Cours de Cracking

(14ième Partie)

Mon objectif : créer un crack en turbo pascal

Dans le 5ième cours de cracking, Smeita nous avait expliqué comment faire son propre patcheur via le language Pascal et son compilateur TPC... Maintenant, nous allons voir comment faire un patch mais plus évolué qui tiendra compte de plus de paramètres avec notamment une verification CRC du fichier a patcher (CRC = Contrôle de Redondance Cyclique) !

Comme dans le 5ième cours de cracking, je vous donne le code commenté, puis un exemple sans les commentaires ! Ici, les textes en rouge sont ceux que vous devez modifier pour faire correctement votre patcheur. Les texte en jaune sont ceux que vous pouvez modifier pour des raisons esthétiques...

Le reste, pas touche !! Essayer de bien comprendre, ca sera deja bien :) Bien sur, il faut savoir que certains passages ne sont pas évidents à expliquer, ainsi j'ai essayé de faire le plus simple possible ;)

---------- début ----------

```
Program Crack;  //juste pour indiquer le nom du prog

Uses CRT, DOS;  //indique quelle bibliothèque on va utiliser des
                //routines graphiques DOS

Const           //indique que l' on va charger des constantes
                //en mémoire pour les réutiliser par la suite

    FileN         : String  = 'progde~1.exe';   //FileN va donc indiquer
                                                //le fichier 'progde~1.exe'
                                                //dès que l' en aurat besion

    BytesToChange : Integer = 2;           //2 indique le nombre d' octects à
                                           //patcher

    FileS         : LongInt = 564132;      //564132 indique la taille du fichier.
                                           //Elle s' éxprime en octects

    A             : Array[1..2] of Record  //indique que l' on va donner
```

```
                                        //de modification allant de
                                        //1 à 2, ici en octects


      A : Longint;
      B : Byte;


End =

(
(A:$303e6;B:$90),  //adresses héxadécimales à modifier et leurs octects
(A:$47274;B:$90)   //attention !! sur la derniere ligne=> pas de virgule !!!!
);


Var

  F     : File;
  Ch    : Char;
  Attr  : Word;
  I     : LongInt;


Begin

Textcolor(13);    //pour choisir la couleur de ce qui va s' afficher à l' écran

clrscr;   //efface l' écran, WriteLn ne se charge donc que d' afficher un texte,
          //juste une question de présentation



  Writeln('cRACk fOR     : Prog de merde v0.0');
  Writeln('cRACk tYPE    : Auto-registration!');
  Writeln('cRACKED bY    : NoOne');
  Writeln('');
  Writeln('cONTACT       : monmail@cacamail.com');
  Writeln('wEBsITE       : www.monsite.com');
  Writeln('gREETiNGz     : Everybody !');
  Writeln('');



If (FSearch(FileN,FExpand(FileN))='') then   //cherche le fichier FileN,
                                             //S'il est non présent,
                                             //alors => Message d'erreur...

     Begin

     Textcolor(12);
     Writeln('oOOPS!: File ',FileN,' not found !!!');
     Writeln('-Current bad size has: ', FileSize (F),' bytes.');
     Writeln('-Good size should be : ', FileS ,' bytes.');
     Writeln('Crack aborted...');
     Halt(1);

     end;
```

```
  Assign(F,FileN);
  Reset(F,1);         //pour ouvrir un fichier

  TextColor(8);
  Write('Checking FileSize...');

  If FileSize(F)<>FileS then  //FileSize verifie la taille du fichier...

     begin  //début du mesage d' erreur
            //(si le fichier cible n'a pas la bonne taille..)

     TextColor(12);
     Writeln('ERROR!');
     TextColor(07);
     Writeln('');
     Textcolor(12);
     Writeln('File ',FileN,' has an invalid Size !!!');
     Writeln('Crack aborted...');

     Close(F);   //on ferme le fichier
     Halt(1);    //on arrete le prog
     end         //fin du passage affichant une erreur...

Else

     begin            //ce qui suit est le "patchage" des octets...

     Writeln('OK');   //inscrit OK juste après 'Checking FileSize...'

     end;  //fin de la reconnaissance le la taille du fichier


  TextColor(8);
  Write('Cracking ',FileN,'...');   //Et hop!, on réutilise encore ',FileN,'


  For I := 1 to BytesToChange do    //change les octect de 1 à...'x'

     begin
     Seek(F,A[I].A);     //repere l'offset a patcher

     Ch:=Char(A[I].B);   //on identifie la nouvelle valeur de l'offset
                         //et on la stock dans 'Ch'

     Blockwrite(F,Ch,1); //ouf! Après toutes les conditions remplis, le prog va
                         //enfin pouvoir modifier le fichier cible en inscrivant
                         //des octects par blocs de 1, tels qu'ils sont indiqués
                         //par (A:$303e6;B:$00) et (A:$47274;B:$00) dans l'exemple.

     end;

Writeln('OK'); //inscrit OK juste après 'Cracking ',FileN,'...'

Close(F)       // Pour fermer le fichier

  TextColor(10);
  Writeln('CrACk Successful!');  //message de réussite du patchage :))
```

```
end.
```

---------- fin

[interlude de Smeita...]
Voila !! Vous pourrez constatez que c'est deja plus consistant que le premier patch !
En plus, la verification CRC permet d'eviter de patcher une mauvaise version d'un programme...

Moi je dis : Vive Static REvenge :)) !!

Bon, maintenant, on vous donne un apercu de ce que ca donne sans les commentaires...
Le compilateur turbo pascal est ici

[...Fin d'interlude...]

---------- début

```
Program Crack;
Uses CRT, DOS;
Const

    FileN         : String  = 'progde~1.exe';
    BytesToChange : Integer = 2;
    FileS         : LongInt = 564132;

    A             : Array[1..2] of Record

                    A : Longint;
                    B : Byte;
End =

(
(A:$303e6;B:$90),
(A:$47274;B:$90)
);


Var
  F     : File;
  Ch    : Char;
  Attr  : Word;
  I     : LongInt;


Begin

Textcolor(13);
```

```
clrscr;


  Writeln('cRACk fOR     : Prog de merde v0.0');
  Writeln('cRACk tYPE    : Auto-registration!');
  Writeln('cRACKED bY    : NoOne');
  Writeln('');
  Writeln('cONTACT       : monmail@cacamail.com');
  Writeln('wEBsITE       : www.monsite.com');
  Writeln('gREETiNGz     : Everybody !');
  Writeln('');


If (FSearch(FileN,FExpand(FileN))='') then

     begin

     Textcolor(12);
     Writeln('oOOPS!: File ',FileN,' not found !!!');
     Writeln('-Current bad size has: ', FileSize (F),' bytes.');
     Writeln('-Good size should be : ', FileS ,' bytes.');
     Writeln('Crack aborted...');
     Halt(1);

     end;


  Assign(F,FileN);
  Reset(F,1);

  TextColor(8);
  Write('Checking FileSize...');

  If FileSize(F)<>FileS then

     begin

     TextColor(12);
     Writeln('ERROR!');
     TextColor(07);
     Writeln('');
     Textcolor(12);
     Writeln('File ',FileN,' has an invalid Size !!!');
     Writeln('Crack aborted...');

     Close(F);
     Halt(1);
     end

else

     begin
     Writeln('OK');
     end;

  TextColor(8);
  Write('Cracking ',FileN,'...');

  For I := 1 to BytesToChange do
```

```
     begin
     Seek(F,A[I].A);
          Ch:=Char(A[I].B);
     Blockwrite(F,Ch,1);
     end;

Writeln('OK');
Close(F)

  TextColor(10);
  Writeln('CrACk Successful!');

end.
```

---------- fin ----------

N.B: certains progs, vont chercher un fichier dat et refusent de démarrer. Alors, pour ne pas vous galérer, une petite ligne Pascal va vous permettre de créer un fichier vide. Pour cela il faut d' abord déclarer une variable puis l'exploiter de la façon suivante:

```
Var
k: text;
```

puis:

```
Assign(k,'Info.dat');
Rewrite(k);
```

Ceci est à placer avant la ligne " If (FSearch(FileN,FExpand(FileN))='') then ".

Allez, bonne chance à tous et à bientôt !