fonctionnelle :



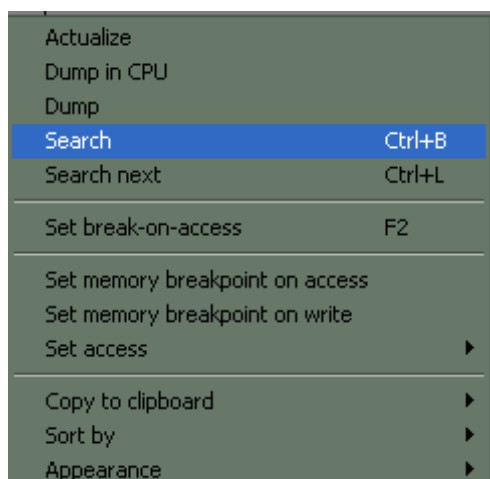
Comme pour la recherche de strings, nous allons nous servir de certains des mots composant ce nag.



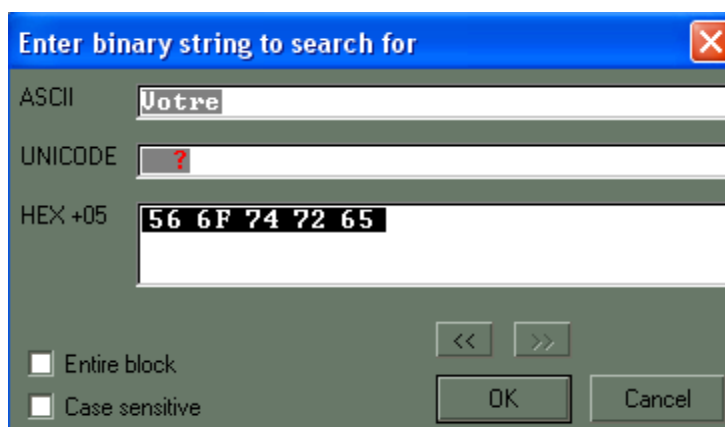
Cliquez sur le bouton M pour obtenir cette fenêtre :



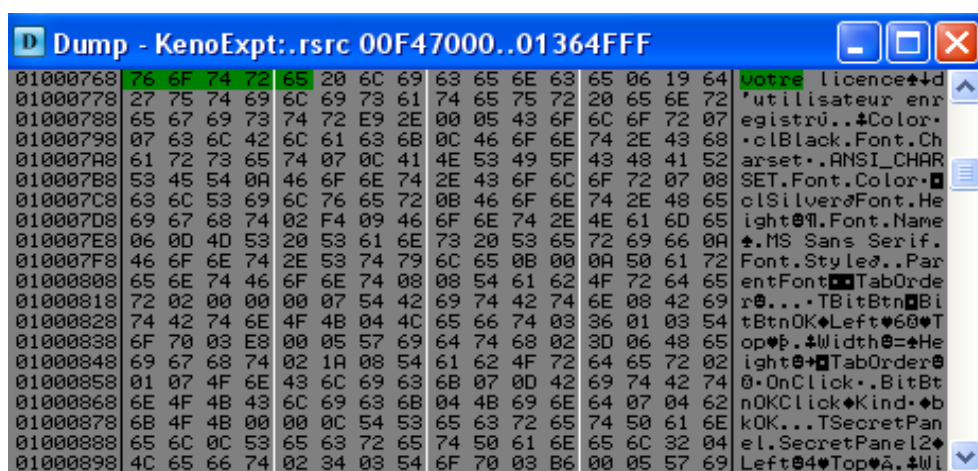
Clic droit dans la fenêtre Memory map apparue, pour obtenir cette boite de dialogue :



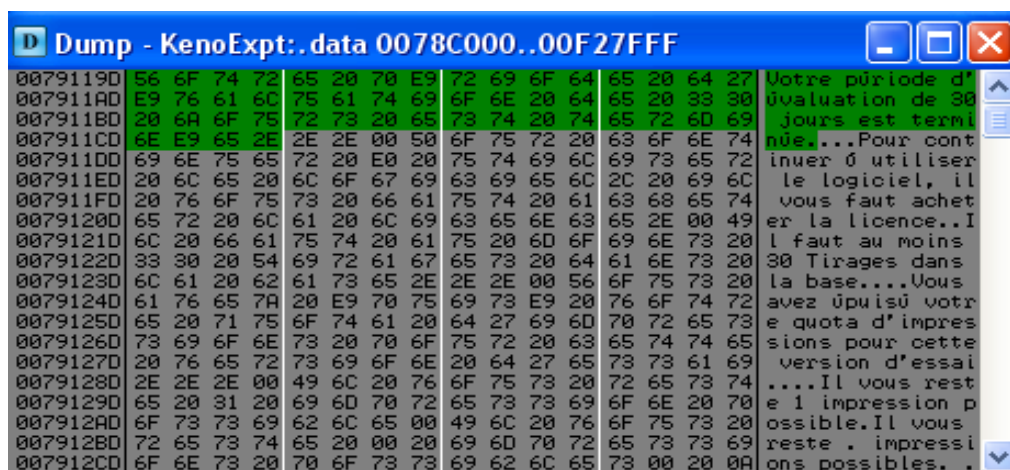
J'ai choisi le mot " Votre" du nag pour tenter une recherche à l'aide de la fenêtre suivante :



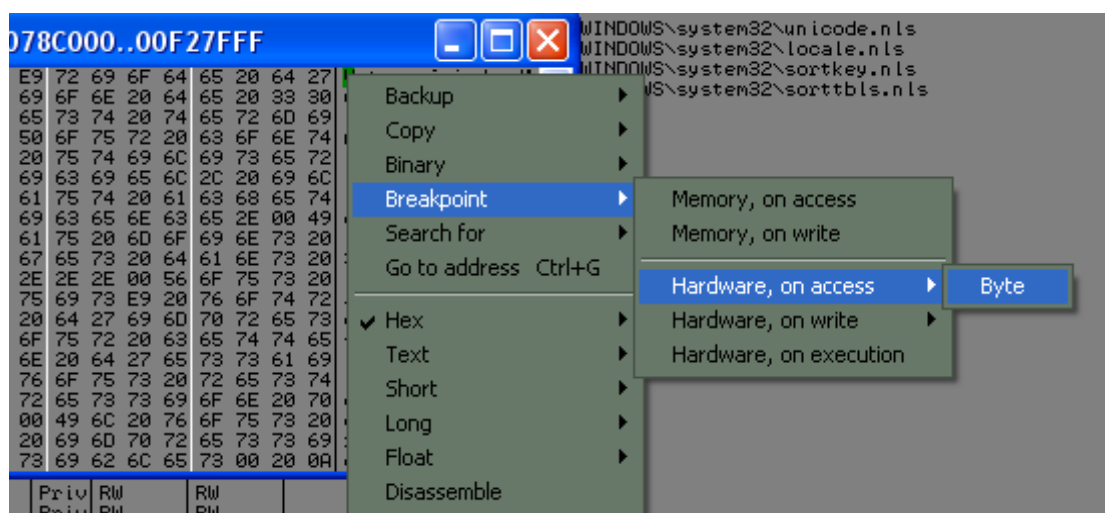
pour arriver ci-dessous ou nous trouvons l'occurence "votre" dans le dump. Mais la suite du message n'est pas celle que nous cherchons.



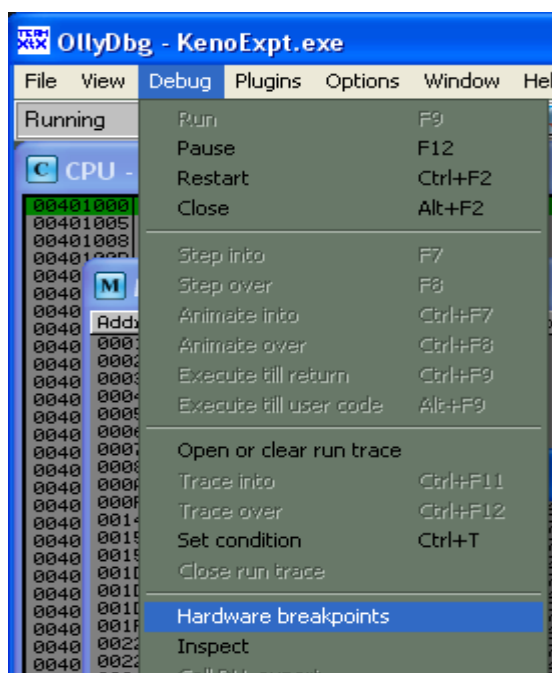
Alors CTRL+L pour passer à la suivante jusqu'à ce que nous trouvions notre message (pour moi 7 fois)



Sélectionnez uniquement le V du mot "Votre" et cliquez droit dessus, puis sélectionnez les options suivantes :

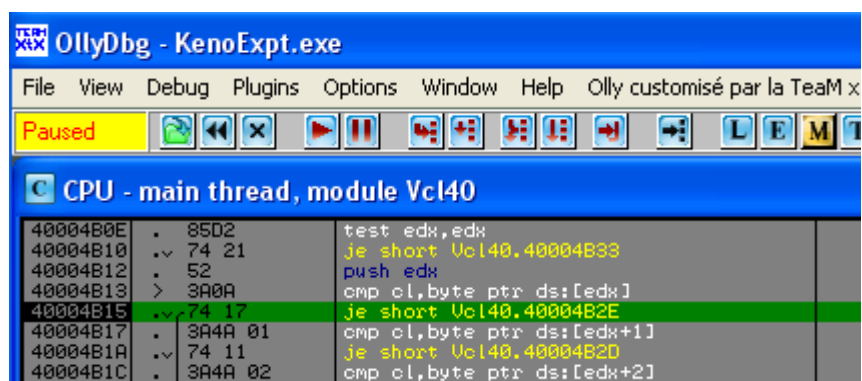


Désormais le BP est posé en mémoire et chaque fois que ce message sera sollicité, Olly breakera. Vous pouvez vérifier que votre BP est bien en place en consultant l'option qui suit :






A présent nous relançons la cible dans Olly qui breake ici :



En bas à gauche nous voyons bien ou Olly a breaké :

Address	Hex dump	ASCII
0078C000	01 00 0C 95 75 00 01 20 C8 95 75 00 01 10 F8 AA	0.2du.0 5u.0#9
0078C010	77 00 01 20 28 9E 75 00 01 20 08 9E 75 00 01 1E	w.0 (xu.0 iXu.0A
0078C020	10 70 76 00 01 20 80 84 76 00 01 20 90 88 76 00	!>u.0 3u.0 5iU.
0078C030	01 20 40 95 76 00 01 20 D0 80 76 00 01 00 B4 C5	0 0v.0 35u.0.1+
0078C040	76 00 01 00 10 C9 76 00 01 05 7C F2 76 00 01 04	u.0. 1Fv.0#1=U.0.
0078C050	05 0F 76 00 01 04 78 0A 77 00 01 0A 00 32 77 00	^U.0x.w.0.2w.
0078C060	01 0A 58 3A 77 00 01 0A 34 35 77 00 01 0A 4C 3F	0.X;w.0.45u.0.L?
0078C070	77 00 01 01 B4 46 77 00 01 01 10 4B 77 00 01 03	w.00Fw.00Kw.00
0078C080	B4 40 77 00 01 02 BC 50 77 00 01 03 A4 52 77 00	1W.00Pw.00W.
0078C090	01 06 50 57 77 00 01 01 08 59 77 00 01 01 F8 50	0APlw.00Vw.00Z

Command :  

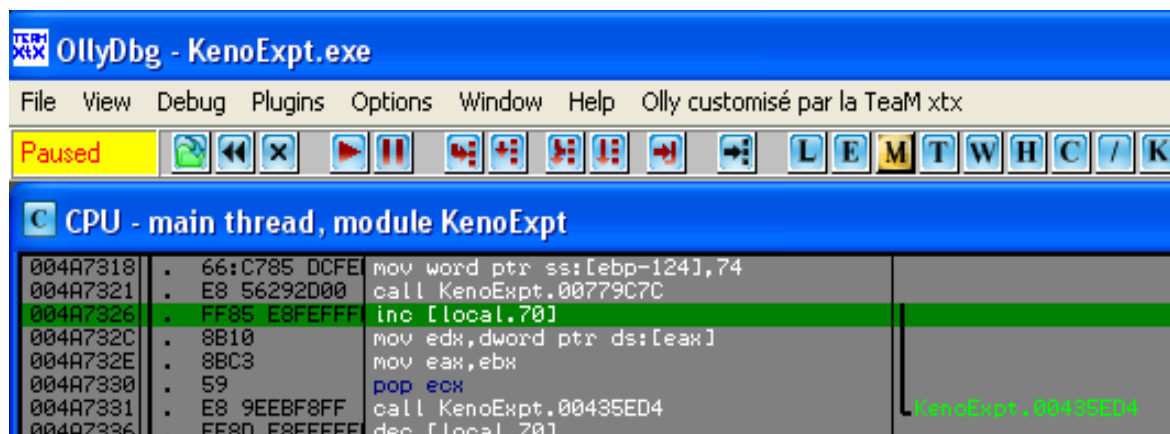
Hardware breakpoint 1 at Vcl40.40004B15 - EIP points to next instruction

Maintenant, parcourir la pile vers le bas jusqu'à trouver un `return to ... from` qui concerne le module `KenoExpt` :

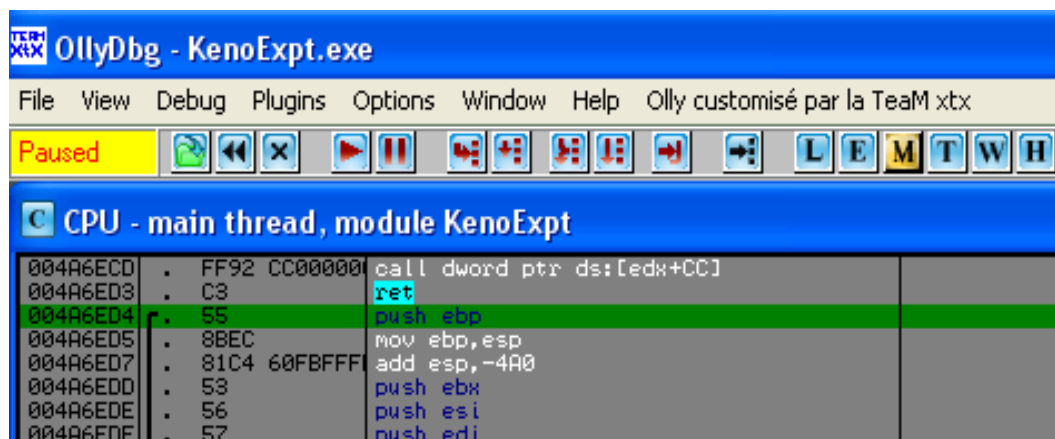
0236F520	00000000	
0236F524	00000000	
0236F528	0236F994	
0236F52C	0236F9E8	
0236F530	004A7326	RETURN to KenoExpt.004A7326 from KenoExpt.00779C7C
0236F534	06BFA7D8	ASCII "Message"
0236F538	06BFA800	
0236F53C	007D1C2C	KenoExpt.007D1C2C
0236F540	0236FC34	
0236F544	02902051	

Il semble que ce soit bien le return que nous cherchions puisque nous voyons le mot Message à la ligne suivante.

Clic droit sur la ligne **return to** puis **Follow in Disassembler** pour arriver à cette adresse :

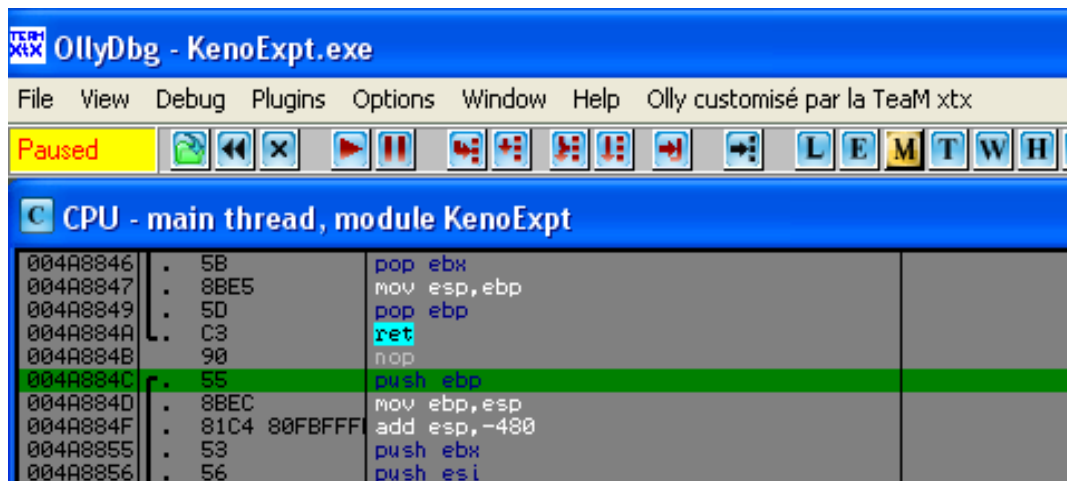


Il ne reste plus qu'à scroller vers le haut pour arriver à la première ligne de ce qui doit être la partie évaluation. (Souvenez vous, dans le 1er tuto, la partie enregistrée suivait toujours la partie évaluation)



Notons l'adresse de la ligne : **004A6ED4**

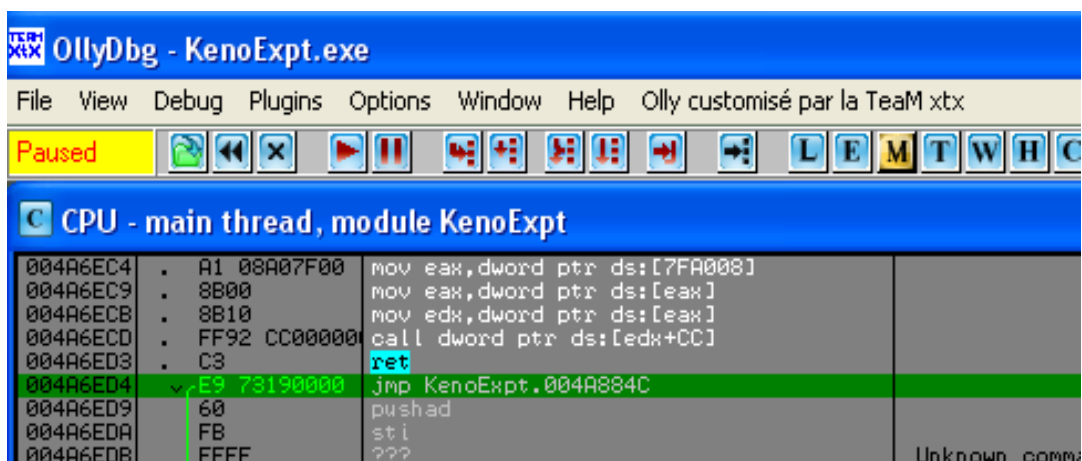
Et scrollons vers le bas pour trouver la première ligne de la partie enregistrée pro.



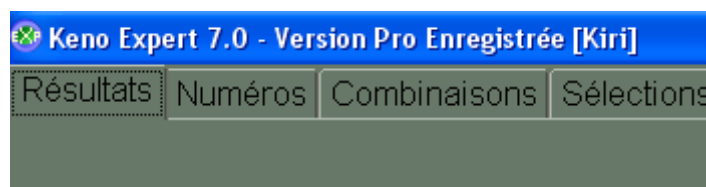
Notons l'adresse de la ligne : **004A884C**

Un petit clic droit au milieu de la zone de code – Go to – expression – 004A6ED4 et nous pouvons patcher pour obtenir notre version enregistrée pro!

Clic droit sur la ligne, Assemble – remplacer PUSH EBP par JMP 004A884C et voilà :  
Sauver les modifications comme vous savez le faire ...



Et relancez la cible pour admirer votre travail :



Et si on clique sur le bouton A propos :



Vous l'aurez compris, pour profiter pleinement du soft, il ne vous restera qu'à appliquer le même traitement à tous les boutons bridés...

Il y a déjà un certain temps que j'ai travaillé sur ce soft, et je n'y suis revenu que pour les besoins du cours. Alors il se peut qu'un bouton pose un cas particulier dont je ne me souviens pas.

Merci de me le signaler alors par l'intermédiaire du forum.

Remerciements plus spécialement à **Burner** (pour ses tutos dont je me suis inspiré et que je vous encourage à consulter sur ce forum), **Kirjo** (pour son astuce à propos du nag et sa relecture de ce cours), **Papa Bango** (pour son autre astuce), et à tous les autres membres de XTX et des autres forums qui me permettent de m'éclater au travers du reverse ...

**Kiri OCTOBRE 2013**