

www.joseouin.fr

JO-OBfuscator

Le protecteur des macros VBA

www.joseouin.fr **Fichier Excel pour l'obfuscation
des macros VBA**

Obfuscation : techniques mises en œuvre pour transformer le code source afin de le rendre illisible pour un être humain tout en le conservant entièrement opérationnel.



www.joseouin.fr



www.joseouin.fr

JO-OBfuscator

Le protecteur des macros VBA

Fichier Excel pour l'obfuscation des macros VBA

1- Définition de l'obfuscation

En programmation informatique, on appelle obfuscation les techniques mises en œuvre pour transformer le code source afin de le rendre illisible pour un être humain après une forme de compilation, tout en le conservant entièrement opérationnel. L'objectif est de protéger les droits de propriété sur une application informatique en empêchant la reconstitution ou la transformation d'un code source compréhensible à partir du code d'origine.

Synonyme d'obfuscation : brouillage

1

Sélectionner un dossier : Parcourir...



A propos de JO-Obfuscator...

Dossier de sauvegarde de la macro à traiter : O:_Travail_INFO1\DOC_DEV_QCM_VBA\Dev_Obfuscator_v_Finale_Publish_PUBLI

Dossier de sauvegarde de la macro traitée : O:_Travail_INFO1\DOC_DEV_QCM_VBA\Dev_Obfuscator_v_Finale_Publish_PUBLI

JO-Obfuscator effectue une sauvegarde de la macro initiale Sub() ou Function() [nom du la macro traitée [nom du fichier : nom_de_la_macro_JO-OB.txt]

Ainsi toutes les macros sont archivées avec leurs noms dans des fichiers séparés (macro

2

Lancer JO-OB pour Sub() ou Function()

Sub Demo_Obfuscator_video ()

Variables VBA : 6

Ch

3

Ouvrir une macro initiale ou traitée

Message1

"JO-OBfuscator"

Message2

"Ceci est une démonstration"

Titre

MesON

Mode d'emploi :

Sélectionner un dossier de travail
Produit

2- Description du classeur Excel JO-OBfuscator

JO-OBfuscator est un classeur Excel permettant de protéger le code source VBA d'une macro Excel en le rendant quasiment illisible. Pour ce faire, cet outil effectue un cryptage du nom des variables, transforme les chaînes de caractères et supprime les commentaires ainsi que l'indentation des lignes de code. Toutes ces opérations rendent le code source de la macro quasiment illisible.

De plus JO-OBfuscator sauvegarde le code source initial dans un fichier au format texte (.txt) ainsi que le code source après obfuscation dans un fichier séparé (.txt). Il dresse également la liste des variables cryptées et des chaînes de caractères transformées.

Ainsi toutes les macros sont archivées avec leurs noms dans des fichiers séparés (macros initiales et macros traitées). Cet archivage peut être utile, par exemple, pour retrouver facilement une macro que l'on souhaite utiliser dans un autre projet VBA Excel.

Exemple

Un exemple de macro Excel avant et après le traitement pas JO-Obfuscator.

```
Sub Demo_Obfuscator_Video()  
'-- Ceci est une macro de démonstration.  
'-- Déclaration des variables  
Dim Message1 As String  
Dim Message2 As String  
Dim Titre As String  
Dim MesON As Integer  
Dim Kp As Integer  
Dim Rep As Integer  
'-- Fin de déclaration des variables  
  
Message1 = "JO-Obfuscator" & vbCrLf & vbCrLf  
Message2 = "Ceci est une démonstration de JO-Obfuscator." & vbCrLf & vbCrLf  
'-- Remarque : vbCrLf est une constante VBA (Cr : Carriage return et Lf : Line feed)  
Kp = Len(Message1)  
  
If Kp = 17 Then  
    '-- On met la variable MesON à 1  
    MesON = 1  
Else  
    '-- On met la variable MesON à 0  
    MesON = 0  
End If  
  
'-- Contenu du message :  
Titre = "JO-Obfuscator : Vidéo de démonstration"  
Message = Message1  
Message = Message & Message2  
Message = Message & "Téléchargez la version originale sur www.joseouin.fr" & vbCrLf & vbCrLf  
Message = Message & "Auteur : José OUIIN - Site Internet : www.joseouin.fr"  
  
'-- Affichage de la boîte si le classeur Excel n'est pas en lecture seule et si MesON = 1  
If ThisWorkbook.ReadOnly = False And MesON = 1 Then  
    Rep = MsgBox(Message, vbInformation + vbOKOnly, Titre)  
End If  
'-----  
End Sub
```

Remarques :

Le code source ci-dessus comporte des commentaires, des lignes vierges ainsi qu'une indentation des lignes de codes. Tout ceci participe à la lisibilité du code source et c'est pour cette raison que JO-Obfuscator retire tous ces éléments.

Résultat : Contenu de la macro Excel après le traitement pas JO-Obfuscator :

```

Sub Demo_Obfuscator_Video()
Dim JOOBetjKa7lPQURVDETaHdF16 As String
Dim JOOBetjKa7lPQURVDFTaHdF16 As String
Dim JOOBetjKa7sTRTVTaHdF16 As String
Dim JOOBetjKa7lPQesTaHdF16 As Integer
Dim JOOBetjKa7jlTaHdF16 As Integer
Dim JOOBetjKa7qPNTaHdF16 As Integer
JOOBetjKa7lPQURVDETaHdF16 = Chr(74) & Chr(79) & Chr(45) & Chr(79) & Chr(66) & Chr(102) & Chr(117) & Chr(115) & Chr(99) & Chr(97) & Chr(116) & Chr(111) & Chr(114) & vbCrLf & vbCrLf
JOOBetjKa7lPQURVDFTaHdF16 = Chr(67) & Chr(101) & Chr(99) & Chr(105) & Chr(32) & Chr(101) & Chr(115) & Chr(116) & Chr(32) & Chr(117) & Chr(117) & Chr(110) & Chr(110) & Chr(101) & Chr(32) & Chr(100) & Chr(233) & Chr(109) & Chr(111) & Chr(110) & Chr(115) & Chr(116) & Chr(114) & Chr(97) & Chr(116) & Chr(105) & Chr(111) & Chr(110) & Chr(32) & Chr(100) & Chr(101) & Chr(32) & Chr(74) & Chr(79) & Chr(45) & Chr(79) & Chr(66) & Chr(102) & Chr(117) & Chr(115) & Chr(99) & Chr(97) & Chr(116) & Chr(111) & Chr(114) & Chr(46) & vbCrLf & vbCrLf
JOOBetjKa7jlTaHdF16 = Len(JOOBetjKa7lPQURVDETaHdF16)
If JOOBetjKa7jlTaHdF16 = 17 Then
JOOBetjKa7lPQesTaHdF16 = 1
Else
JOOBetjKa7lPQesTaHdF16 = 0
End If
JOOBetjKa7sTRTVTaHdF16 = Chr(74) & Chr(79) & Chr(45) & Chr(79) & Chr(66) & Chr(102) & Chr(117) & Chr(115) & Chr(99) & Chr(97) & Chr(116) & Chr(111) & Chr(114) & Chr(32) & Chr(58) & Chr(32) & Chr(86) & Chr(105) & Chr(100) & Chr(233) & Chr(111) & Chr(32) & Chr(100) & Chr(101) & Chr(32) & Chr(100) & Chr(233) & Chr(109) & Chr(111) & Chr(110) & Chr(115) & Chr(116) & Chr(114) & Chr(97) & Chr(116) & Chr(105) & Chr(111) & Chr(110)
Message = JOOBetjKa7lPQURVDETaHdF16
Message = Message & JOOBetjKa7lPQURVDFTaHdF16
Message = Message & Chr(84) & Chr(233) & Chr(108) & Chr(233) & Chr(99) & Chr(104) & Chr(97) & Chr(114) & Chr(103) & Chr(101) & Chr(122) & Chr(32) & Chr(108) & Chr(97) & Chr(32) & Chr(118) & Chr(101) & Chr(114) & Chr(115) & Chr(105) & Chr(111) & Chr(110) & Chr(32) & Chr(111) & Chr(114) & Chr(105) & Chr(103) & Chr(105) & Chr(110) & Chr(97) & Chr(108) & Chr(101) & Chr(32) & Chr(115) & Chr(117) & Chr(114) & Chr(32) & Chr(119) & Chr(119) & Chr(119) & Chr(46) & Chr(106) & Chr(111) & Chr(115) & Chr(101) & Chr(111) & Chr(117) & Chr(105) & Chr(110) & Chr(46) & Chr(102) & Chr(114) & vbCrLf & vbCrLf
Message = Message & Chr(65) & Chr(117) & Chr(116) & Chr(101) & Chr(117) & Chr(114) & Chr(32) & Chr(58) & Chr(32) & Chr(74) & Chr(111) & Chr(115) & Chr(233) & Chr(32) & Chr(79) & Chr(85) & Chr(73) & Chr(78) & Chr(32) & Chr(45) & Chr(32) & Chr(83) & Chr(105) & Chr(116) & Chr(101) & Chr(32) & Chr(73) & Chr(110) & Chr(116) & Chr(101) & Chr(114) & Chr(110) & Chr(101) & Chr(116) & Chr(32) & Chr(58) & Chr(32) & Chr(119) & Chr(119) & Chr(119) & Chr(46) & Chr(106) & Chr(111) & Chr(115) & Chr(101) & Chr(111) & Chr(117) & Chr(105) & Chr(110) & Chr(46) & Chr(102) & Chr(114)
If ThisWorkbook.ReadOnly = False And JOOBetjKa7lPQesTaHdF16 = 1 Then
JOOBetjKa7qPNTaHdF16 = MsgBox(Message, vbInformation + vbOKOnly, JOOBetjKa7sTRTVTaHdF16)
End If
End Sub

```

JO-Obfuscator dresse un récapitulatif à la fin du fichier traité (récapitulatif destiné uniquement à l'auteur de la macro Excel) :

```

----- Récapitulatif -----
Liste des noms des variables et des constantes de : Sub Demo_Obfuscator_Video ()
-----
Variables de la macro :
Message1 ---> JOOBetjKa7lPQURVDETaHdF16
Message2 ---> JOOBetjKa7lPQURVDFTaHdF16
Titre ---> JOOBetjKa7sTRTVTaHdF16
MesON ---> JOOBetjKa7lPQesTaHdF16
Rep ---> JOOBetjKa7qPNTaHdF16
Kp ---> JOOBetjKa7jlTaHdF16
-----
Constantes de la macro :
"JO-Obfuscator"
"Ceci est une démonstration de JO-Obfuscator."
"JO-Obfuscator : Vidéo de démonstration"
"Téléchargez la version originale sur www.joseouin.fr"
"Auteur : José OUIIN - Site Internet : www.joseouin.fr"

```


3- Propriétés de JO-OBfuscator

Cet outil effectue les actions suivantes :

1. Cryptage du nom des variables ;
2. Transformation des chaînes de caractères ;
3. Suppression de l'indentation des lignes de code ;
4. Sauvegarde du code source initial dans un fichier au format texte (.txt) ;
5. Sauvegarde du code source après transformation dans un fichier séparé (.txt) ;
6. Listage du nom des variables (initiales et transformées) et des chaînes de caractères.

4- Conseils et remarques sur l'utilisation de JO-OBfuscator



Limitier la longueur des chaînes de caractères au niveau de l'éditeur VBA

JO-OBfuscator transforme les chaînes de caractères à l'aide de la fonction Chr() et des codes Ascii des caractères de la chaîne. Par exemple, le mot **VBA** sera remplacé par **Chr(86) & Chr(66) & Chr(65)**.

Ces **3** caractères généreront : 3 x **7** caractères + 4 espaces + 2 **&** = **27** caractères.

Dans l'éditeur VBA, une ligne de code ne peut pas dépasser 1024 caractères (à moins d'utiliser l'underscore « _ »).

Il ne faut donc pas dépasser 1024/9, soit 110 caractères environ par ligne de code.

Mon conseil : limiter les lignes de codes à 90 caractères.

Attention ! La ligne de code suivante (suite à l'obfuscation) dépassera les 1024 caractères si vous saisissez ceci :

Oh non ! 😞

Message = "JO-Obfuscator : Ceci est un message qui assez long qu'il faudra couper en deux au niveau de votre code car la transformation en Chr() augmente le nombre de caractères de la ligne de code."

Solutions

Solution 1 :

Couper votre ligne en 2, voire plus, de cette manière :

Ah oui ! 😊

Message = "JO-Obfuscator : Ceci est un message qui assez long qu'il faudra couper en deux au niveau"

Message = Message & " de votre code car la transformation en Chr() augmente le nombre de caractères de la ligne de code."

Solution 2 :

Couper votre ligne en 2, voire plus, en utilisant plusieurs variables :

Ah oui ! 😊

Ligne1 = "JO-Obfuscator : Ceci est un message qui assez long qu'il faudra couper en deux au niveau"

Ligne2 = " de votre code car la transformation en Chr() augmente le nombre de caractères de la ligne de code."

Message = Ligne1 & Ligne2

Solution 3 :

Couper votre ligne en 2, voire plus, en utilisant l'underscore (tiret bas de la touche « 8 » : " _ ") :

Ah oui ! 😊

Message = "JO-Obfuscator : Ceci est un message qui assez long qu'il faudra couper en deux au niveau" & _
" de votre code car la transformation en Chr() augmente le nombre de caractères de la ligne de code."



www.joseouin.fr Limiter la longueur des instructions au niveau de l'éditeur VBA

Il faut utiliser le symbole underscore (tiret bas de la touche « 8 » : "_") pour une instruction assez longue.

On y gagne en lisibilité du code pour le développeur et cela permet de rester sous la barre des 1024 caractères au niveau de chaque ligne de code.

Oh non ! 😞

```
Feuill1.Range(Plg_Data_A).Sort Key1:=Feuill1.Range(Cell_Gr_A), Order1:=xlDescending, Header:=xlYes, OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom, DataOption1:=xlSortNormal
```

Ah oui ! 😊

```
Feuill1.Range(Plg_Data_A).Sort Key1:=Feuill1.Range(Cell_Gr_A), _  
Order1:=xlDescending, Header:=xlYes, OrderCustom:=1, MatchCase:=False, _  
Orientation:=xlTopToBottom, DataOption1:=xlSortNormal
```



Déclarer les variables et les chaînes de caractères

Il faut déclarer vos variables si vous souhaitez qu'elles soient cryptées.

Oh non ! 😞

```
Message = "Bonjour à tous"
```

```
Longueur = 7
```

Solution :

Déclarez les variables, sinon elles ne seront pas modifiées par JO-OBfuscator

Ah oui ! 😊

```
Dim Message As String  
Dim Longueur As Integer
```

```
Message = "Bonjour à tous"
```

```
Longueur = 7
```

Si vous rencontrez un problème avec une variable au niveau du cryptage par JO-OBfuscator, retirez-la des déclarations et elle ne sera pas modifiée par cet outil. C'est le cas notamment des variables i, j, k, m, n que vous pouvez utiliser dans les boucles While ou For et qui sont des variables qui sont juste-là pour le « fonctionnement » des boucles.



Attention aux noms de variables trop courts ou ambigus

Le meilleur endroit pour cacher un arbre c'est dans la forêt, mais attention de ne pas le confondre vous-même avec un autre arbre...

Exemple :

Oh non !

Dim Book As String ← Ok pour la variable « Book » car JO-OBfuscator distingue « Book » et « book » (il respecte la casse)
Dim ReadOnly As Boolean
Dim k As Integer

```
If ThisWorkbook.k.ReadOnly = False And MesON = 1 Then
    Rep = MsgBox(Message, vbInformation + vbOKOnly, Titre)
End If
```

k est déclaré comme variable donc toutes les occurrences du caractère « k » seront remplacées ! Même chose pour la variable ReadOnly.

Solutions

Solution 1 :

Tenir compte du fait que JO-OBfuscator respecte la casse (« ReadOnly » n'est pas identique à « readonly »).

Ah oui !

Dim Book As String ← Ok pour la variable « Book » car JO-OBfuscator distingue « Book » et « book » (il respecte la casse)
Dim readonly As Boolean
Dim K As Integer

```
If ThisWorkbook.ReadOnly = False And MesON = 1 Then
    Rep = MsgBox(Message, vbInformation + vbOKOnly, Titre)
End If
```

Solution 2 :

Choisir des variables à la Française et comportant plusieurs lettres : MonK, LectureSeule, Livre, etc.

Ah oui !

Dim Livre As String
Dim LectureSeule As Boolean
Dim MonK As Integer

```
If ThisWorkbook.ReadOnly = False And MesON = 1 Then
    Rep = MsgBox(Message, vbInformation + vbOKOnly, Titre)
End If
```

Solution 3 :

Ne pas déclarer les variables que l'on ne peut ou ne veut pas crypter. Par exemple, si la variable « k » n'est pas déclarée, elle restera telle quelle dans le code source de la macro et les caractères « k » du mot « Workbook » ne seront pas modifiés.

Ah oui !

Dim Book As String
Dim readonly As Boolean
~~Dim k As Integer~~

```
If ThisWorkbook.ReadOnly = False And MesON = 1 Then
    Rep = MsgBox(Message, vbInformation + vbOKOnly, Titre)
End If
```

5- Mode d'emploi de JO-OBfuscator

1

Sélectionner un dossier de travail

Sélectionner un dossier de travail où seront placés les fichiers produits par JO-OBfuscator.



A propos de JO-Obfuscator...

1

Sélectionner un dossier : Parcourir...

Dossier de sauvegarde de la macro à traiter : O:_Travail_INFO1\DOC_DEV_QCM_VBA\Dev_Obfuscator_v_Finale_Publish

Dossier de sauvegarde de la macro traitée : O:_Travail_INFO1\DOC_DEV_QCM_VBA\Dev_Obfuscator_v_Finale_Publish

2

Lancer JO-OB

3

Ouvrir une ma

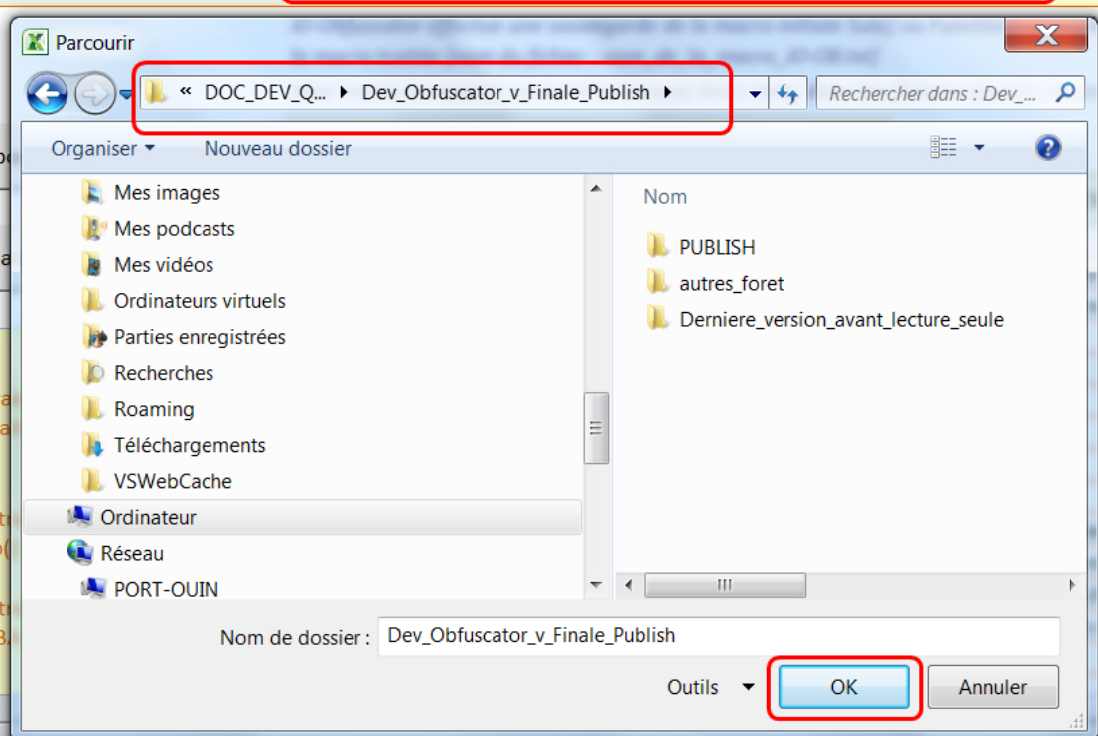
e d'emploi :

électionner un dossier de tra
s produits par JO-OBfusca

cer JO-OB :

ouvrir ou coller la macro à t
effectuer le traitement (Sub()

ouvrir le fichier de la macro tr
er-le dans votre projet VBA



Remarque :

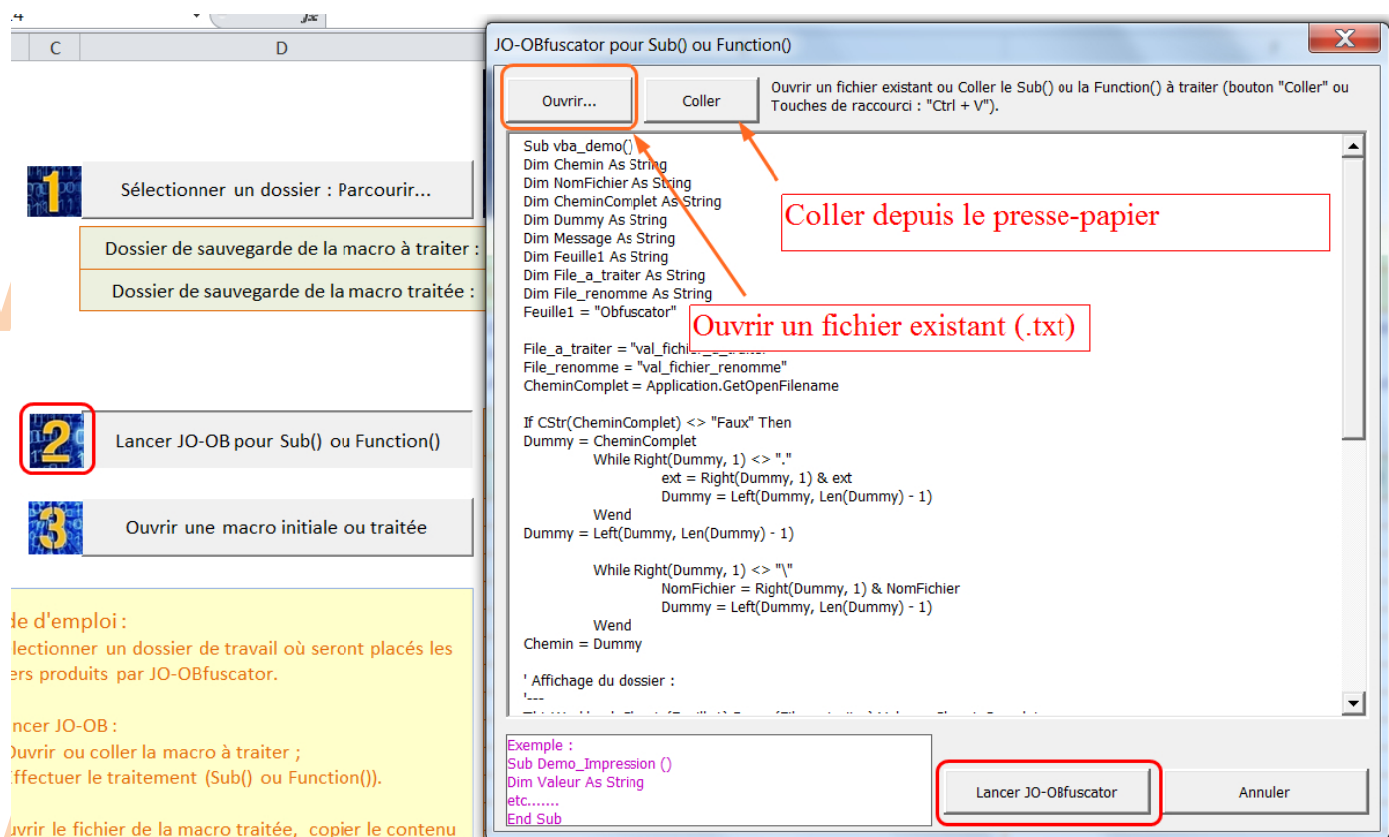
JO-OBfuscator effectue une sauvegarde de la macro initiale Sub() ou Function() (nom du fichier : nom_de_la_macro.txt) ainsi que de la macro traitée (nom du fichier : nom_de_la_macro_JO-OB.txt).

Ainsi toutes les macros sont archivées avec leurs noms dans des fichiers séparés (macros initiales et macros traitées).



Lancer JO-OBfuscator

- 2.1 Ouvrir ou coller la macro à traiter ;
- 2.2 Effectuer le traitement (Sub() ou Function()).



Remarque :

Cette fenêtre permet de coller votre macro :

- soit après l'avoir copiée depuis l'éditeur VBA ;
- soit en ouvrant un fichier au format texte (.txt).



Ouvrir le fichier relatif à une macro

Ouvrir le fichier d'une macro traitée, copier le contenu et coller-le dans votre projet VBA.

1

Sélectionner un dossier : Parcourir...

Dossier de sauvegarde de la macro à traiter :

Dossier de sauvegarde de la macro traitée :

2

Lancer JO-OB pour Sub() ou Function()

3

Ouvrir une macro initiale ou traitée

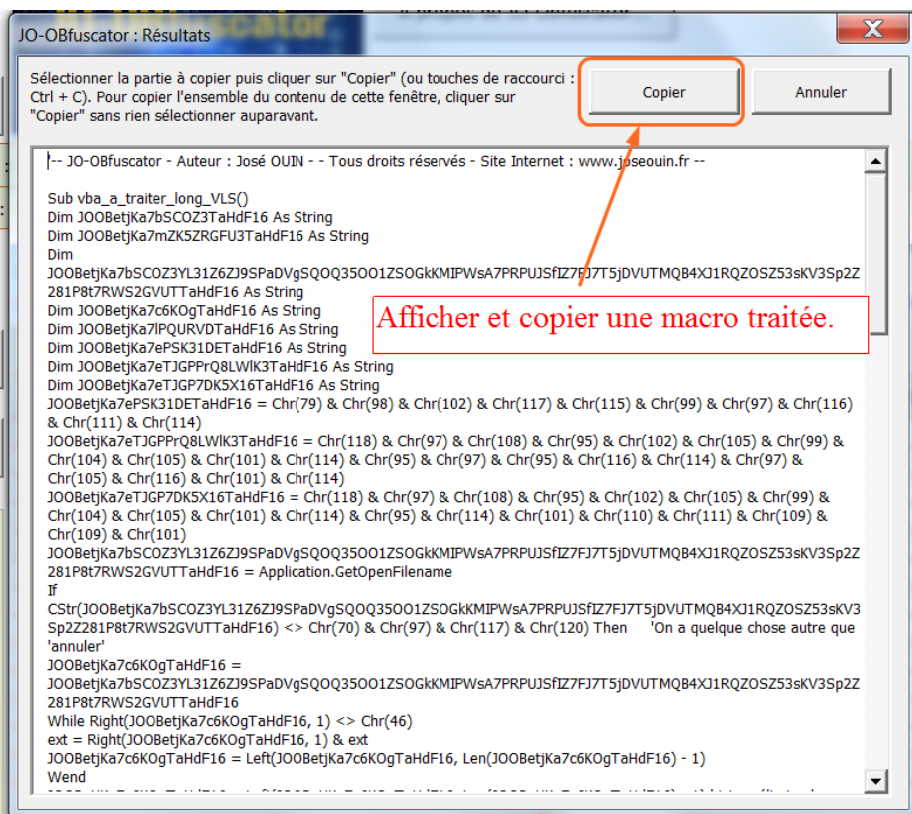
le d'emploi :

Sélectionner un dossier de travail où seront placés les fichiers produits par JO-OBfuscator.

Lancer JO-OB :

Ouvrir ou coller la macro à traiter ;
Effectuer le traitement (Sub() ou Function()).

Ouvrir le fichier de la macro traitée, copier le contenu
et coller-le dans votre projet VBA.



Remarque :

JO-OBfuscator dresse un récapitulatif des variables et des chaînes de caractères présentes. Ce récapitulatif se trouve également à la fin de chaque fichier traité. Ceci permet de garder une trace du nom initial de chacune des variables.

Rappel :

JO-OBfuscator effectue une sauvegarde de la macro initiale Sub() ou Function() (nom du fichier : nom_de_la_macro.txt) ainsi que de la macro traitée (nom du fichier : nom_de_la_macro_JO-OB.txt). Ainsi toutes les macros sont archivées avec leurs noms dans des fichiers séparés (macros initiales et macros traitées). **Aucune donnée n'est effacée par JO-OBfuscator. Aucune macro n'est effacée, elles sont toutes archivées dans des fichiers séparés.**

Toutes vos remarques et vos suggestions sont les bienvenues et m'aideront à faire évoluer cet outil. Je suis heureux de pouvoir offrir JO-OBfuscator à tous et c'est pour moi une manière de remercier toutes celles et ceux qui donnent de leur temps pour mettre en ligne des codes sources ou des conseils. Car c'est grâce à eux et grâce aux données qu'ils ont bien voulu partager que j'ai pu apprendre ce qu'aujourd'hui je sais.

José OUIN - www.joseouin.fr