INTERNET PROFESSIONNEL
Septembre 1999

Mise en œuvre: sécurité

Développez un outil de sécurité NT avec ADSI et WSH

Patrick CHAMBET
http://www.chambet.com

L'objectif de cet article:

Compléter NT en matière de sécurité à l'aide du Windows Scripting Host, tout en apprenant ADSI (Active Directory Server Interface) et VBScript.

Les outils utilisés:

- Windows NT 4.0 SP4 ou SP5
- ADSI 2.5
- Windows Scripting Host 2.0 (WSH)
- L'éditeur de code source Microsoft Visual Interdev 6.0 (ou un simple éditeur de texte comme Notepad ou l'excellent TextPad : http://www.textpad.com).

Pour en savoir plus:

- http://www.microsoft.com/windows/server/Technical/directory/adsilinks.asp Introduction à ADSI
- http://www.microsoft.com/ntserver/nts/downloads/other/ADSI25/default.asp Downloader ADSI
- http://www.15seconds.com/focus/ADSI.htm Un site bourré d'astuces ADSI, d'exemples de code, d'une FAQ et d'une liste de diffusion sur ADSI
- http://msdn.microsoft.com/scripting Tout sur le WSH 2.0 et les langages de script (Jscript, VBScript)

Windows NT comporte un oubli par rapport à d'autres systèmes d'exploitation: un utilisateur ne peut pas surveiller l'utilisation qui est faite de son compte pendant qu'il n'est pas logué.

Avec cet outil développé en VBScript, utilisant les technologies WSH et ADSI de Microsoft, vous pouvez vérifier que personne ne s'est logué avec votre compte en votre absence, mais aussi obtenir des informations sur les autres comptes NT.

Qu'est-ce qu'ADSI ?

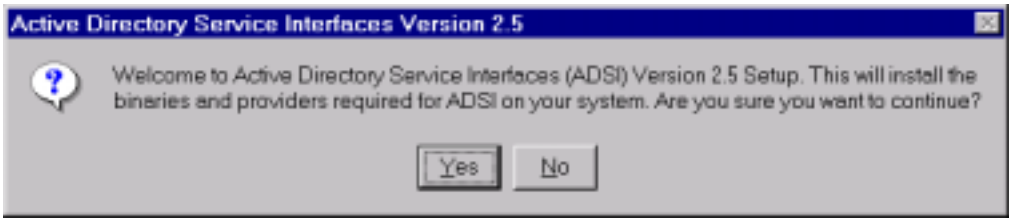
ADSI (Active Directory Server Interface) est une interface vers le modèle objet extrêmement puissant de Windows NT, qui préfigure la manière dont Windows 2000 gère les ressources. Ces dernières apparaissent sous forme arborescente, et peuvent être listées et manipulées sous forme de « collections ». On peut administrer de cette façon des serveurs, des utilisateurs, des imprimantes et des travaux d'impression, par exemple.

1 - Installez ADSI

Récupérez ADSI 2.5 à l'URL suivante (téléchargez la version qui correspond à votre environnement) :

http://www.microsoft.com/ntserver/nts/downloads/other/ADSI25/default.asp

Il suffit ensuite de lancer l'exécutable pour installer ADSI 2.5. Notez que vous n'avez pas besoin de rebooter.



Si WSH n'est pas déjà installé sur votre ordinateur, vous pouvez l'installer à partir de l'Option Pack de Windows NT (version Server ou Worksation) en cochant la checkbox correspondante, ou bien le télécharger à l'URL suivante :

http://msdn.microsoft.com/scripting/windowshost/beta

2 - Editez le fichier AccountChecker.vbs

Lancez Notepad ou TextPad, et éditez le fichier **AccountChecker.vbs** :

```
' --- NT Account Checker ---
' Check your NT account usage on the whole NT Domain
' Author: Patrick Chambet - pchambet@fr.ibm.com
' Based on Microsoft Active Directory Server Interface (ADSI) technology
' You need to install Windows Scripting Host (WSH)
' and ADSI (http://www.microsoft.com/ntserver/nts/downloads/other/ADSI25)
' or use Windows 2000

3       Option Explicit
        Dim InputAccnt, Accnt, AccntObj, Msg, Pos

4       InputAccnt = InputBox("--- NT Account Checker ---" & VbCrLf & _
        "(Needs Microsoft ADSI installed)" & VbCrLf & VbCrLf & _
        "Enter the NT Domain name and account you want to check (DOMAIN\User) :", "NT Account
Checker")
        Msg = "                    N T   A C C O U N T   C H E C K E R " & VbCrLf & _
                "_____________________________________________"

5       Pos = InStr(1, InputAccnt, "\", 1)
        If Pos > 0 Then
                Accnt = Left(InputAccnt, Pos-1) & "/" & Right(InputAccnt, Len(InputAccnt)-Pos)
                InputAccnt = UCase(Left(InputAccnt, Pos-1)) & "\" & Right(InputAccnt, Len(InputAccnt)-
Pos)
        End If

6       On Error Resume Next ' Mandatory for the ADSI command on the next line
        Set AccntObj = GetObject("WinNT://" & Accnt & ",user")

        If Err <> 0 Then
7
                Msg = Msg & VbCrLf & "No such NT account: '" & InputAccnt & "'," & VbCrLf & _
                        "or ADSI is not installed on your computer." & VbCrLf & VbCrLf
                Err.Clear

        Else
8
                Dim AccntLastLogin, AccntLastLogoff, AccntLastFailed, AccntLastAddress, BadLoginCount,
PwdExpirationDate, MaxPwdAge, PwdLastChanged

                AccntLastLogin =        "n/a"
                AccntLastLogin          =       AccntObj.LastLogin
                AccntLastLogoff =       "n/a"
                AccntLastLogoff =       AccntObj.LastLogoff
                AccntLastFailed =       "n/a"
                AccntLastFailed =       AccntObj.LastFailedLogin
                AccntLastAddress =      "n/a"
                AccntLastAddress =      AccntObj.BadLoginAddress
                BadLoginCount =         "n/a"
                BadLoginCount =         AccntObj.BadLoginCount
                PwdExpirationDate = "n/a"
                PwdExpirationDate = AccntObj.PasswordExpirationDate
                MaxPwdAge =             "n/a"
                If Int(AccntObj.MaxPasswordAge) > 0 Then
                        MaxPwdAge =     Int(AccntObj.MaxPasswordAge) / (3600 * 24)
                Else
                        MaxPwdAge = "None"
                End If
                'A trick there: compute Password Last Changed Date = Password Expiration Date minus
Password Expiration Period :-)
                'Because PwdLastChanged = AccntObj.PasswordLastChanged, though documented, isn't
supported by IADsUser object
                PwdLastChanged =        "n/a"
                PwdLastChanged          =       AccntObj.PasswordExpirationDate -
(AccntObj.MaxPasswordAge / (3600 * 24))
                Err.Clear

9               ' Format message
                Msg = Msg & VbCrLf & "Here is how the '" & InputAccnt & "' NT account has been used:"
& VbCrLf & VbCrLf & _
                "Last Login: " & Chr(9) & Chr(9) & AccntLastLogin & VbCrLf & _
                "Last Logoff: " & Chr(9) & Chr(9) & AccntLastLogoff & VbCrLf & _
```

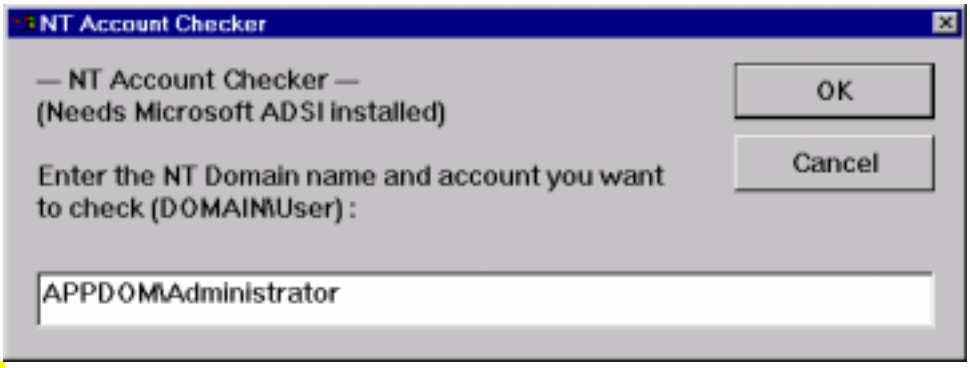
```
                "Last Failed Login: " & Chr(9) & Chr(9) & AccntLastFailed & VbCrLf & _
                "Last Bad Login Address: " & Chr(9) & AccntLastAddress & VbCrLf & _
                "Bad Login Count: " & Chr(9) & Chr(9) & BadLoginCount & VbCrLf & _
                "Password Expiration Date: " & Chr(9) & PwdExpirationDate & VbCrLf & _
                "Password Max Age (days): " & Chr(9) & CStr(MaxPwdAge) & VbCrLf & _
                "Password Last Changed: " & Chr(9) & PwdLastChanged & VbCrLf & VbCrLf

        End If

10      ' Display results
        MsgBox Msg, 0, "NT Account Checker"

11      ' Cleanup
        Set AccntObj = Nothing

'End of the Script
```

Fichier AccountChecker.vbs

3 - Commençons par déclarer nos variables. L'option « Explicit » permet de rendre la déclaration obligatoire, ce qui impose une bonne rigueur de développement.

4 – Nous allons utiliser une *InputBox* à la fois comme écran d'accueil et pour permettre la saisie du compte NT à vérifier. Les 2 paramètres d'une *InputBox* sont le titre de la fenêtre et le message qui y est affiché. Le résultat est la valeur de la saisie, de type *Variant*.

Fig. 1 : Ecran d'accueil et champ de saisie de *AccountCheck*



5 – Il faut maintenant convertir la syntaxe NT (DOMAINE\Login) en syntaxe ADSI (Domaine/Login). Pour cela, une simple manipulation de chaînes suffit.

6 – Puis on récupère l'objet *AccntObj* en utilisant un *GetObject* et une syntaxe ADSI. Notez qu'une erreur pouvant survenir à ce moment si le compte NT n'existe pas, un *On Error* va permettre de capturer l'exception et d'effectuer un traitement d'erreur :

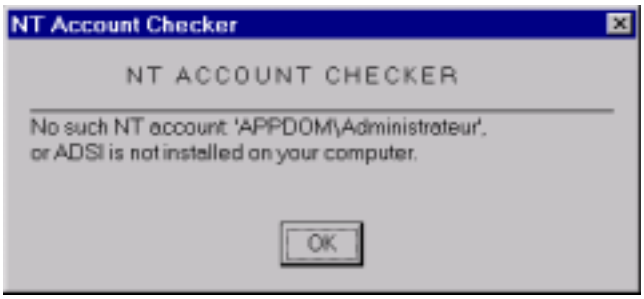


Fig. 2 : Le compte n'existe pas.

7 – Si une erreur est survenue, on affiche un message précisant que le compte NT saisi n'existe pas ou qu'ADSI n'a pas été installé.
Un *Err.Clear* réinitialise l'objet *Err* afin de permettre un traitement d'erreurs ultérieur.

8 – Si le compte NT existe, on va récupérer les informations le concernant.
Notez l'utilisation des doubles affectations de variables, de façon à optimiser le traitement d'erreur relativement pauvre offert par VBScript.
En effet, une erreur lors de la récupération d'un attribut de l'objet *AccntObj* conduira à une non affectation de la chaîne de caractères correspondante. Celle-ci gardera sa valeur initiale, fixée ici à « n/a ».
Notez aussi l'astuce consistant à calculer la date de dernier changement de mot de passe en soustrayant la période d'expiration à la date d'expiration du mot de passe. Cela est rendu nécessaire par le fait que l'attribut *PasswordLastChanged*, bien qu'il soit documenté, n'est pas valide. Il en va de même de certains autres attributs, et cela peut dépendre de votre configuration.

9 – Nous pouvons maintenant mettre le message final en forme. L'utilisation de tabulations (code ASCII 9) permet d'avoir un affichage plus clair.

10 – Une simple boîte de message permet d'afficher le résultat de l'outil. Les 3 paramètres d'une *MsgBox* sont : le message affiché, les boutons et icônes affichés et le titre de la boîte de message.

Le deuxième paramètre est construit en additionnant les constantes suivantes, en fonction des éléments que vous désirez afficher :

0 : bouton OK seulement
1 : boutons OK et Cancel
2 : boutons Abort, Retry, et Ignore
3 : boutons Yes, No, et Cancel
4 : boutons Yes et No
5 : boutons Retry et Cancel
16 : icône « Critical Message »
32 : icône « Warning Query »

48 : icône « Warning Message »
64 : icône « Information Message »

Nous avons besoin du bouton OK seulement, sans icône. Nous utilisons donc la valeur 0.

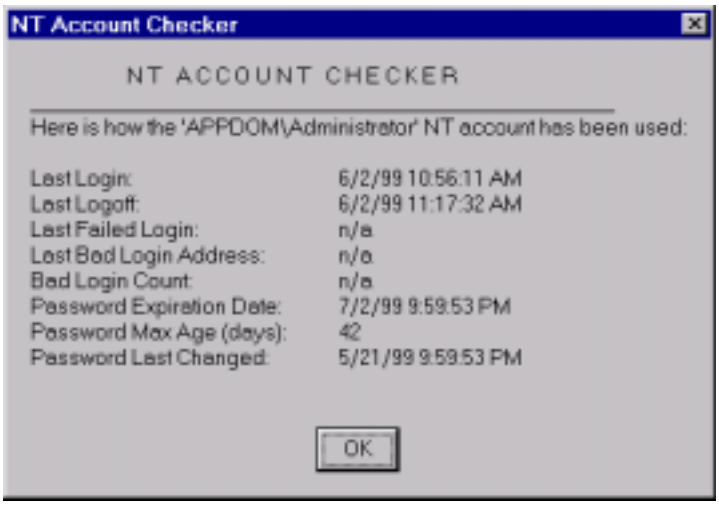


Fig. 3 : Résultat d'*AccountChecker*.

11 – Enfin, nous pouvons libérer de la mémoire en détruisant l'objet *AccntObj* avant de terminer le script. Pour cela, on lui affecte la valeur *Nothing*.

Précautions d'emploi :

Il faut parfois interpréter les informations fournies par *AccountChecker* par l'intermédiaire d'ADSI. Ainsi, si la date d'expiration d'un mot de passe se situe dans le passé, c'est le plus souvent que c'est un compte dont le mot de passe n'expire jamais.

De la même façon, si vous vous loguez avec votre compte, la date de dernier login sera bien sûr le moment où vous êtes logué vous-même. C'est donc la date de dernier login qui sera intéressante pour savoir si on a utilisé votre compte en votre absence.

Vous pouvez obtenir avec cet outil des informations sur tous les comptes de votre domaine, même si vous n'êtes pas administrateur. De plus, vous pouvez aussi obtenir des informations sur les comptes de tous les domaines avec lesquels vous avez des relations d'approbation, et ce, même si vous n'êtes pas administrateur de domaine. Il ne faut pas oublier qu'ADSI est une interface extrêmement puissante, destinée principalement aux développeurs.

Patrick Chambet