

Introduction au cracking

(explications de base)

(Les explications données dans ce tuto sont très simplifiées, afin que les plus débutants puissent comprendre facilement, ce tuto n'a pas pour but de donner un cours précis sur le cracking, mais de faire comprendre aux novices en quoi cela consiste .)



By Mr_roW

1/ Qu'est ce que le cracking

Le cracking, c'est le fait de contourner la sécurité des logiciels, exemple :

Vous avez un programme en version d'essai, il suffit de le « cracker » pour avoir une version illimitée.

Bon d'accord là je fais un peu simple, pour expliquer correctement, techniquement, cracker un programme consiste à le désassembler, et donc , d'obtenir son code source en langage Assembleur, afin de comprendre son fonctionnement et de modifier certaines données de celui ci pour , en général, contourner les sécurité, exemple, le programme demande d'entrer une clé d'activation, en modifiant certaines parties du programme, on peut modifier la condition qui teste le mot de passe entré pour que le programme accepte même un mot de passe faux.

2/ De quoi ai-je besoin pour cracker ?

Eh bien vous avez besoin de :

1/ Un désassembleur, comme : Ollydbg

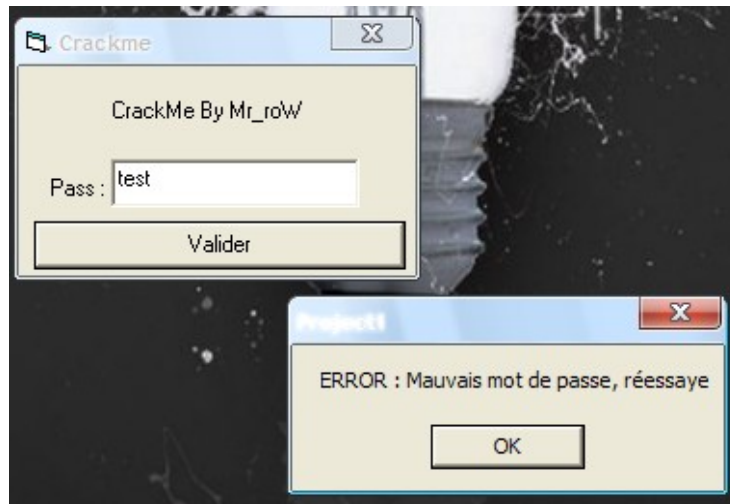
2/ Un éditeur hexadécimal, comme : EditHexa

3/ Des connaissances en Assembleur (car je ne donnerai pas de cours ici)

4/ Un cerveau performant

3/ Explications

Bon, pour vous expliquer cela correctement, j'ai créé un crack-me tout simple ;)



Comment faire (à part en devinant le mot de passe :p) pour réussir à contourner la sécurité ?

Eh bien c'est simple, pour cela on va désassembler notre programme à l'aide du désassembleur (dans le cas présent : Ollydbg).

The screenshot displays the Ollydbg disassembler interface. The main window shows assembly code for the 'crack-me' program. A red line highlights the instruction at address 0040119C: `PUSH crack-me.00401380`. The registers window on the right shows the current state of the CPU registers, with EIP at 0040119C. The bottom status bar shows the current instruction: `00401380=crack-me.00401380 (ASCII "UB5%&.*")`. A hex dump and ASCII dump are visible at the bottom of the disassembler window.

Olalalalah mais qu'est ce que c'est que tout ça ? Quel démon à pu inventer une horreur pareille ?

^^

Non , honnêtement ça paraît monstrueusement compliqué à première vue mais ça ne l'est pas tant que ça ;)

Le code source du programme est donc ce que vous voyez dans cette fenêtre en haut à gauche.

Et là un outil magique va nous venir en aide, on fait un clic droit dans la fenêtre en haut à droite, et on clique sur : search for > All referenced text strings

Ce sont les chaînes de caractères du programme qui sont disponibles dans leur format ASCII .

Donc une fois que l'on a cliqué là dessus, on a les chaînes de caractères ASCII du programme (celles qui sont disponibles évidemment ^^)

Nous avons vu plus haut que le programme affiche un message d'erreur en cas de faux mot de passe, on peut donc se douter qu'il mettra un message de validation en cas de password valide, en cherchant un peu dans les « referenced text strings », on arrive à trouver cela

```
00401C00 ASCII "vbaStrIn00e",0
00401C0C ASCII "vbaStrCat",0
00401CE8 ASCII "vbaFreeVarList"
00401CF8 ASCII 0
00401CFC ASCII "vbaVarDup",0
00401D08 ASCII "vbaFreeObj",0
00401D18 ASCII "vbaFreeStr",0
00401D28 ASCII "vbaResultChec"
00401D38 ASCII "kObj",0
00401D40 ASCII "vbaObjSet",0
00401D4C ASCII "vbaStrCmp",0
00401D5C ASCII "vbaStrCopy",0
00401E65 MOV EDX,crack-me.00401B98
00401F19 MOV DWORD PTR SS:[EBP-68],crack-me.00401B98 UNICODE "cracked"
00401F5C PUSH crack-me.00401C10 UNICODE "Bien jou"
00401F61 PUSH crack-me.00401C70 UNICODE "j"
00401F74 PUSH crack-me.00401C7C UNICODE "Soit tu as contourn"
004020EC ASCII "MSUBVM60.DLL",0
004020FC ASCII "Cicos",0
00402106 ASCII "adj_fptan",0
00402114 ASCII "vbaFreeVarList"
00402124 ASCII 0
00402128 ASCII "adj_fdiv_m64",0
00402138 ASCII "adj_fprem1",0
00402146 ASCII "vbaStrCat",0
00402154 ASCII "vbaStrCat",0
```

Tiens donc ^^ mais ne serai-ce pas notre message d'erreur sur la ligne que j'ai surligné en rouge ?

Et si on regarde bien en dessous, on voit un « Bien jou... » (chaîne coupée, car le «é» ne passe pas au désassembleur ;)

Vous vous doutez bien que cela est notre message de validation .

Bon double cliquons sur la ligne du message d'erreur, on se retrouve dans le code source (en ASM) à l'endroit ou le message d'erreur apparait

Instruction JE en hexadécimal

```
00401EFC . 66:3BF7    CMP SI,DI
00401EFF . 894D A8    MOV DWORD PTR SS:[EBP-58],ECX
00401F02 . 8945 A0    MOV DWORD PTR SS:[EBP-60],EAX
00401F05 . 894D B8    MOV DWORD PTR SS:[EBP-48],ECX
00401F08 . 8945 B0    MOV DWORD PTR SS:[EBP-50],EAX
00401F0B . 894D C8    MOV DWORD PTR SS:[EBP-38],ECX
00401F0E . 8945 C0    MOV DWORD PTR SS:[EBP-40],EAX
00401F11 . 74 43     JE SHORT crack-me.00401F56
00401F13 . 8D55 90    LEA EDX,DWORD PTR SS:[EBP-70]
00401F16 . 8D4D D0    LEA ECX,DWORD PTR SS:[EBP-30]
00401F19 . C745 98 EC1B4 MOV DWORD PTR SS:[EBP-68],crack-me.0040 UNICOD "ERROR : Mauvais mot de passe, r"
00401F20 . C745 90 00000 MOV DWORD PTR SS:[EBP-70],8
00401F27 . FF15 7C104000 CALL DWORD PTR DS:[&MSUBVM60.__vbaVarD MSUBVM60.__vbaVarDup
```

On arrive ici, la ligne surlignée en rouge correspond au message d'erreur, vous l'aurez compris ;)

Si on regarde un peu plus haut au dessus du message d'erreur (ligne indiquée par une flèche verte)

Il y a une instruction nommée **JE** , en assembleur, l'instruction JE est un saut conditionnel, (JUMP if EQUAL), Je n'ai pas besoin de vous faire un dessin, le programme fait un saut au message « Bien joué » si le mot de passe entré par l'utilisateur est égal à la variable qui le contient

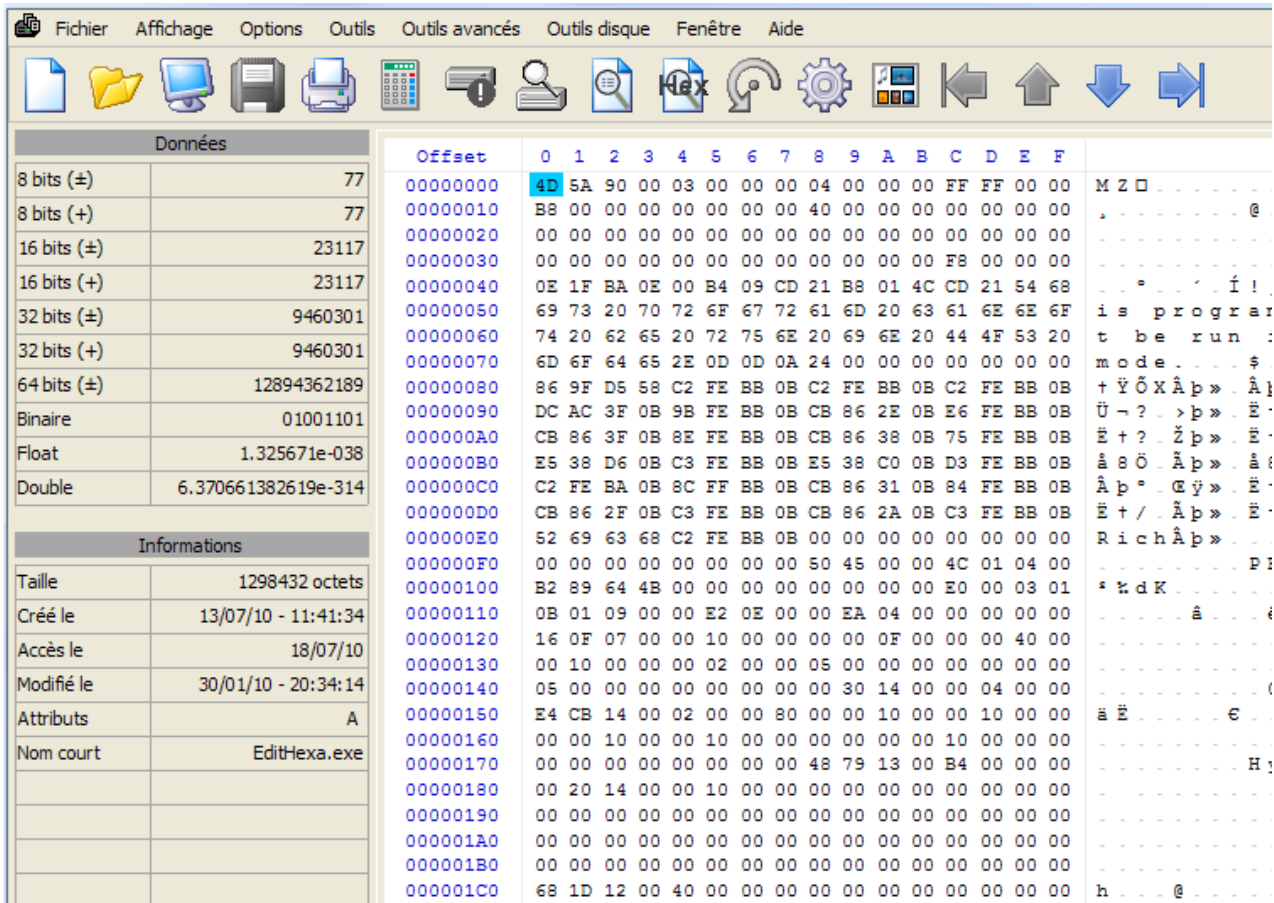
(d'ailleurs on peut le remarquer dans Ollydbg, car si on se met sur la ligne du JE et que l'on appuie sur Entrée, le désassembleur nous place à l'endroit du code où le message « Bien joué » apparait)

Pour vous renseigner sur les instruction principales en assembleur,

jetez un oeil ici : <http://www.commentcamarche.net/contents/asm/liste.php3>

Il nous suffit donc de modifier cette condition dans le programme, l'instruction JE en hexadécimal indiquée sur le screen est **74 43** , il nous suffit donc de modifier le 74 en 75 à l'aide de l'éditeur hexadécimal, pourquoi 75 ? car 75 est un «JNE» c'est à dire une instruction qui fait le contraire de JE, il fait un saut au message de réussite «Bien Joué», tant que le mot de passe est faux dans le champ texte ^^ , il n'interdira donc pas l'accès ,

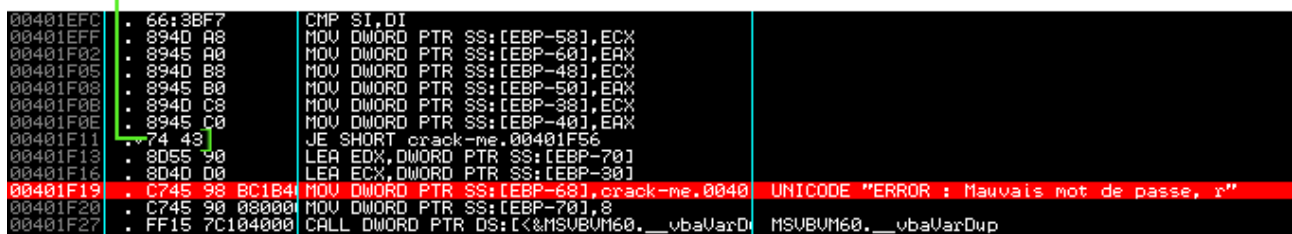
On ouvre donc notre crack-me avec l'éditeur hexadécimal :



Eh oui, tout ce que vous voyez là, c'est notre programme sous forme de chaînes hexadécimales, (avec les caractères ASCII disponibles à droite ^^)

Donc, pour retrouver notre JE, il faut retourner à OllyDbg , et regarder la chaîne exacte de l'instruction (avec la chaîne précédente et la chaîne suivante)

Instruction JE en hexadécimal



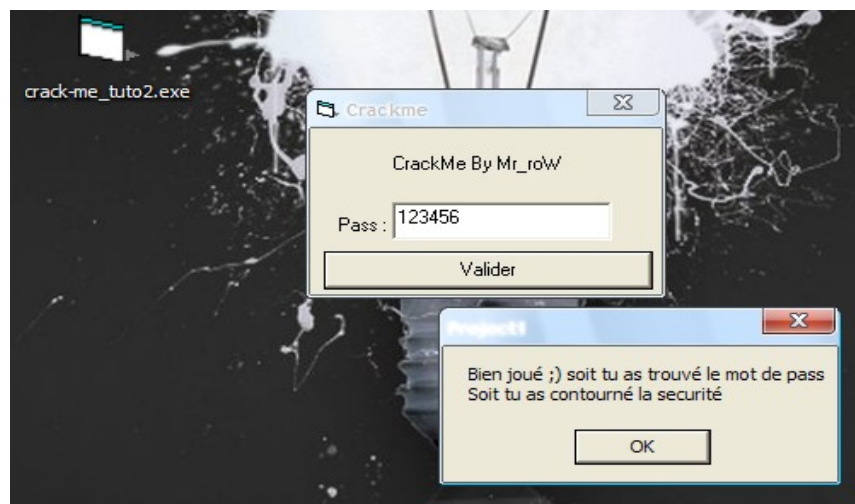
On voit donc que si on prends la chaîne du JE avec la chaîne lui succédant, la chaîne hexadécimale est : **74438d5590**

Pourquoi prendre la chaîne suivant notre instruction ? Hahaaa ? Car on peut facilement retrouver plusieurs fois la même occurrence dans le programme, donc afin de ne pas se retrouver avec une vingtaine de 74 43 à trier pour trouver le bon ^^ on prends une chaîne plus grande, comme ça, moins de chances de la retrouver plusieurs fois :) (Généralement on prends aussi un morceau de la chaîne précédant l'instruction, je ne l'ai pas fait dans mon exemple, mais je le précise ;)

Données	Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8 bits (±)	116	0C	1B	40	00	70	1B	40	00	E0	32	40	00	CC	CC	CC	CC
8 bits (+)	116	E9	E9	E9	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC	CC
16 bits (±)	17268	55	8B	EC	83	EC	0C	68	B6	10	40	00	64	A1	00	00	00
16 bits (+)	17268	00	50	64	89	25	00	00	00	00	81	EC	A0	00	00	00	53
32 bits (±)	1435321204	C6	83	E0	01	89	45	FC	83	E6	FE	56	89	75	08	8B	0E
32 bits (+)	1435321204	FF	51	04	33	FF	BA	98	1B	40	00	8D	4D	E8	89	7D	E8
64 bits (±)	-3436935289563692172	89	7D	E4	89	7D	E0	89	7D	D0	89	7D	C0	89	7D	B0	89
Binaire	01110100	7D	A0	89	7D	90	FF	15	6C	10	40	00	8B	16	56	FF	92
Float	1.941511e+013	04	03	00	00	50	8D	45	E0	50	FF	15	20	10	40	00	8B
Double	-6.844003431933e+078	F0	8D	55	E4	52	56	8B	0E	FF	91	A0	00	00	00	3B	C7
Informations		DB	E2	7D	12	68	A0	00	00	00	68	A8	1B	40	00	56	50
Taille	20480 octets	FF	15	18	10	40	00	8B	45	E4	8B	4D	E8	50	51	FF	15
Créé le	18/07/10 - 19:27:11	3C	10	40	00	8B	1D	94	10	40	00	8B	F0	F7	DE	1B	F6
Accès le	18/07/10	8D	4D	E4	F7	DE	F7	DE	FF	D3	8D	4D	E0	FF	15	98	10
Modifié le	18/07/10 - 15:35:21	40	00	B9	04	00	02	80	B8	0A	00	00	66	3B	F7	89	
Attributs	A	4D	A8	89	45	A0	89	4D	B8	89	45	B0	89	4D	C8	89	45
Nom court	CRACK~1.EXE	00	74	43	8D	55	90	8D	4D	D0	C7	45	98	BC	1B	40	00
		C7	45	90	08	00	00	00	FF	15	7C	10	40	00	8D	55	A0
		8D	45	B0	52	8D	4D	C0	50	51	8D	55	D0	57	52	FF	15
		24	10	40	00	8D	45	A0	8D	4D	B0	50	8D	55	C0	51	8D
		45	D0	52	50	EB	5B	8B	35	14	10	40	00	68	10	1C	40
		00	68	70	1C	40	00	FF	D6	8B	D0	8D	4D	E4	FF	15	84
		10	40	00	50	68	7C	1C	40	00	FF	D6	8D	4D	A0	89	45
		D8	8D	55	B0	51	8D	45	C0	52	50	8D	4D	D0	57	51	C7
		45	D0	08	00	00	00	FF	15	24	10	40	00	8D	4D	E4	FF
		D3	8D	55	A0	8D	45	B0	52	8D	4D	C0	50	8D	55	D0	51
		52	6A	04	FF	15	08	10	40	00	83	C4	14	89	7D	FC	68
		FE	1F	40	00	EB	2E	8D	4D	E4	FF	15	94	10	40	00	8D
		4D	E0	FF	15	98	10	40	00	8D	45	A0	8D	4D	B0	50	8D
		55	C0	51	8D	45	D0	52	50	6A	04	FF	15	08	10	40	00
		83	C4	14	C3	8D	4D	E8	FF	15	94	10	40	00	C3	8B	45

Nous n'avons plus qu'à modifier notre 74 43 en 75 43 comme je l'ai mentionné plus haut, on sauvegarde ensuite notre fichier (de préférence sans écraser notre fichier d'origine, donc sauvegardez sous un autre nom)

Ensuite, on lance le crack-me modifié, on entre un mot de passe bidon, exemple : 123456 et on valide ;)



Voilà, c'était une brève introduction au cracking pour ceux qui se demandaient en quoi ça consistait. Évidemment, pour de vrais programme cela n'est pas aussi rapide, il faut un certain temps avant de trouver LA ou LES sécurités a contourner. Si après cette introduction le cracking vous intéresse, je vous conseille ce site afin de suivre des cours bien construits, bien expliqués et qui vous aideront à avancer à grands pas dans le monde de la sécurité logicielle.

Lien du site : <http://www.deezdynasty.xdir.org/tutorials.html>

Cordialement, Mr_Row ...