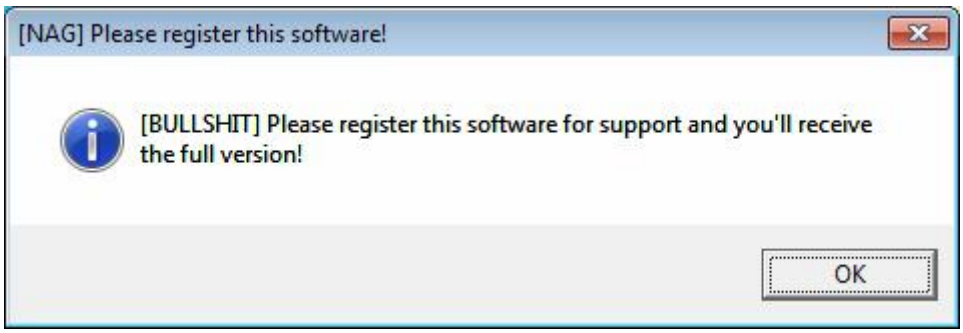
سری آموزش های R4NDOM-فارسی

قسمت ۱۴-How to remove nag screens

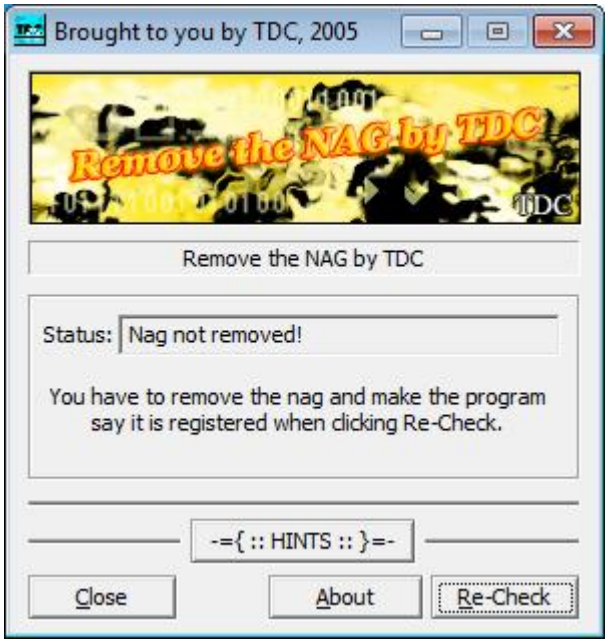
پنجره های مزاحم یا (Nag) ،پنجره هایی هستند که به صورت آزار دهنده ایی هر بار که برنامه را اجرا میکنید و یا حتی در هنگام اجرای برنامه ،مدام نمایش داده میشوند و به این معنی هستند که زمان تست برنامه ی مورد نظر در حال اتمام هست ویا شما از نسخه ی تریال استفاده میکنید (با قابلیت های کمتر از نسخه ی اصلی) و نیاز است که برنامه برای اجرای کامل از طریق وب سایت (آنلاین) و یا دریافت لایسنس معتبر(به صورت آف لاین) رجیستر شود.همچنین در این آموزش از یک پلاگین مربوط به دیباگر Olly استفاده شده که قابلیت های گوناگون و جالبی دارد در این آموزش تا حدودی خصوصیت ها و کاربرد آن تشریح شده اما فیلم آموزش کامل آن ضمیمه شده که تووصیه میکنم حتما آن را ببینید ،دراین قسمت از آموزش ما از دو برنامه استفاده میکنیم که دانش ما را به چالش خواهد کشید

اولین برنامه

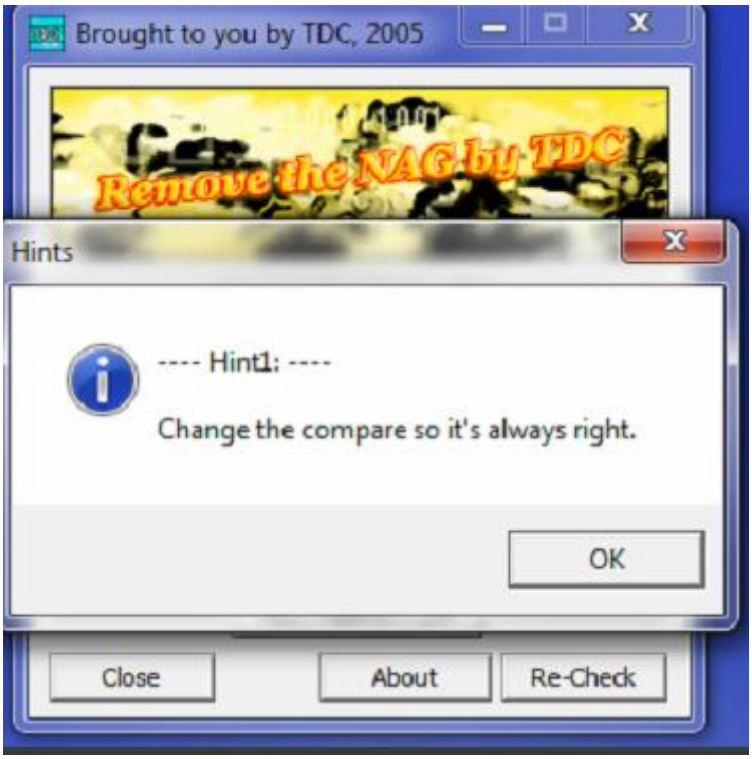
نخست برنامه ی ضمیمه شده در این آموزش را با نام Nag1.exe را اجرا کنید ، در ابتدا پنجره ی مزاحم (Nag) نمایش داده میشود:



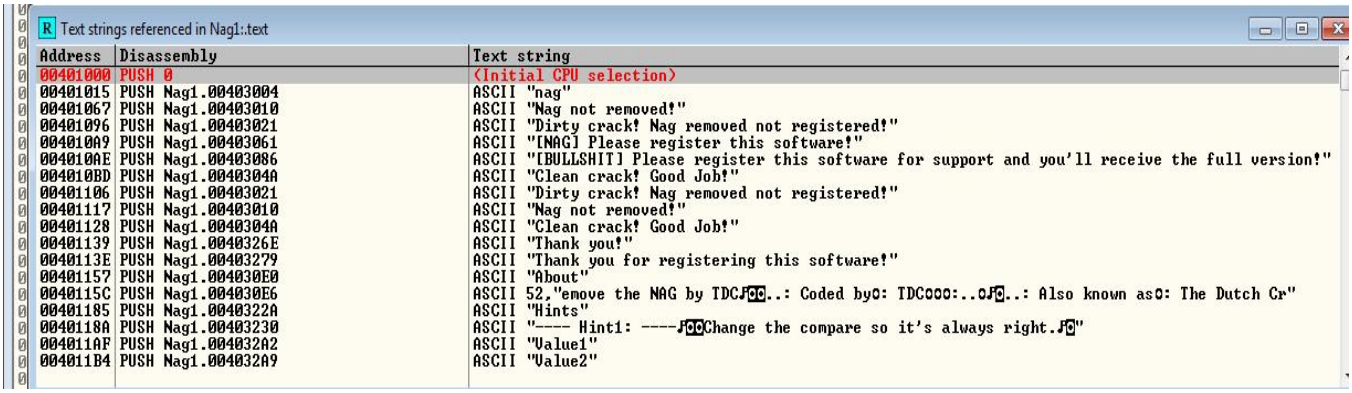
به وضوح میتوان گفت که توسط کرکر ساخته شده است ☺ ، به هر حال بعد از کلیک بروی دکمه OK وارد پنجره ی اصلی میشوید:



همانطور که میبیند پیام "Nag not removed " نمایش داده شده و همینطور اگر بروی دکمه ی "Hints" کلیک کنید یک راهنمایی دریافت خواهیم کرد :



بسیار خوب،برنامه را درون دیباگر باز کنید و مثله همیشه از تکنیک قدیمی ولی قابل اعتماد search for strings استفاده میکنیم :



Text strings referenced in Nag1:.text

Address	Disassembly	Text string
00401000	PUSH 0	(Initial CPU selection)
00401015	PUSH Nag1.00403004	ASCII "nag"
00401067	PUSH Nag1.00403010	ASCII "Nag not removed!"
00401096	PUSH Nag1.00403021	ASCII "Dirty crack! Nag removed not registered!"
004010A9	PUSH Nag1.00403061	ASCII "[NAG] Please register this software!"
004010AE	PUSH Nag1.00403086	ASCII "[BULLSHIT] Please register this software for support and you'll receive the full version!"
004010BD	PUSH Nag1.0040304A	ASCII "Clean crack! Good Job!"
00401106	PUSH Nag1.00403021	ASCII "Dirty crack! Nag removed not registered!"
00401117	PUSH Nag1.00403010	ASCII "Nag not removed!"
00401128	PUSH Nag1.0040304A	ASCII "Clean crack! Good Job!"
00401139	PUSH Nag1.0040326E	ASCII "Thank you!"
0040113E	PUSH Nag1.00403279	ASCII "Thank you for registering this software!"
00401157	PUSH Nag1.004030E0	ASCII "About"
0040115C	PUSH Nag1.004030E6	ASCII 52,"emove the NAG by TDC♪♫..: Coded by♫: TDC♫♫♫:..♫♪..: Also known as♫: The Dutch Cr"
00401185	PUSH Nag1.0040322A	ASCII "Hints"
0040118A	PUSH Nag1.00403230	ASCII "---- Hint1: ----♪♫♫Change the compare so it's always right.♪♫"
004011AF	PUSH Nag1.004032A2	ASCII "Value1"
004011B4	PUSH Nag1.004032A9	ASCII "Value2"

مثله اینکه امروز روی شانسیم ☺ شما میتوانید متن "Nag Screen" را در آدرس 4010AE ببینید بروی آن دبل کلیک کنید تا ببینیم کجا ساخته میشود:

```
- [*G.P.U* - main thread, module Nag1]
File View Debug Plugins Options Window Help Tools BreakPoint->

00401058  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040105A  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
0040105D  .  E8 96010000    CALL <JMP.&user32.GetDlgItem>            GetDlgItem
00401062  .  A3 0C304000    MOV DWORD PTR [40300C],EAX
00401067  .  68 10304000    PUSH Nag1.00403010                       Text = "Nag not removed!"
0040106C  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040106E  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
00401071  .  E8 9A010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
00401076  .  E8 34010000    CALL Nag1.004011AF
0040107B  .  803D B0324000  CMP BYTE PTR [4032B0],3
00401082  .v 74 12          JE SHORT Nag1.00401096
00401084  .  803D B0324000  CMP BYTE PTR [4032B0],2
0040108B  .v 74 1A          JE SHORT Nag1.004010A7
0040108D  .  803D B0324000  CMP BYTE PTR [4032B0],1
00401094  .v 74 27          JE SHORT Nag1.004010BD
00401096  >  68 21304000    PUSH Nag1.00403021                       Text = "Dirty crack! Nag removed not registered!"
0040109B  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040109D  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
004010A0  .  E8 6B010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
004010A5  .v EB 27          JMP SHORT Nag1.004010CE
004010A7  >  6A 40          PUSH 40                                  Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
004010A9  .  68 61304000    PUSH Nag1.00403061                       Title = "[NAG] Please register this software!"
004010AE  .  68 86304000    PUSH Nag1.00403086                       Text = "[BULLSHIT] Please register this software
004010B3  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hOwner
004010B6  .  E8 49010000    CALL <JMP.&user32.MessageBoxA>           MessageBoxA
004010BB  .v EB 11          JMP SHORT Nag1.004010CE
004010BD  >  68 4A304000    PUSH Nag1.0040304A                       Text = "Clean crack! Good Job!"
004010C2  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
004010C4  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
004010C7  .  E8 44010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
004010CC  .v EB 00          JMP SHORT Nag1.004010CE
004010CE  >v E9 D6000000    JMP Nag1.004011A9
004010D3  >  817D 0C 1101(  CMP DWORD PTR [EBP+C],111
004010DA  .v 0F85 B900000(  JNZ Nag1.00401199
004010E0  .  837D 10 6F     CMP DWORD PTR [EBP+10],6F
004010E4  .v 75 69          JNZ SHORT Nag1.0040114F
004010E6  .  E8 C4000000    CALL Nag1.004011AF
004010EB  .  803D B0324000  CMP BYTE PTR [4032B0],3
004010F2  .v 74 12          JE SHORT Nag1.00401106
004010F4  .  803D B0324000  CMP BYTE PTR [4032B0],2
004010FB  .v 74 1A          JE SHORT Nag1.00401117
004010FD  .  803D B0324000  CMP BYTE PTR [4032B0],1
00401104  .v 74 22          JE SHORT Nag1.00401128
00401106  >  68 21304000    PUSH Nag1.00403021                       Text = "Dirty crack! Nag removed not registered!"
0040110B  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040110D  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
```

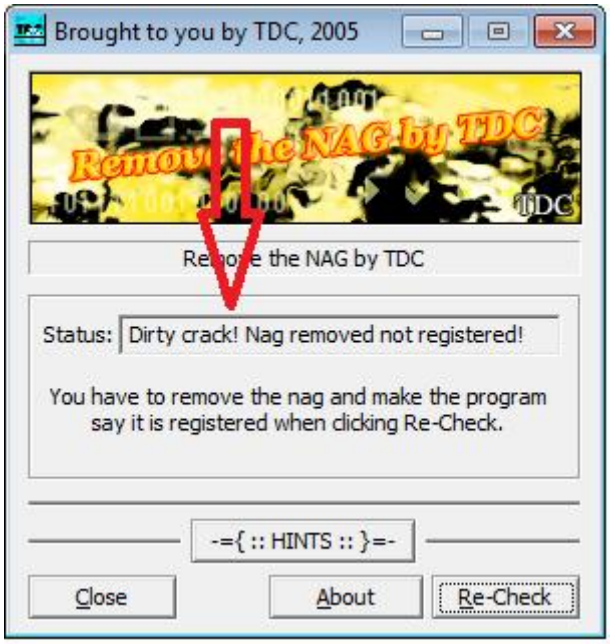
رشته های پایین را نگاه کنید ،اما فعلا با اینها کاری نداریم ،الان به ابتدای تابع MessageBoxA در خط 4010A7بروید تا ببینم این فراخوانی از کجا صادر شده:

```
00401082  .v 74 12          JE SHORT Nag1.00401096
00401084  .  803D B0324000 02 CMP BYTE PTR DS:[4032B0],2
0040108B  .v 74 1A          JE SHORT Nag1.004010A7
0040108D  .  803D B0324000 01 CMP BYTE PTR DS:[4032B0],1
00401094  .v 74 27          JE SHORT Nag1.004010BD
00401096  >  68 21304000    PUSH Nag1.00403021                       Text = "Dirty crack! Nag removed not registered!"
0040109B  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040109D  .  FF75 08        PUSH [ARG.1]                             hWnd = 7EFDE000
004010A0  .  E8 6B010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
004010A5  .v EB 27          JMP SHORT Nag1.004010CE
004010A7  >  6A 40          PUSH 40                                  Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
004010A9  .  68 61304000    PUSH Nag1.00403061                       Title = "[NAG] Please register this software!"
004010AE  .  68 86304000    PUSH Nag1.00403086                       Text = "[BULLSHIT] Please register this software for support and y
004010B3  .  FF75 08        PUSH [ARG.1]                             hOwner = 7EFDE000
```

همانطور که در تصویر بالا میبینید این فراخوانی از یک دستور JE در خط 4010B8 درست بعد از یک مقایسه صادر شده ☺ ،بروی همین خط یک BP بگذارید و برنامه را اجرا کنید ،برنامه بروی خط 4010B8 متوقف خواهد شد و اگر وضعیت این مقایسه را نگاه کنید میبینید که پرش انجام خواهد شد،وظیفه ی ما این است که مطمئن شویم این پرش صورت نگیرد ، برای این کار z Flag را برابر 0 تنظیم میکنیم :

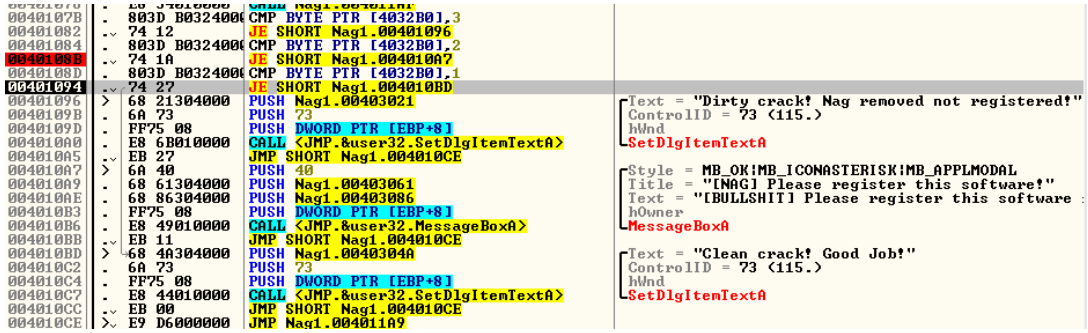
```
C 0
P 1
A 0
Z 0
S 0
T 0
D 0
O 0
```

در ادامه برنامه را با کلید F9 اجرا کنید :

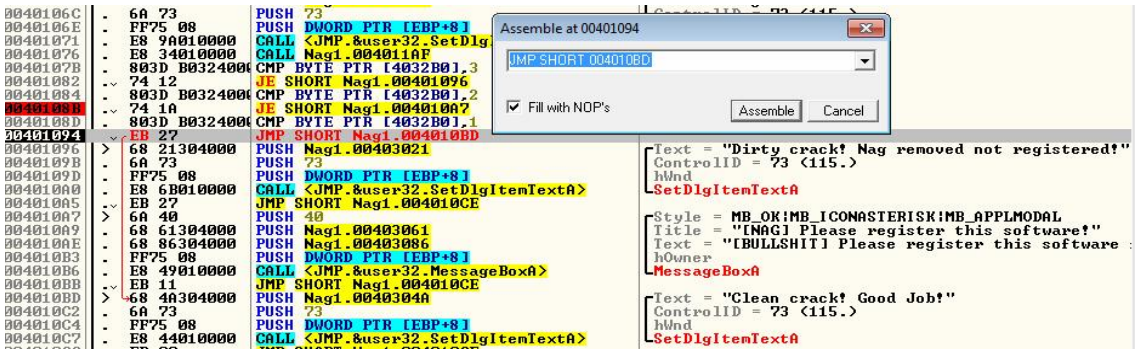


همانطور که میبینید پیام "Dirty Crack" ظاهر شد این یعنی پچ ما کافی نبوده ،بیایید برنامه را دوباره تحت دیباگر اجرا کنیم ، Olly در همان آدرس (4010B8) متوقف میشود و البته مقدار Z flag را دوباره برابر با 0 قرار میدهیم :

حال کلید F8 را دو بار بفشارید، حدس ما بر این است که این پرش اگر انجام شود ما را به سوی Good Boy ارجاع میدهد، و در غیر این صورت به سوی Bad Boy خواهیم رفت :



نظرتان درباره ی اینکه این پرش همواره انجام شود چیست؟ بیایید پچ خود را با تغییر دستورالعمل JE به JMP اعمال کنیم :



حال برنامه را اجرا کنید و خواهیم دید:

خب همانطور که میبیند کار انجام شده ، شما میتوانید به آدرس 4010B8 بروید (همان جایی که مقدار پرچم z=1 را به z=0 قرار دادیم) و این دستور العمل را طوری تغییر دهید که همواره پرشی صورت نگیرد مثلا با دستور NOP جایگزین کنید و نتیجه هم ذخیره کنید ،و این پچ به خوبی کار میکند ،اما من همیشه گفته ام که راه های زیادی برای پچ کردن وجود دارد برای این کار ابتدا بروی آدرس 401082یک BP بگذارید و سپس برنامه را Restart کنید :

```
00401082  .v 74 12        JE SHORT Nag1.00401096
00401084  .  803D B032400 CMP BYTE PTR [4032B0],2
0040108B  .v 74 1A        JE SHORT Nag1.004010A7
0040108D  .  803D B032400 CMP BYTE PTR [4032B0],1
00401094  v  EB 27        JMP SHORT Nag1.004010BD
00401096  >  68 21304000  PUSH Nag1.00403021          Text = "Dirty crack! Nag removed not registered!"
0040109B  .  6A 73        PUSH 73                     ControlID = 73 (115.)
0040109D  .  FF75 08      PUSH DWORD PTR [EBP+8]      hWnd
004010A0  .  E8 6B010000  CALL <JMP.&user32.SetDlgItemTextA>  SetDlgItemTextA
004010A5  .v EB 27        JMP SHORT Nag1.004010CE
004010A7  >  6A 40        PUSH 40                     Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
004010A9  .  68 61304000  PUSH Nag1.00403061          Title = "[NAG] Please register this software!"
004010AE  .  68 86304000  PUSH Nag1.00403086          Text = "[BULLSHIT] Please register this software
004010B3  .  FF75 08      PUSH DWORD PTR [EBP+8]      hOwner
004010B6  .  E8 49010000  CALL <JMP.&user32.MessageBoxA>  MessageBoxA
004010BB  .v EB 11        JMP SHORT Nag1.004010CE
004010BD  >  68 4A304000  PUSH Nag1.0040304A          Text = "Clean crack! Good Job!"
004010C2  .  6A 73        PUSH 73                     ControlID = 73 (115.)
004010C4  .  FF75 08      PUSH DWORD PTR [EBP+8]      hWnd
004010C7  .  E8 44010000  CALL <JMP.&user32.SetDlgItemTextA>  SetDlgItemTextA
004010CC  .v EB 00        JMP SHORT Nag1.004010CE
```

اگر بخواهیم این دستورالعمل های بالا را با زبان سطح بالا نشان دهیم اینگونه خواهد بود :

if (contents of 4032B0 == 3)
jump “Dirty Crack”
else if(contents of 4032B0 == 2)
jump to “Show Nag Screen
else if (contents of 4032B0 == 1)
jump to Good Boy Msg
else
Display “Dirty Crack”

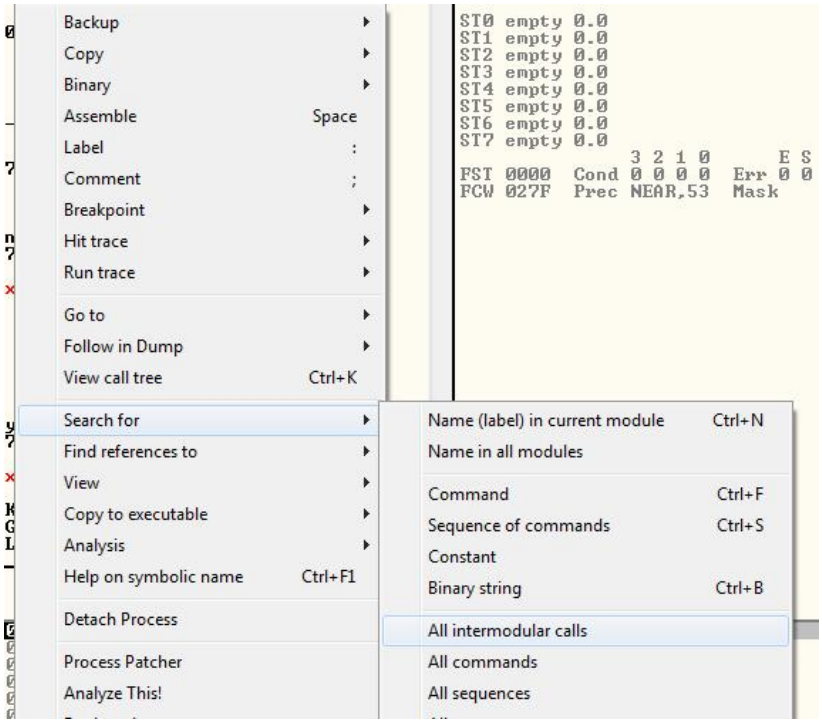
همانطور که میدانیم این برنامه همیشه به صورت پیش فرض Nag Screen را نمایش میدهد این یعنی مقدار آدرس حافظه ی 4032B0 همواره برابر با 2 است که به معنی انجام شدن پرش است ،حال اگر دستور العمل JE SHORT 00401096 واقع در آدرس 401082 را با JMP SHORT 004010BD تعویض کنیم چه اتفاقی رخ خواهد داد؟بله ما مستقیم به سوی Good Boy میرویم تنها با یک بار پچ میتوانیم این کارا انجام دهیم :

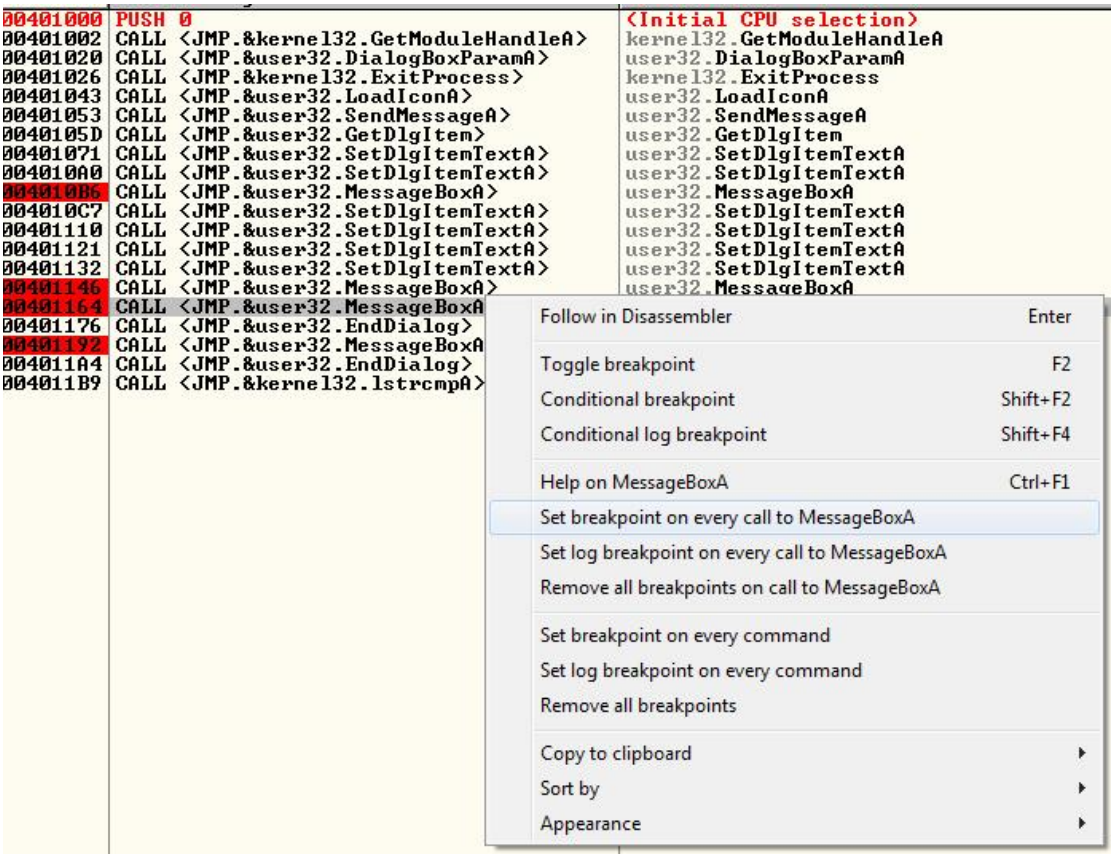
```
0401062  .  A3 0C304000   MOV DWORD PTR [40300C],EAX
0401067  .  68 10304000   PUSH Nag1.00403010
040106C  .  6A 73         PUSH 73
040106E  .  FF75 08       PUSH DWORD PTR [EBP+8]
0401071  .  E8 9A010000   CALL <JMP.&user32.SetDlgItemText
0401076  .  E8 34010000   CALL Nag1.004011AF
040107B     803D B0324000 CMP BYTE PTR [4032B0],3
0401082     EB 39         JMP SHORT Nag1.004010BD
0401084     803D B0324000 CMP BYTE PTR [4032B0],2
040108B     74 1A         JE SHORT Nag1.004010A7
040108D     803D B0324000 CMP BYTE PTR [4032B0],1
0401094     74 27         JE SHORT Nag1.004010BD
0401096  >  68 21304000   PUSH Nag1.00403021
040109B  .  6A 73         PUSH 73
040109D  .  FF75 08       PUSH DWORD PTR [EBP+8]
04010A0  .  E8 6B010000   CALL <JMP.&user32.SetDlgItemTextA>
04010A5  .  EB 27         JMP SHORT Nag1.004010CE
04010A7  >  6A 40         PUSH 40
04010A9  .  68 61304000   PUSH Nag1.00403061
04010AE  .  68 86304000   PUSH Nag1.00403086
04010B3  .  FF75 08       PUSH DWORD PTR [EBP+8]
04010B6  .  E8 49010000   CALL <JMP.&user32.MessageBoxA>
04010BB  .  EB 11         JMP SHORT Nag1.004010CE
04010BD  >  68 4A304000   PUSH Nag1.0040304A
04010C2  .  6A 73         PUSH 73
04010C4  .  FF75 08       PUSH DWORD PTR [EBP+8]
04010C7  .  E8 44010000   CALL <JMP.&user32.SetDlgItemTextA>
04010CC  .  EB 00         JMP SHORT Nag1.004010CE
04010CE  >  E9 D6000000   JMP Nag1.004011A9
04010D3  >  817D 0C 1101  CMP DWORD PTR [EBP+C],111

JMP SHORT 004010BD
Fill with NOP's   Assemble   Cancel

Text = "Dirty crack! Nag removed not registered!"
ControlID = 73 (115.)
hWnd
SetDlgItemTextA

Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
Title = "[NAG] Please register this software!"
Text = "[BULLSHIT] Please register this software
hOwner
MessageBoxA

Text = "Clean crack! Good Job!"
ControlID = 73 (115.)
hWnd
SetDlgItemTextA
```

و سپس برنامه را اجرا میکنیم :

همانطور که میبینیم این پچ هم درست کار میکند ،اما یک راه دیگر این است که با توجه به این مطلب که برای نمایش Bad boy مقدار حافظه ی 4032B0 باید برابر 2باشد و برای نمایش Good Boyاین مقدار باید برابر 1 باشد ،یعنی به محل آدرس 4032B0 رفته و با استفاده از Binary Edit تغییرات خود را لحاظ کنید من این کار را به عهده ی شما میگذارم.تست کنید ببینید کار میکند یا نه !

مطلب دیگر که میخواهم عنوان کنم این است که اگر به هر دلیلی نمی خواهیم یا نمیتوانیم از تکنیـک جـستجوی رشته ها استفاده کنیم میتوان از قابلیت search for Intermodular calls بهره برد :





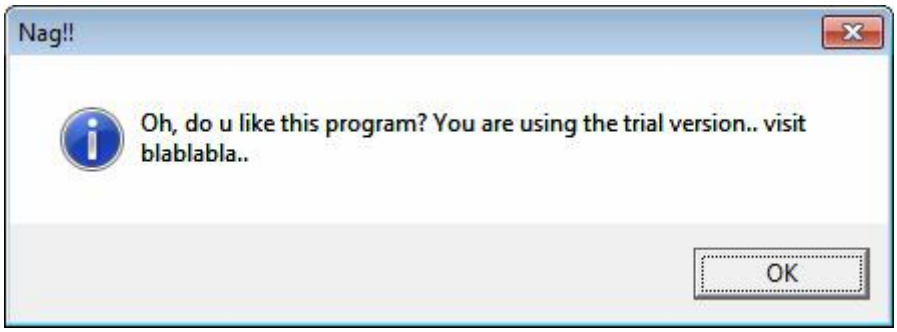
توجه داشته باشید که در اینجا ۴ تابع MessageBoxA وجود دارد ،شما میتوانید بروی یکی راست کلیک کنید و عبارت "Set breakpoint on every call to MessageBoxA" را انتخاب کنید تا بروی تمامی فراخوانی این تابع BP گذارده شود.و وقتی برنامه را اجرا میکنید قبل از هرچیز بروی خط 4010B6 متوقف میشود :

```
0040109B  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
0040109D  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
004010A0  .  E8 6B010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
004010A5  .v EB 27          JMP SHORT Nag1.004010CE
004010A7  >  6A 40          PUSH 40                                  Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
004010A9  .  68 61304000    PUSH Nag1.00403061                       Title = "[NAG] Please register this software!"
004010AE  .  68 86304000    PUSH Nag1.00403086                       Text = "[BULLSHIT] Please register this software
004010B3  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hOwner
004010B6  .  E8 49010000    CALL <JMP.&user32.MessageBoxA>           MessageBoxA
004010BB  .v EB 11          JMP SHORT Nag1.004010CE
004010BD  >  68 4A304000    PUSH Nag1.0040304A                       Text = "Clean crack! Good Job!"
004010C2  .  6A 73          PUSH 73                                  ControlID = 73 (115.)
004010C4  .  FF75 08        PUSH DWORD PTR [EBP+8]                   hWnd
004010C7  .  E8 44010000    CALL <JMP.&user32.SetDlgItemTextA>       SetDlgItemTextA
004010CC  .v EB 00          JMP SHORT Nag1.004010CE
004010CE  >v E9 D6000000    JMP Nag1.004011A9
004010D3  >  817D 0C 11010  CMP DWORD PTR [EBP+C],111
004010DA  .v 0F85 B9000000  JNZ Nag1.00401199
```

محلش آشناست نه؟ این Messagebox مربوط به نمایش Nag است ،پس همیشه یادتان باشد که بیشتر از یک راه برای انجام چیزی وجود دارد، به زودی چند تکنیک یاد خواهیم گرفت که بسیار مفید هستند مثلـه (windows message handlers)

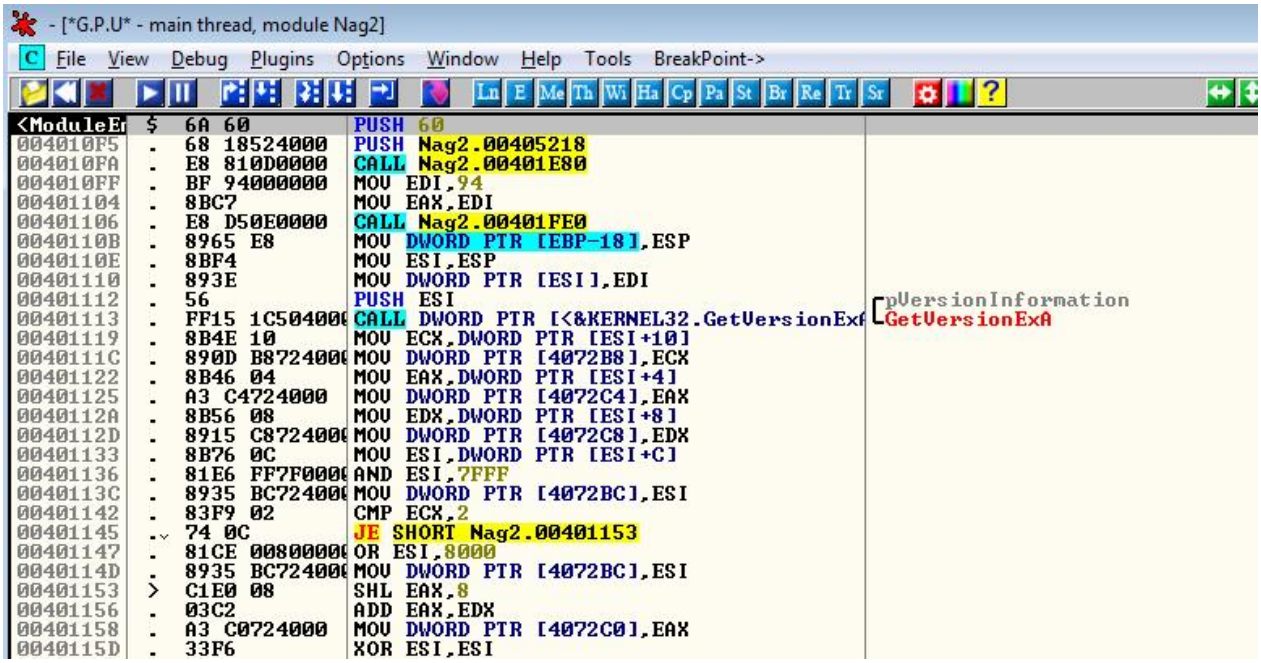
برنامه دوم

در این قسمت برنامه ی هدف Nag2.exe نام دارد که ضمیمه شده ،این مثال شبیه به Nag1.exe هست ولی به روش های دیگری حل خواهد شد. پس از اجرای برنامه :



پس از کلیک بروی دکمه ی Ok وارد پنجره ی اصلی میشویم :

برنامه را درون Olly بارگذاری کنید :



قبل از اینکه به سراغ تکنیک قدیمی جستجوی رشته ها بپردازیم ، میخواهم یک پلاگین با نام IDAFicator را معرفی کنم ، این پلاگین متشکل از تعدادی دکمه است که در سمت راست بالای صفحه قرار دارد که در حقیقت دسترسی ما را به نقاط از پیش تعیین شده راحت میکند برای مثال اگر بروی دکمه ی سبز رنگ (Str) کلیک کنید تمامی رشته های ASCII و Unicode را نمایش میدهد

: همان طور که از شکل این دکمه پیداست کارش عقب و جلو کردن کردن است ،برای مثال اگر شما بروی یک call و یا یک JUMP کلید Enter را بفشارید و وارد آن شوید و بخواهید به محل قبل برگردید با فشردن این دکمه به محل قبل ارجاع داده خواهید شد ،اهمیت این دکمه هنگامی مشهود خواهد شد که درون توابع تو در تو یا در حال آنالیز از نوع استاتیک هستید.

:این دکمه برای یافتن ابتدای تابع فعلی به کار میرود

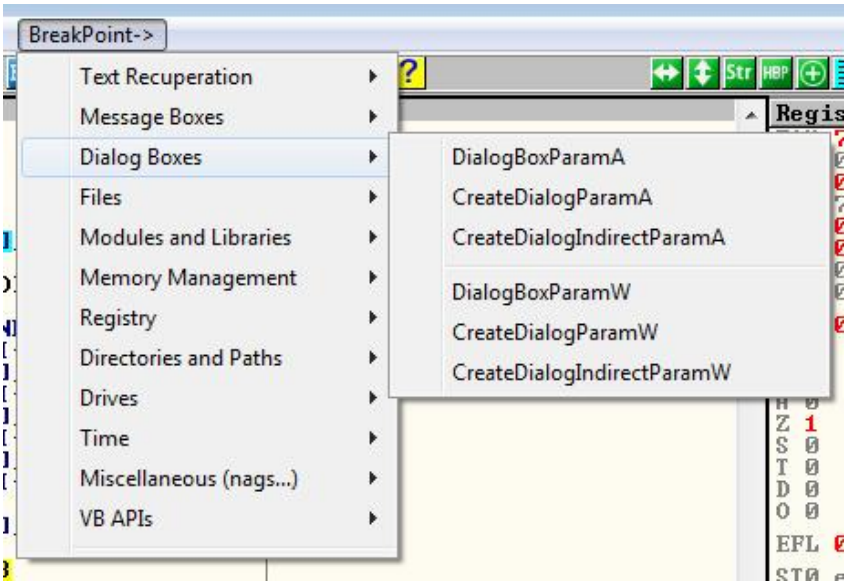
Str: این دکمه برای جستجوی رشته بکار میرود

HBP : این دکمه برای مدیریت نقاط توقف سخت افزاری است

: این دکمه برای نمایش محل فایل در حال دیباگ مورد استفاده قرار میگیرد

: این دکمه برای وارد کردن دستورات اسمبلی به فایل اجرایی کاربرد دارد ،البته آموزش کامل این ورژن از پلاگین قابلیت های دیگری دارد که در فیلم آموزشی (ضمیمه شده) گنجانده شده است.

همچنین در منوی اصلی Olly یک آیتم با نام "Breakpoint-->" گنجانده شده که شامل بسیاری از توابع مهم و قابل استفاده توسط برنامه ها است ،همینطور میتوانید با انتخاب هرکدام به صورت خودکار بروی آن BP قرار دهید:



و در اخر یک گزینه ی دیگر اضافه شده که در آینده درباره ی آن بحث میکنیم ،اکنون دکمه ی (Str) را بفشارید:

```
Found strings are
Address   Disassembly                                  Text string
00401000  MOV EAX,DWORD PTR [ESP+8]                    (Initial CPU selection)
00401031  PUSH Nag2.0040520C                           ASCII "Info"
00401036  PUSH Nag2.00405180                           ASCII "    KillNag    I did this one for your newbies who wants to practice m
00401051  PUSH Nag2.00405178                           ASCII "Nag!"
00401056  PUSH Nag2.00405158                           ASCII "Oh, did u forget this one? :P"
0040107F  PUSH Nag2.00405100                           ASCII "Oh, do u like this program? You are using the trial version.. visit bl
004012DE  PUSH Nag2.00405224                           ASCII "CorExitProcess"
004014F8  PUSH Nag2.00405634                           ASCII "<program name unknown>"
0040152B  PUSH Nag2.00405630                           ASCII "..."
0040155F  PUSH Nag2.00405614                           ASCII "Runtime Error!Program: "
00401571  PUSH Nag2.00405610                           ASCII "..."
0040158D  PUSH Nag2.004055E8                           ASCII "Microsoft Visual C++ Runtime Library"
004020EC  PUSH Nag2.004056D0                           ASCII "user32.dll"
00402107  PUSH Nag2.004056C4                           ASCII "MessageBoxA"
00402118  PUSH Nag2.004056B4                           ASCII "GetActiveWindow"
00402120  PUSH Nag2.004056A0                           ASCII "GetLastActivePopup"
0040213B  PUSH Nag2.00405684                           ASCII "GetUserObjectInformationA"
0040214C  PUSH Nag2.0040566C                           ASCII "GetProcessWindowStation"
00403D8E  MOV EDI,Nag2.0040586C                        ASCII "Unknown security failure detected!"
00403D93  MOV DWORD PTR [EBP-128],Nag2.004057B8        ASCII "A security error of unknown cause has been detected which has corrupted
00403DA4  MOV EDI,Nag2.00405798                        ASCII "Buffer overrun detected!"
00403DA9  MOV DWORD PTR [EBP-128],Nag2.004056F8        ASCII "A buffer overrun has been detected which has corrupted the program's int
00403DD2  PUSH Nag2.00405634                           ASCII "<program name unknown>"
00403E13  PUSH Nag2.00405630                           ASCII "..."
00403E43  MOV EDI,Nag2.00405610                        ASCII "..."
00403E4F  PUSH Nag2.004056EC                           ASCII "Program: "
00403E79  PUSH Nag2.004055E8                           ASCII "Microsoft Visual C++ Runtime Library"
```

بروی خط 4010F7 دبل کلیک کنید تا به محل آن برویم:

```
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >  8B4C24 04     MOV ECX,DWORD PTR [ESP+4]          Case 110 (WM_INITDIALOG) of switch 00401004
00401078  .  6A 40         PUSH 40                            Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
0040107A  .  68 50514000   PUSH Nag2.00405150                  Title = "Nag!!"
0040107F  .  68 00514000   PUSH Nag2.00405100                  Text = "Oh, do u like this program? You are using
00401084  .  51            PUSH ECX                           hOwner
00401085  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
0040108B  .  B8 01000000   MOV EAX,1
00401090  .  C2 1000       RET 10
00401093  >  8B5424 04     MOV EDX,DWORD PTR [ESP+4]          Case 10 (WM_CLOSE) of switch 00401004
00401097  .  6A 00         PUSH 0                             Result = 0
00401099  .  52            PUSH EDX                           hWnd
0040109A  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
004010A0  >  B8 01000000   MOV EAX,1                          Default case of switch 00401021
```

اگر به بالا و پایین این بلوک نگاه کنید میبینید که توسط دو دستور RET قرار داریم ما میخواهیم بدانیم این فراخوانی به سمت Nag Screen از کجا صادر شده است برای این کار به ابتدای این بلوک یعنی آدرس 401074 میرویم :

```
00401007  .v 0F84 86000000 JE Nag2.00401093
0040100D  .  2D 00010000   SUB EAX,100
00401012  .v 74 60         JE SHORT Nag2.00401074
00401014  .  48            DEC EAX
00401015  .v 74 05         JE SHORT Nag2.0040101C
00401017  .  33C0          XOR EAX,EAX                        Default case of switch 00401004
00401019  .  C2 1000       RET 10
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]         Case 111 (WM_COMMAND) of switch 00401004
00401021  .  2D E9030000   SUB EAX,3E9                        Switch (cases 3E9..3EA)
00401026  .v 74 22         JE SHORT Nag2.0040104A
00401028  .  48            DEC EAX
00401029  .v 75 75         JNZ SHORT Nag2.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]          Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                            Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2.0040520C                  Title = "Info"
00401036  .  68 80514000   PUSH Nag2.00405180                  Text = "    KillNag    I did this one for your
0040103B  .  50            PUSH EAX                           hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                           Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                            Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000   PUSH Nag2.00405178                  Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2.00405158                  Text = "Oh, did u forget this one? :P"
0040105B  .  56            PUSH ESI                           hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                             Result = 1
00401064  .  56            PUSH ESI                           hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >  8B4C24 04     MOV ECX,DWORD PTR [ESP+4]          Case 110 (WM_INITDIALOG) of switch 00401004
00401078  .  6A 40         PUSH 40                            Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
0040107A  .  68 50514000   PUSH Nag2.00405150                  Title = "Nag!!"
0040107F  .  68 00514000   PUSH Nag2.00405100                  Text = "Oh, do u like this program? You are using
00401084  .  51            PUSH ECX                           hOwner
00401085  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
0040108B  .  B8 01000000   MOV EAX,1
00401090  .  C2 1000       RET 10
00401093  >  8B5424 04     MOV EDX,DWORD PTR [ESP+4]          Case 10 (WM_CLOSE) of switch 00401004
00401097  .  6A 00         PUSH 0                             Result = 0
00401099  .  52            PUSH EDX                           hWnd
0040109A  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
Jump from 00401012
```

همانطور که در تصویر بالا میبینید این ارجاع از یک دستورالعمل JE واقع در آدرس 401012 صورت میگیرد ،بیایید بروی آن یک BP قرار دهیم و برنامه را اجرا کنیم :

```
00401000  .  8B4424 08     MOV EAX,DWORD PTR [ESP+8]
00401004  .  83E8 10       SUB EAX,10                          Switch (cases 10..111)
00401007  .v 0F84 86000000 JE Nag2.00401093
0040100D  .  2D 00010000   SUB EAX,100
00401012  .v 74 60         JE SHORT Nag2.00401074
00401014  .  48            DEC EAX
00401015  .v 74 05         JE SHORT Nag2.0040101C
00401017  .  33C0          XOR EAX,EAX                         Default case of switch 00401004
00401019  .  C2 1000       RET 10
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]          Case 111 (WM_COMMAND) of switch 00401004
00401021  .  2D E9030000   SUB EAX,3E9                         Switch (cases 3E9..3EA)
00401026  .v 74 22         JE SHORT Nag2.0040104A
00401028  .  48            DEC EAX
00401029  .v 75 75         JNZ SHORT Nag2.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]           Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                             Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2.0040520C                  Title = "Info"
00401036  .  68 80514000   PUSH Nag2.00405180                  Text = "    KillNag    I did this one for your
0040103B  .  50            PUSH EAX                            hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                            Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                             Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000   PUSH Nag2.00405178                  Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2.00405158                  Text = "Oh, did u forget this one? :P"
0040105B  .  56            PUSH ESI                            hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                              Result = 1
00401064  .  56            PUSH ESI                            hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >  8B4C24 04     MOV ECX,DWORD PTR [ESP+4]           Case 110 (WM_INITDIALOG) of switch 00401004
00401078  .  6A 40         PUSH 40                             Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
0040107A  .  68 50514000   PUSH Nag2.00405150                  Title = "Nag!"
```

همانطور که میبینید بروی دستورالعمل JE متوقف شده ایم ،توجه کنید که این فراخوانی مربوط به Nag Screen نیست !! در واقع ما الان در وسط Window's message handler هستیم ، در آموزش های بعد ما مفصل message handlers را کنکاش میکنیم ، اما فعلا این را بدانید که تمام برنامه های GUI ویندوز همگی دارای message handlers هستند و ویندوز پیام های مختلف را از طریقه اینها میفرستد.برای مثال وقتی شما بروی "X" کلیک میکنید (میخواهید پنجره ایی را ببندید) ویندوز از طریق message handler پیامی با مضمون "این کاربر میخواهد پنجره را ببندد" را ارسال میکند.ما میتوانیم این پیام را به گونه ایی Trap (به دام بیندازیم) که به محض ارسال بسته شدن این پیام را نمایش دهد : "You have not saved, are you sure you want to quit ?"

و اما BP ما درست در وسط این عملیات واقع شده ،و اولین پیام ارسالی با برنامه همخوانی ندارد در نتیجه آن را برای نمایش Nag Screen نادیده میگیرد :

```
0040100D  .  2D 00010000   SUB EAX,100
00401012  .v 74 60         JE SHORT Nag2.00401074
00401014  .  48            DEC EAX
00401015  .v 74 05         JE SHORT Nag2.0040101C
00401017  .  33C0          XOR EAX,EAX                         Default case of switch 00401004
00401019  .  C2 1000       RET 10
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]          Case 111 (WM_COMMAND) of switch 0
00401021  .  2D E9030000   SUB EAX,3E9                         Switch (cases 3E9..3EA)
00401026  .v 74 22         JE SHORT Nag2.0040104A
00401028  .  48            DEC EAX
00401029  .v 75 75         JNZ SHORT Nag2.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]           Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                             Style = MB_OK|MB_ICONASTERISK|MB_
00401031  .  68 0C524000   PUSH Nag2.0040520C                  Title = "Info"
00401036  .  68 80514000   PUSH Nag2.00405180                  Text = "    KillNag    I did th
0040103B  .  50            PUSH EAX                            hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                            Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                             Style = MB_OK|MB_ICONHAND|MB_APPL
00401051  .  68 78514000   PUSH Nag2.00405178                  Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2.00405158                  Text = "Oh, did u forget this one
0040105B  .  56            PUSH ESI                            hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                              Result = 1
00401064  .  56            PUSH ESI                            hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >  8B4C24 04     MOV ECX,DWORD PTR [ESP+4]           Case 110 (WM_INITDIALOG) of switc
```

حالا دوباره کلید F9 را بفشارید،میبینید که برنامه دوباره همانجا متوقف میشود ولی اینار پرش انجام میشود یعنی Nag Screen نمایش داده میشود، خب با تغییر Z Flag =0 جلوی نمایش آن را میگیریم :

حال اگر میخواهیم از این BP عبور کنیم چیزی حدود ۳۴ بار پیام به message handler ارسال میشود و بر سر همین آدرس متوقف میشویم ،اگر میخواهید BP شما حفظ شود به ناچار باید ۳۴ بار کلید F9 را بفشارید البته در حین اینکار میتوانید از مراحل ساخته شدن Nag Screen (ساخت دکمه ها و پنجره و...) لذت ببرید یا اینکه میتوانید BP را بردارید و یک بار کلید F9 را فشار دهید تا برنامه اجرا شود :

```
00401004  .   83E8 10      SUB EAX,10                          Switch (cases 10..111)
00401007  .v  0F84 86000000 JE Nag2.00401093
0040100D  .   2D 00010000  SUB EAX,100
00401012  .v  74 60        JE SHORT Nag2.00401074
00401014  .   48           DEC EAX
00401015  .v  74 05        JE SHORT Nag2.0040101C
00401017  .   33C0         XOR EAX,EAX
00401019  .   C2 1000      RET 10
0040101C  >   0FB74424 0C  MOVZX EAX,WORD PTR [ESP+C]          Case 111 (WM_COMMAND) of switch 00401004
00401021  .   2D E9030000  SUB EAX,3E9                         Switch (cases 3E9..3EA)
00401026  .v  74 22        JE SHORT Nag2.0040104A
00401028  .   48           DEC EAX
00401029  .v  75 75        JNZ SHORT Nag2.004010A0
0040102B  .   8B4424 04    MOV EAX,DWORD PTR [ESP+4]           Case 3EA of switch 00401021
0040102F  .   6A 40        PUSH 40                             Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .   68 0C524000  PUSH Nag2.0040520C                  Title = "Info"
00401036  .   68 80514000  PUSH Nag2.00405180                  Text = "    KillNag    I did this one for yo
0040103B  .   50           PUSH EAX                            hOwner
0040103C  .   FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .   B8 01000000  MOV EAX,1
```

۳۴ بار کلیک یا اینکه نقطه ی توقف را بردارید

و سپس داریم :

توجه کنید که پنجره ی اول Nag از بین رفته

پچ کردن برنامه

به صورت معمول با تغییر دستور JE در آدرس 401012 به دستورالعمل Nop میتوان بـه نتیجـه ی دلخـواه رسـید طوری که هیچوقت پرشی صورت نگیرد،اما میخواهم روشی دیگر برای پچ کردن نشان دهم ،مـا میـدانیم کـه وقتـی پیام درست از طریق message handler (یا مدیریت پیام) (البته در مورد پیام دوم،دو تا Nag داریم یکی پنجـره ی اول و دیگری هنگام کلیک بروی دکمه ی exit) فراخوانی میشود،اما چه میشود اگر خط 401074 یک ارجاعی وجـود داشت که Nagرا به خط 401014 هدایت میکرد؟ درست است هیچ وقت Nag اولـی نمـایش داده نمیـشد ،پـس بیایید کار را شروع کنیم :

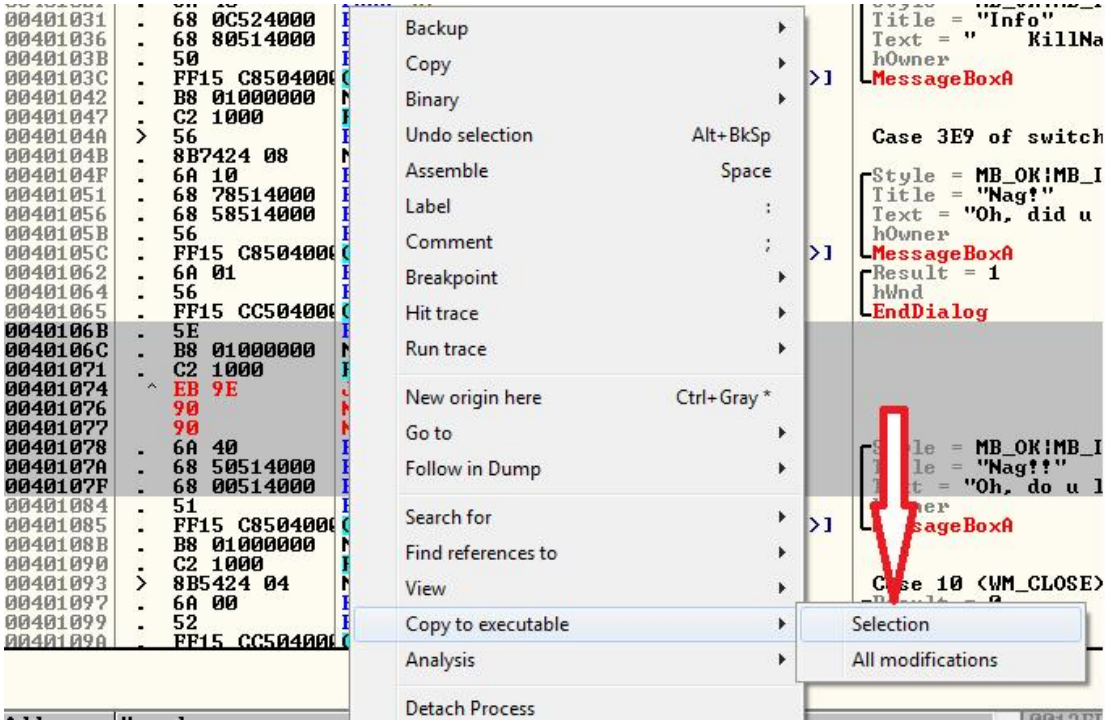
```
00401007  .v 0F84 86000000  JE Nag2.00401093
0040100D  .  2D 00010000    SUB EAX,100
00401012  .v 74 60          JE SHORT Nag2.00401074
00401014  .  48             DEC EAX
00401015  .v 74 05          JE SHORT Nag2.0040101C
00401017  .  33C0           XOR EAX,EAX
00401019  .  C2 1000        RETN 10
0040101C  >  0FB74424 0C    MOVZX EAX,WORD PTR SS:[ESP+C]
00401021  .  2D E9030000    SUB EAX,3E9
00401026  .v 74 22          JE SHORT Nag2.0040104A
00401028  .  48             DEC EAX
00401029  .v 75 75          JNZ SHORT Nag2.00401000
0040102B  .  8B4424 04      MOV EAX,DWORD PTR SS:[ESP+4]
0040102F  .  6A 40          PUSH 40
00401031  .  68 0C524000    PUSH Nag2.0040520C
00401036  .  68 80514000    PUSH Nag2.00405180
0040103B  .  50             PUSH EAX
0040103C  .  FF15 C8504000  CALL DWORD PTR DS:[<&USER32.MessageBoxA>]
00401042  .  B8 01000000    MOV EAX,1
00401047  .  C2 1000        RETN 10
0040104A  >  56             PUSH ESI
0040104B  .  8B7424 08      MOV ESI,DWORD PTR SS:[ESP+8]
0040104F  .  6A 10          PUSH 10
00401051  .  68 78514000    PUSH Nag2.00405178
00401056  .  68 58514000    PUSH Nag2.00405158
0040105B  .  56             PUSH ESI
0040105C  .  FF15 C8504000  CALL DWORD PTR DS:[<&USER32.MessageBoxA>]
00401062  .  6A 01          PUSH 1
00401064  .  56             PUSH ESI
00401065  .  FF15 CC504000  CALL DWORD PTR DS:[<&USER32.EndDialog>]
0040106B  .  5E             POP ESI
0040106C  .  B8 01000000    MOV EAX,1
00401071  .  C2 1000        RETN 10
00401074  ^  EB 9E          JMP SHORT Nag2.00401014
00401076     90             NOP
00401077     90             NOP
00401078  .  6A 40          PUSH 40
0040107A  .  68 50514000    PUSH Nag2.00405150
0040107F  .  68 00514000    PUSH Nag2.00405100
00401084  .  51             PUSH ECX
00401085  .  FF15 C8504000  CALL DWORD PTR DS:[<&USER32.MessageBoxA>]
0040108B  .  B8 01000000    MOV EAX,1
0040109A  .  C2 1000        RETN 10

kernel32.BaseThreadInitThunk
kernel32.BaseThreadInitThunk; Default case of switch
Case 111 (WM_COMMAND) of switch 00401004
Switch (cases 3E9..3EA)
kernel32.BaseThreadInitThunk
Case 3EA of switch 00401021
Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
Title = "Info"
Text = "    KillNag    \n I did this one for your newb
hOwner = 76973388
MessageBoxA

Assemble at 00401076
jmp 401014
Fill with NOP's
Assemble
Cancel

Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
Title = "Nag!!"
Text = "Oh, do u like this program? You are using the
hOwner = NULL
MessageBoxA
```

همانطور که در تصویر بالا میبینید با تغییر دستور در خط 401076 به JMP 401014 برنامه به محض ارجاع از خـط 401012 به اینجا به خط 401014 ارجاع داده:

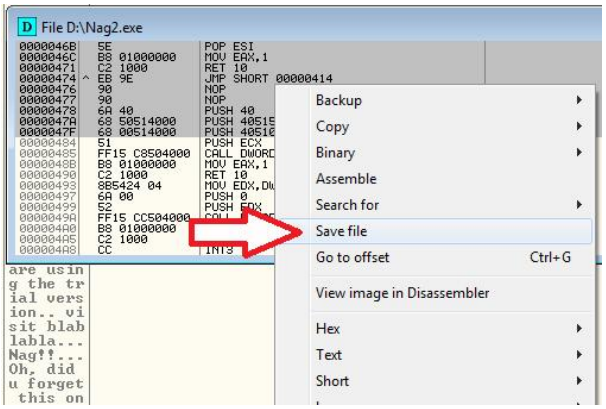
```
00401007  .v 0F84 86000000  JE Nag2.00401093
0040100D  .  2D 00010000    SUB EAX,100
00401012  .v 74 60          JE SHORT Nag2.00401074
00401014  .  48             DEC EAX
00401015  .v 74 05          JE SHORT Nag2.0040101C
00401017  .  33C0           XOR EAX,EAX
00401019  .  C2 1000        RET 10
0040101C  >  0FB74424 0C    MOVZX EAX,WORD PTR [ESP+C]
00401021  .  2D E9030000    SUB EAX,3E9
00401026  .v 74 22          JE SHORT Nag2.0040104A
00401028  .  48             DEC EAX
00401029  .v 75 75          JNZ SHORT Nag2.004010A0
0040102B  .  8B4424 04      MOV EAX,DWORD PTR [ESP+4]
0040102F  .  6A 40          PUSH 40
00401031  .  68 0C524000    PUSH Nag2.0040520C
00401036  .  68 80514000    PUSH Nag2.00405180
0040103B  .  50             PUSH EAX
0040103C  .  FF15 C8504000  CALL DWORD PTR [<&USER32.MessageBoxA>]
00401042  .  B8 01000000    MOV EAX,1
00401047  .  C2 1000        RET 10
0040104A  >  56             PUSH ESI
0040104B  .  8B7424 08      MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10          PUSH 10
00401051  .  68 78514000    PUSH Nag2.00405178
00401056  .  68 58514000    PUSH Nag2.00405158
0040105B  .  56             PUSH ESI
0040105C  .  FF15 C8504000  CALL DWORD PTR [<&USER32.MessageBoxA>]
00401062  .  6A 01          PUSH 1
00401064  .  56             PUSH ESI
00401065  .  FF15 CC504000  CALL DWORD PTR [<&USER32.EndDialog>]
0040106B  .  5E             POP ESI
0040106C  .  B8 01000000    MOV EAX,1
00401071  .  C2 1000        RET 10
00401074  ^  EB 9E          JMP SHORT Nag2.00401014
00401076     90             NOP
00401077     90             NOP
00401078  .  6A 40          PUSH 40

Default case of switch 00401004
Case 111 (WM_COMMAND) of switch 00401004
Switch (cases 3E9..3EA)
Case 3EA of switch 00401021
Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
Title = "Info"
Text = "    KillNag    I did this one for your
hOwner
MessageBoxA
Case 3E9 of switch 00401021
Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
Title = "Nag!"
Text = "Oh, did u forget this one? :P"
hOwner
MessageBoxA
Result = 1
hWnd
EndDialog
Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
```

در واقع این روش فرقی با پچ اولی (تغییر JE به NOP) در خط 401012 ندارد، هدف نشان دادن روش های گوناگون برای پچ کردن است،چون بعضی وقت ها تغییر یک دستورالعمل به Nop ایده ی خوبی نیست ،حالا برنامه را اجرا کنید :

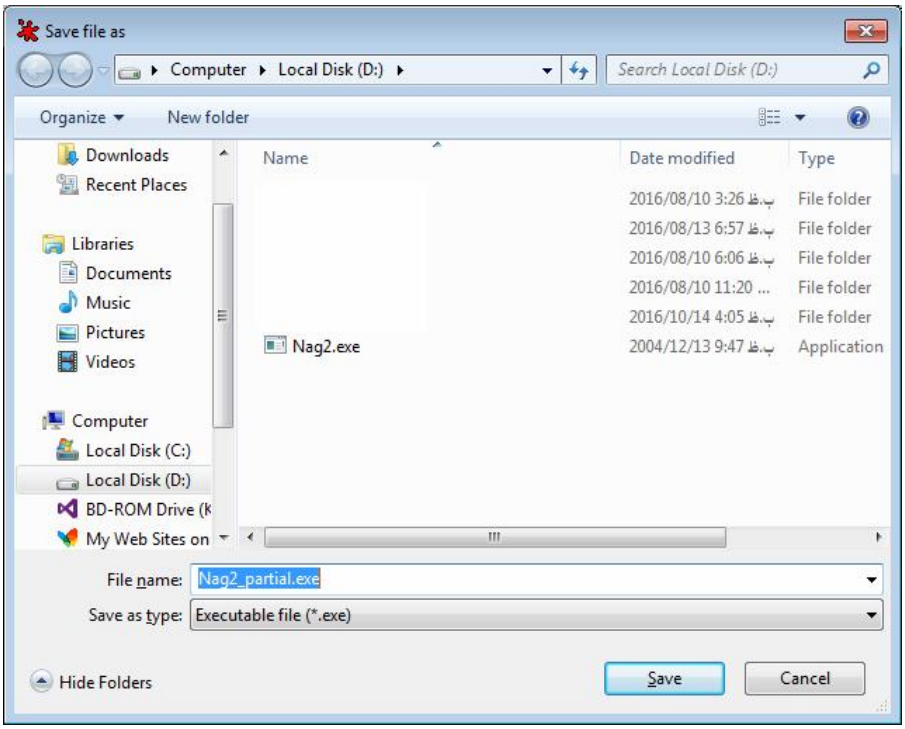
حالا تغییرات را ذخیره کنید با انتخاب محدودی (های لایت شده که شامل تغییرات در کد است) و انتخاب گزینه ی "Copy to executable" -> "Selection":



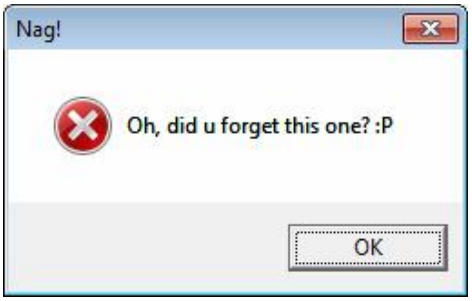
و سپس راست کلیک کرده و Save File را انتخاب کنید:



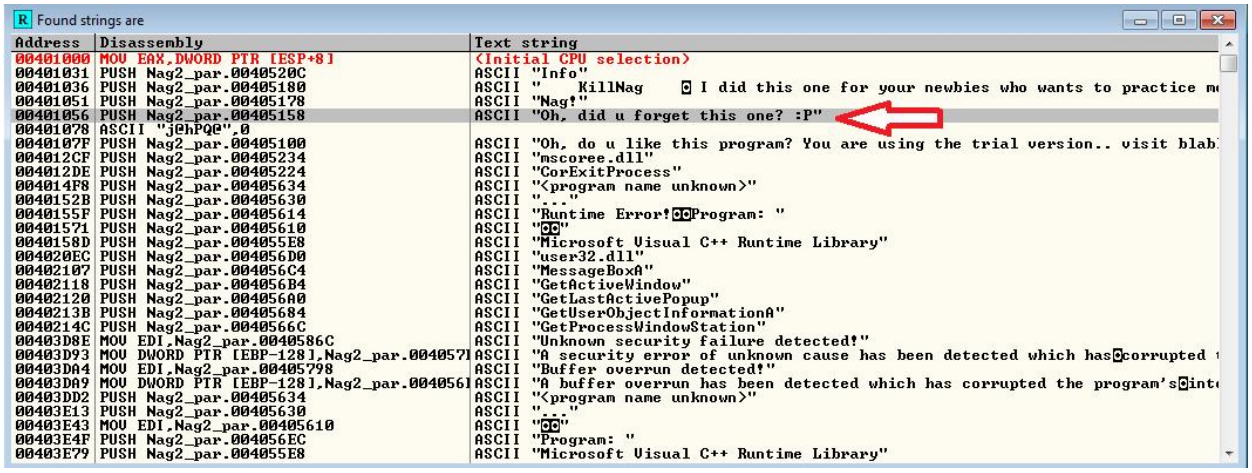
من با اسم Nag2_partial.exe فایل جدید را ذخیره میکنم علت اینکار را می فهمید (Partial به معنای ناتام است)



بسیار عالی،حالا همین فایل پچ شده را در OllyDBG باز کنید ،برنامه را اجرا کنید ،بله برنامه درست پچ شده و در آخر بروی دکمه ی Exit کلید کنید، و داریم:



انتظارش را نداشتید درسته (البته قبلا گفته بودم) ولی به هر حال معمولا برنامه باید بسته میشد ،پس ما یک Nag جدید پیدا کردیم ،بسیار خوب ،برنامه را درون Olly دوباره بارگذاری کنید و با کلیک بروی Str به دنبال رشته مورد نظر میگردیم :



خیلی از برنامه اینکار را میکنند یعنی در موقع اجرا یک Nag را نشان میدهند و موقع بسته شدن هم یکی دیگر را ، بروی متن مورد نظر دبل کلیک کرده تا به محل آن برویم:

```
0040104F  .  6A 10          PUSH 10                                  Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000    PUSH Nag2_par.00405178                   Title = "Nag!"
00401056  .  68 58514000    PUSH Nag2_par.00405158                   Text = "Oh, did u forget this one? :P"
0040105B  .  56             PUSH ESI                                 hOwner
0040105C  .  FF15 C8504000  CALL DWORD PTR [<&USER32.MessageBoxA>]   MessageBoxA
00401062  .  6A 01          PUSH 1                                   Result = 1
00401064  .  56             PUSH ESI                                 hWnd
00401065  .  FF15 CC504000  CALL DWORD PTR [<&USER32.EndDialog>]     EndDialog
0040106B  .  5E             POP ESI
0040106C  .  B8 01000000    MOV EAX,1
00401071  .  C2 1000        RET 10
00401074  >^ EB 9E          JMP SHORT Nag2_par.00401014              Case 110 of switch 00401004
00401076     90             NOP
00401077     90             NOP
00401078  .  6A 40 68 50 5  ASCII "j@hPQ@",0
0040107F  .  68 00514000    PUSH Nag2_par.00405100                   Text = "Oh, do u like this program? You are using
00401084  .  51             PUSH ECX                                 hOwner
00401085  .  FF15 C8504000  CALL DWORD PTR [<&USER32.MessageBoxA>]   MessageBoxA
0040108B  .  B8 01000000    MOV EAX,1
00401090  .  C2 1000        RET 10
00401093  >  8B5424 04      MOV EDX,DWORD PTR [ESP+4]                Case 10 of switch 00401004
00401097  .  6A 00          PUSH 0                                   Result = 0
00401099  .  52             PUSH EDX                                 hWnd
0040109A  .  FF15 CC504000  CALL DWORD PTR [<&USER32.EndDialog>]     EndDialog
004010A0  >  B8 01000000    MOV EAX,1                                Default case of switch 00401021
004010A5  .  C2 1000        RET 10
004010A8     CC             INT3
```

همانطور که در تصویر بالا میبینید NAG دومی درست بعد از Nag اولی فراخوانی شده و وقتی کاربر کلید exit را میفشارد این Nag فراخوانی میشود.شاید بگویید که چرا از روش قبلی استفاده نمیکنیم؟

جواب: اگر به تصویر پایین نگاه کنید متوجه میشوید بعد از نمایش Nag بلافاصله تابع EndDialog فراخوانی میشود پس این روش جواب نمیدهد و nag هیچ وقت بسته نمیشود.خب یک راه دیگر این است که دستورالعمل JE در آدرس 401026 را طوری تغییر دهیم که به ابتدای تابع EndDialog در آدرس 401062 برود درست بعد از تابع MessageBoxA ، به نظر فکر خوبی است پس بیایید آن را امتحان کنیم :

```
00401021  .  2D E9030000    SUB EAX,3E9                              .3EA>
00401026  .v 74 22          JE SHORT Nag2_par.0040104A               از اینجا
00401028  .  48             DEC EAX
00401029  .v 75 75          JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04      MOV EAX,DWORD PTR [ESP+4]                Case 3EA of switch 00401021
0040102F  .  6A 40          PUSH 40                                  Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000    PUSH Nag2_par.0040520C                   Title = "Info"
00401036  .  68 80514000    PUSH Nag2_par.00405180                   Text = "    KillNag    I did this one for y
0040103B  .  50             PUSH EAX                                 hOwner
0040103C  .  FF15 C8504000  CALL DWORD PTR [<&USER32.MessageBoxA>]   MessageBoxA
00401042  .  B8 01000000    MOV EAX,1
00401047  .  C2 1000        RET 10
0040104A  >  56             PUSH ESI                                 Case 3E9 of switch 00401021
0040104B  .  8B7424 08      MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10          PUSH 10                                  Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000    PUSH Nag2_par.00405178                   Title = "Nag!"
00401056  .  68 58514000    PUSH Nag2_par.00405158                   Text = "Oh, did u forget this one? :P"
0040105B  .  56             PUSH ESI                                 hOwner
0040105C  .  FF15 C8504000  CALL DWORD  به اینجا  MessageBoxA>]      MessageBoxA
00401062  .  6A 01          PUSH 1                                   Result = 1
00401064  .  56             PUSH ESI                                 hWnd
00401065  .  FF15 CC504000  CALL DWORD PTR [<&USER32.EndDialog>]     EndDialog
0040106B  .  5E             POP ESI
0040106C  .  B8 01000000    MOV EAX,1
00401071  .  C2 1000        RET 10
```

برای این کار دستور JE SHORT 0040104A واقع در آدرس 401026 را به JE SHORT 00401062 تغییر میدهیم :

```
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]
00401021  .  2D E9030000   SUB EAX,3E9                          Switch (cases 3E9..3EA)
00401026  .  74 3A         JE SHORT Nag2_par.00401062
00401028  .  48            DEC EAX
00401029  .  75 75         JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]
0040102F  .  6A 40         PUSH 40                              MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2_par.0040520C
00401036  .  68 80514000   PUSH Nag2_par.00405180               this one for y
0040103B  .  50            PUSH EAX
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.Messa
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                             Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                              Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000   PUSH Nag2_par.00405178               Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2_par.00405158               Text = "Oh, did u forget this one? :P"
0040105B  .  56            PUSH ESI                             hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                               Result = 1
00401064  .  56            PUSH ESI                             hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >^ EB 9E         JMP SHORT Nag2_par.00401014          Case 110 of switch 00401004
```

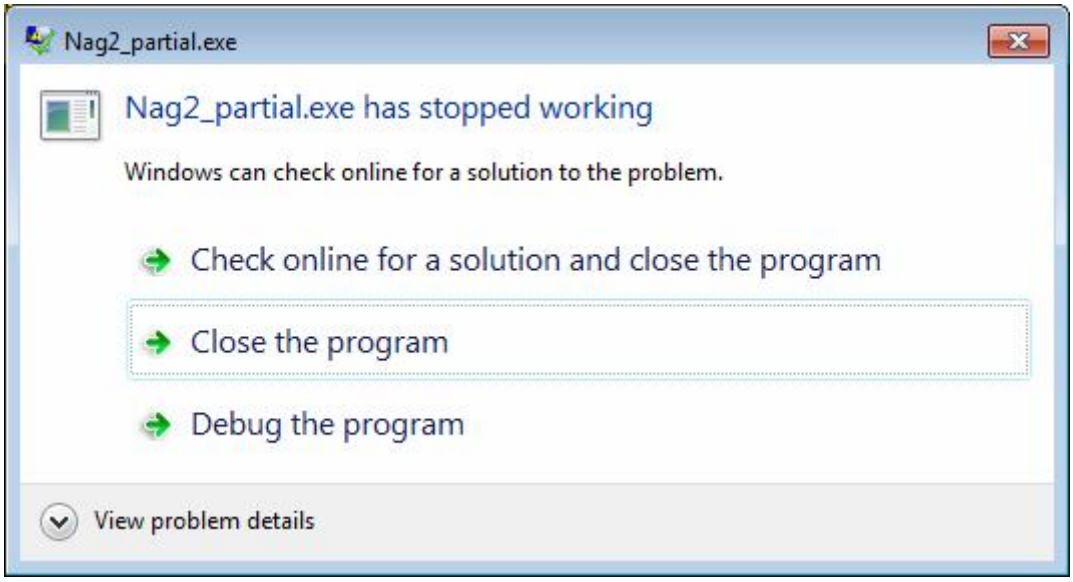
Assemble at 00401026

JE SHORT 00401062

☑ Fill with NOP's

Assemble | Cancel

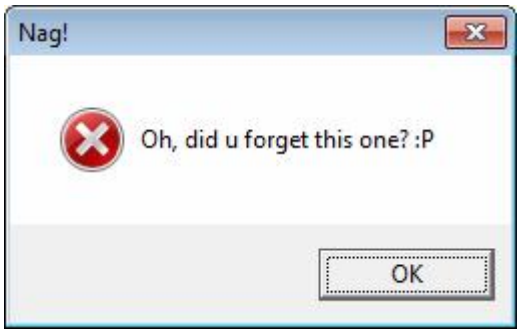
و سپس برنامه را اجرا میکنیم :



امیدوار کننده نبود ،این یعنی جایی از کار اشتباه است،بسیار خوب دوباره برنامه را بارگذاری کنید و آن را اجرا کنید (بدون اعمال پچ):

```
00401026  .  74 22         JE SHORT Nag2_par.0040104A
00401028  .  48            DEC EAX
00401029  .  75 75         JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]            Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                              Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2_par.0040520C               Title = "Info"
00401036  .  68 80514000   PUSH Nag2_par.00405180               Text = "    KillNag    I did this one for your
0040103B  .  50            PUSH EAX                             hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                             Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                              Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000   PUSH Nag2_par.00405178               Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2_par.00405158               Text = "Oh, did u forget this one? :P"
0040105B  .  56            PUSH ESI                             hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                               Result = 1
00401064  .  56            PUSH ESI                             hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
```

با کلید F8 شروع به Trace کردن کنید تا موقعی که Nag را دریافت کنید:



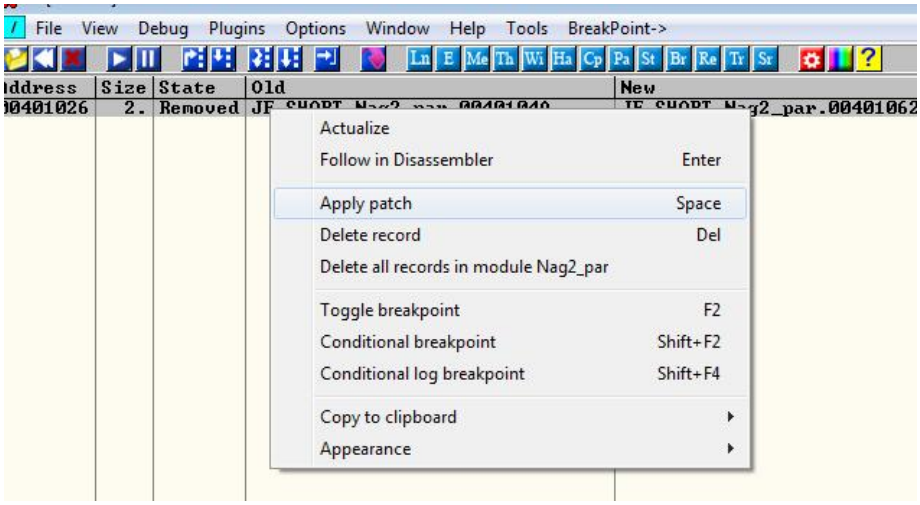
بروی خط 401062 یک BP بگذارید و بروی دکمه OK کلیک کنید ،حال کلید F8 را دوبار بفشارید تا بروی تابع EndDialog قرار بگیرید:

```
00401021  .  2D E9030000   SUB EAX,3E9                          Switch (cases 3E9..3EA)
00401026  v  74 22         JE SHORT Nag2_par.0040104A
00401028  .  48            DEC EAX
00401029  .v 75 75         JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]            Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                              Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2_par.0040520C                Title = "Info"
00401036  .  68 80514000   PUSH Nag2_par.00405180                Text = "    KillNag    I did this one for y
0040103B  .  50            PUSH EAX                             hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                             Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                              Style = MB_OK|MB_ICONHAND|MB_APPLMODAL
00401051  .  68 78514000   PUSH Nag2_par.00405178                Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2_par.00405158                Text = "Oh, did u forget this one? :P"
0040105B  .  56            PUSH ESI                             hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                               Result = 1
00401064  .  56            PUSH ESI                             hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
```

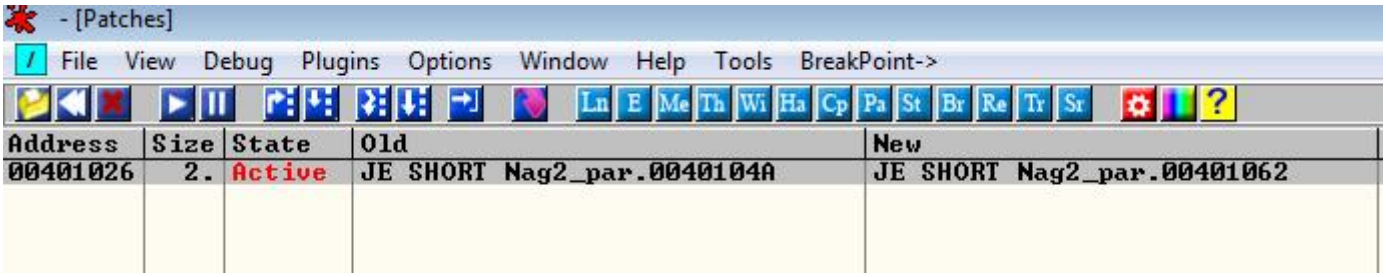
بسیار عالی،حال به پنجره ی Stack نگاه کنید ،همانطور که میبینید ۴ آیتم درون Stack وجود دارد که مسئول مدیریت پنجره ی ما است:

```
0012F924  00170582  hWnd = 00170582 ('KillNag - KiTo',class='#32770')
0012F928  00000001  Result = 1
0012F97C  00000111
0012F980  7580C4E7  RETURN to user32.7580C4E7
0012F984  00170582
0012F988  00000111
0012F98C  000003E9
0012F990  0055055E
0012F994  00000111
0012F998  DCBAABCD
0012F99C  00000000
0012F9A0  0012F9F8
0012F9A4  00000111
0012F9A8  0012FA24
0012F9AC  75825B7C  RETURN to user32.75825B7C from user32.7580C4C4
0012F9B0  00401000  Nag2_par.00401000
```

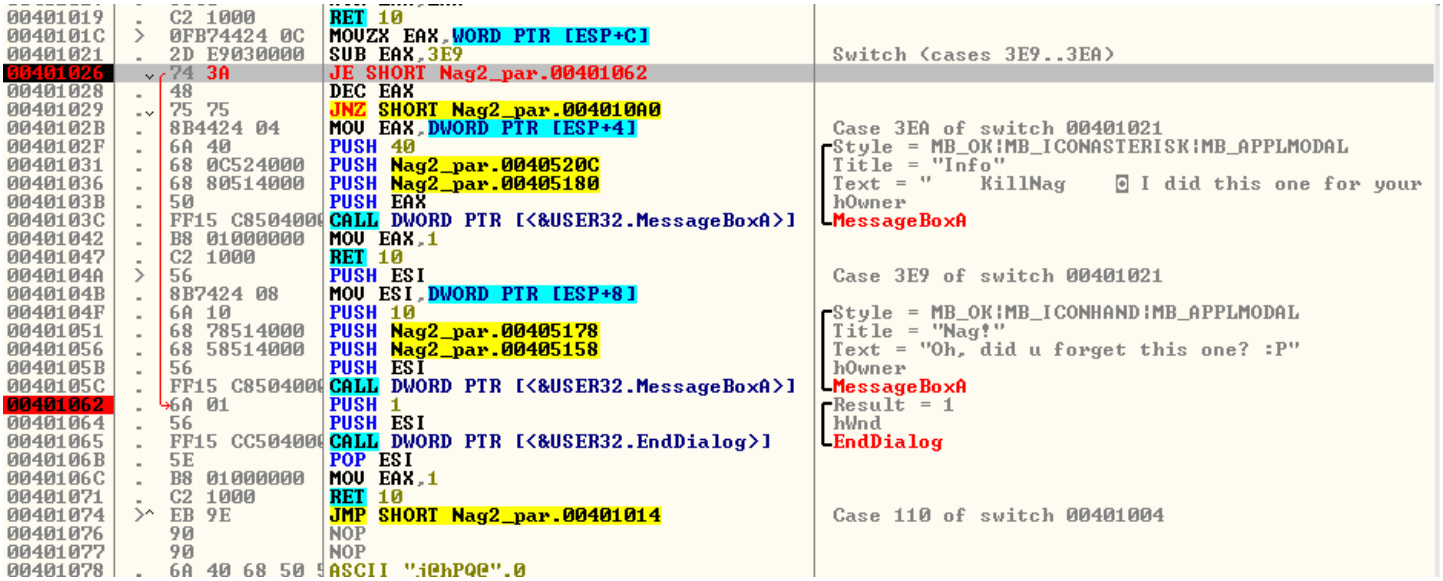
در خط اول ، مقدار Pointer در کد ما 111 00000 است و آدرس بازگشت را در user32 میدهد ، حالا برنامه را دوباره بارگذاری کنید و برنامه را اجرا کنید ، olly در آدرس 401026 متوقف میشود ،سپس کلید Ctrl+P را بفشارید:



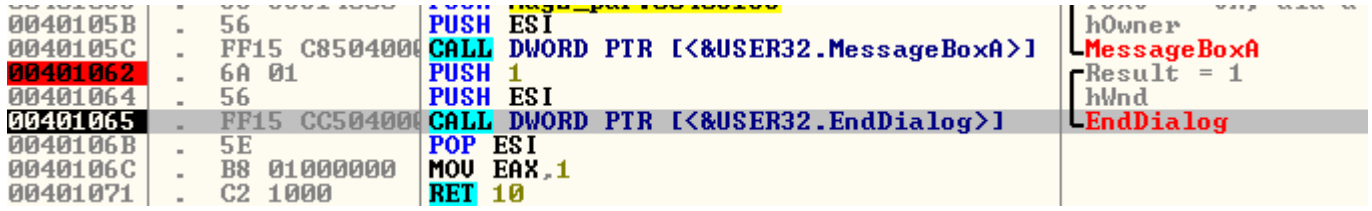
و از پنجره ی Patches نقطه ی توقف را فعال کنید :



حال کلید F9 را بفشارید :



و دوباره کلید F9 تا به آدرس 401062 برسیم و سپس دوبار کلید F8 را بفشارید تا به تابع Enddialog برسیم:



حالا به پنجره ی Stack نگاه کنید:

```
0012F978  00000111  hWnd = 00000111
0012F97C  00000001  Result = 1
0012F980  7580C4E7  RETURN to user32.7580C4E7
0012F984  00AB0446
0012F988  00000111
0012F98C  000003E9
0012F990  006F05CE
0012F994  00000111
```

همانطور که میبینید در مدیریت آدرس بازگشتی به user32 خطایی رخ داده، Pointer اولین خط برابر است با 00000111 ، اگر کمی به بالا نگاه کنید میبینید که قبل از اینکه Nag دومی ساخته بشود ثبات ESI درون پشته Push میشود :

```
00401031  .  68 0C524000   PUSH Nag2_par.0040520C          Title = "Info"
00401036  .  68 80514000   PUSH Nag2_par.00405180          Text = "    KillNag
0040103B  .  50            PUSH EAX                       hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                       Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10                        Style = MB_OK|MB_ICONHAND|M
00401051  .  68 78514000   PUSH Nag2_par.00405178          Title = "Nag!"
00401056  .  68 58514000   PUSH Nag2_par.00405158          Text = "Oh, did u forget th
0040105B  .  56            PUSH ESI                       hOwner
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401062  .  6A 01         PUSH 1                         Result = 1
00401064  .  56            PUSH ESI                       hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
```

مقدار این ثبات حاوی Pointer ما است ،ولی ما با تغییری که ایجاد کرده بودیم ،در واقع از روی این کد گذشتیم ،به هرحال ما برای اجرای صحیح تابع EndDialog به آن احتیاج داریم،بسیار خوب برای اینکار ما از آدرس 40104F تا 40105C را بادستور Nop جایگزین خواهیم کرد برای این منظور منطقه ی مورد نظر را هایلایت کرده و سپس راست کلیک کرده و گزینه ی "Binary" -> "fill with NOPs" را انتخاب کنید:

```
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]      Switch (cases 3E9..3EA)
00401021  .  2D E9030000   SUB EAX,3E9
00401026  .  74 3A         JE SHORT Nag2_par.00401062
00401028  .  48            DEC EAX
00401029  .  75 75         JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]      Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                        Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2_par.0040520C          Title = "Info"
00401036  .  68 80514000   PUSH Nag2_par.00405180          Text = "    KillNag    I did this one for your
0040103B  .  50            PUSH EAX                       hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F  .  6A 10         PUSH 10
00401051  .  68 78514000   PUSH Nag2_par.00405178
00401056  .  68 58514000   PUSH Nag2_par.00405158
0040105B  .  56            PUSH ESI
0040105C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.Me
00401062  .  6A 01         PUSH 1
00401064  .  56            PUSH ESI
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.Er
0040106B  .  5E            POP ESI
0040106C  .  B8 01000000   MOV EAX,1
00401071  .  C2 1000       RET 10
00401074  >^ EB 9E         JMP SHORT Nag2_par.0040101
00401076     90            NOP
00401077     90            NOP
```

Backup
Copy
Binary
Assemble Space
Label :
Comment ;
Breakpoint
Hit trace
Run trace

Edit Ctrl+E
Fill with 00's
Fill with NOPs
Binary copy
Binary paste

و داریم:

```
0040101C  >  0FB74424 0C   MOVZX EAX,WORD PTR [ESP+C]
00401021  .  2D E9030000   SUB EAX,3E9                          Switch (cases 3E9..3EA)
00401026  .  74 3A         JE SHORT Nag2_par.00401062
00401028  .  48            DEC EAX
00401029  .  75 75         JNZ SHORT Nag2_par.004010A0
0040102B  .  8B4424 04     MOV EAX,DWORD PTR [ESP+4]            Case 3EA of switch 00401021
0040102F  .  6A 40         PUSH 40                              Style = MB_OK|MB_ICONASTERISK|MB_APPLMODAL
00401031  .  68 0C524000   PUSH Nag2_par.0040520C               Title = "Info"
00401036  .  68 80514000   PUSH Nag2_par.00405180               Text = "    KillNag    I did this one for
0040103B  .  50            PUSH EAX                             hOwner
0040103C  .  FF15 C8504000 CALL DWORD PTR [<&USER32.MessageBoxA>]  MessageBoxA
00401042  .  B8 01000000   MOV EAX,1
00401047  .  C2 1000       RET 10
0040104A  >  56            PUSH ESI                             Case 3E9 of switch 00401021
0040104B  .  8B7424 08     MOV ESI,DWORD PTR [ESP+8]
0040104F     90            NOP                                  Style
00401050     90            NOP
00401051     90            NOP                                  Title
00401052     90            NOP
00401053     90            NOP
00401054     90            NOP
00401055     90            NOP
00401056     90            NOP                                  Text
00401057     90            NOP
00401058     90            NOP
00401059     90            NOP
0040105A     90            NOP
0040105B     90            NOP                                  hOwner
0040105C     90            NOP                                  MessageBoxA
0040105D     90            NOP
0040105E     90            NOP
0040105F     90            NOP
00401060     90            NOP
00401061     90            NOP
00401062  .  6A 01         PUSH 1                               Result = 1
00401064  .  56            PUSH ESI                             hWnd
00401065  .  FF15 CC504000 CALL DWORD PTR [<&USER32.EndDialog>]  EndDialog
0040106B  .  5E            POP ESI
```

حالا برنامه را اجرا کنید، موفق شدیم ☺ ،هیچ اثری از پنجره ی مزاحم نیست،حالا تمامی تغییرات را ذخیره کنید

تا آموزش بعدی ،موفق باشید