

بسم الله الرحمن الرحيم

سنقوم بعمل Packer أي برنامج يقوم بجمع مجموعة من الملفات في ملف واحد، بحيث يكون الملف الناتج ذاتي الفتح Self eXtracting ما يختصر بـ SFX

في الحقيقة هما برنامجين
الأول لجمع الملفات في ملف واحد
الثاني يقوم بالفتح الذاتي

لا نقوم بأي ضغط أو تشفير فالهدف ليس كتابة برنامج كامل وإنما توضيح فكرة SFX

بما أننا سنقوم بدمج عدة ملفات في ملف واحد نحتاج لحفظ بعض المعلومات لكي نستطيع إعادة الملفات كما كانت.
المعلومات هي

FSign
NFiles
Name1
Size1
Name2
Size2
...
File1
File2
....

عدد الملفات NFiles
إسم كل الملف FName
حجم كل الملف FSize

لتمييز هذه المعلومات نضع قبلها توقيع Fsign و ليكن
مثلاً: "DSFX"

ترتيب المعلومات في الملف كما بالجدول

نبدأ البرنامج الأول والذي يقوم بالخطوات التالية:

- إنشاء جدول معلومات من الملفات المختارة
- إنشاء ملف وكتابة التوقيع و عدد الملفات
- كتابة جدول المعلومات في الملف
- نضيف أول ملف إلى الملف الناتج
- نكرر العملية السابقة مع الملف التالي حتى الأخير
- نغلق الملف و انتهى ☺

```

Const
  fSign='DSFX';
  MaxFiles=2005;

type
  TFileinfo = packed record
    FName: string[255];
    FSize: LongInt;
  end;

var
  Form1: TForm1;
  NFiles : Dword;
  xFiles : Array[0..MaxFiles] of TFileinfo;

procedure MakeDSFX;
Var
  Rw,i : integer;
  Fx,F : File;
  Buff : Array[0..1024*$10-1] of byte;
begin

  NFiles := FilesList.Items.Count;
  For i:=0 To NFiles-1 Do Begin
    xFiles[i].FName := FilesList.Items[i];
    AssignFile(F,xFiles[i].FName);
    FileMode:=0; Reset(F,1);
    xFiles[i].FSize := FileSize(F);
    CloseFile(F);
  End;

  AssignFile(Fx,'DSFX.DSFX');
  ReWrite(Fx,1);
  BlockWrite(Fx,fSign[1],4);
  BlockWrite(Fx,NFiles,4);

  For i:=0 To NFiles-1 Do
    BlockWrite(Fx,xFiles[i],Sizeof(TFileinfo));

  For i:=0 To NFiles-1 Do Begin
    AssignFile(F,xFiles[i].FName);
    FileMode:=0; Reset(F,1);
    Repeat
      BlockRead(F,Buff,Sizeof(Buff),Rw);
      BlockWrite(Fx,Buff,Rw,Rw);
    Until Rw=0;
    CloseFile(F);
  End;

```

ملاحظة: FilesList عبارة عن ListBox نضعه على الفورم و بإستخدام خاصية Items نضيف كل الملفات المراد دمجها

طبعا هناك الكثير من الخيارات و الإضافات لم يتم التطرق لها
نمر الان للبرنامج الثاني و هو يعمل عكس البرنامج الأول:

- فتح الملف DSFX.DSFX
- قراءة ٤ بيت و التآكل من التوقيع DSFX و التوقف في حال عدم وجوده
- قراءة عدد الملفات ثم قراءة جدول معلومات الملفات حسب العدد
- قراءة الملف الأول و إنشائه
- غلق الملف DSFX.DSFX

```
procedure ExtractDSFX;
Var
  Len : LongInt;
  Rw,i : integer;
  Fx,F : File;
  Sign : String;
  Buff : Array[0..1024*$10-1] of byte;
begin
  AssignFile(Fx,'DSFX.DSFX');
  ReSet(Fx,1);
  Sign:='....';
  BlockRead(Fx,Sign[1],4);
  if FSign<>Sign Then Begin
    ShowMessage('NOT DSFX ARCHIVE OR BAD FILE');
    CloseFile(Fx);
    Exit;
  End;
  BlockRead(Fx,NFiles,4);
  FilesList.Items.Clear;
  For i:=0 To NFiles-1 Do Begin
    BlockRead(Fx,xFiles[i],Sizeof(TFileinfo));
  end;
  For i:=0 To NFiles-1 Do Begin
    AssignFile(F,xFiles[i].FName);
    ReWrite(F,1);
    Len := xFiles[i].FSize;
    Repeat
      if Len>Sizeof(Buff) then
        BlockRead(Fx,Buff,Sizeof(Buff),Rw)
      Else BlockRead(Fx,Buff,Len,Rw);
      Len:=Len-Rw;
      BlockWrite(F,Buff,Rw,Rw);
    Until Len=0;

    FilesList.items.Add(xFiles[i].FName);
    CloseFile(F);
  end;

  CloseFile(Fx);
  ShowMessage('Done.');
```

وصلنا لأهم شيء، بدلاً من جمع الملفات في ملف DSFX.DSFX نضيفها في نهاية البرنامج الثاني كما بالشكل

SFX PROGAM
Fsign
NFiles
Name1
Size1
Name2
Size2
...
File1
File2
....

الآن نغير البرنامج الثاني بحيث يفتح نفسه بدلاً من فتح الملف DSFX.DSFX

```
AssignFile(Fx, 'DSFX.DSFX');
```

تصبح

```
AssignFile(Fx, ParamStr(0));
```

عند فتح الملف يكون مؤشر الوضع في بداية الملف التنفيذي، يجب تغييره إلى موضع FSign أي نهاية الملف سابقاً، هناك عدة طرق لمعرفة هذا الموضع منها كتابته بأخر الملف أو حسابه بقراءة PE Header الملف التنفيذي لكننا سنسعمل أسهل طريقة و هي عمل Build للبرنامج نري حجم الملف التنفيذي EXE نم نعود للسورس و نكتب الحجم في ثابت و ليكن مثلاً SFXSize **يجب تكرار هذه العملية عند أي تغيير في سورس البرنامج**

```
Seek(Fx, SFXSize);
```

بما أن البرنامج التنفيذي يقرأ من نفسه و هو ينفذ في نفس الوقت ليس لديه صلاحية تغيير في نفسه و عند فتحه يحدث خطأ، لتفاديه نغير طريقة فتح الملف للقراءة فقط

```
FileMode := 0 ;
```

لتجربة البرنامج نستخدم ملف BAT لإضافة الملف DSFX.DSFX لملفنا التنفيذي، ننشئ ملف TXT نكتب فيه الآتي

```
Copy SFXSTUB.EXE SFX.EXE  
COPY /B SFX.EXE+DSFX.DSFX
```

ونسمي الملف مع تغيير اللاحقة إلى BAT. ثم نقوم بتنفيذه نحصل على الملف التنفيذي SFX.EXE و هو المطلوب ملف تنفيذي ذاتي الفتح...

بقي شيء، هو تغيير البرنامج الأول و جعله يقوم بنفس عمل الملف BAT.
نأخذ نسخة من البرنامج الثاني مع التحديث و نضعه في نفس المجلد و نسميه مثلاً
SFXSTB.EXE

نضيف إلى برنامجنا الثاني هذا الجزء

```
AssignFile(F,'SFXSTUB.EXE');  
Reset(F,1);  
Repeat  
  BlockRead(F, Buff, Sizeof(Buff), Rw);  
  BlockWrite(Fx, Buff, Rw, Rw);  
Until Rw=0;  
CloseFile(F);
```

والحمد لله أنتهي
كان الهدف الأول هو توضيح فكرة SFX – Self eXtracting
البرنامج ينقصه الكثير مثل إختيار الملفات المراد دمجها و التأكد منها و التأكد من و
جود مساحة كافية على القرص و غيرها...

يمكن تحميل كل الملفات من هنا:

www.geocities.com/deltaaziz/DeltaSFX/

لأي استفسار يمكن التواصل بالمنتدى أو عبر البريد الإلكتروني

أخوكم محمد عبد العزيز

DeltaAziz@yahoo.com