

Installation sans assistance "automatique" Windows 7 x64

Mise en place "Petit Icônes Bureau"

Pour mettre en place l'affichage de "Petites Icônes Bureau", il faut effectuer une installation standard sur un poste hôte. Configurer "Petites Icônes Bureau" normalement depuis bouton droite souris sur le bureau et sélectionner l'option "Affichage\Petites Icônes". Depuis la base de registre, exporter les clefs de registre suivantes :

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\BagMRU]

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\Bags]

[HKEY_CURRENT_USER\Control Panel\Desktop\WindowMetrics]

❖ Scripts : Petites Icônes Desktop

SmallIcon.REG

Windows Registry Editor Version 5.00

```

; =====
;
;                               Small Icon Size
;
; =====
    
```

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell]

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\BagMRU]

```

"NodeSlots"=hex:02
"MRUListEx"=hex:ff,ff,ff,ff
"NodeSlot"=dword:00000001
    
```

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\Bags]

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\Bags\1]

[HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\Bags\1\Desktop]

```

"FFlags"=dword:40200224
"Mode"=dword:00000001
"LogicalViewMode"=dword:00000003
"IconSize"=dword:00000020
"CollInfo"=hex:00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,fd,df,df,fd,10,\
00,00,00,00,00,00,00,00,00,00,00,00,04,00,00,00,18,00,00,00,30,f1,25,b7,ef,47,\
1a,10,a5,f1,02,60,8c,9e,eb,ac,0a,00,00,00,a0,00,00,00,30,f1,25,b7,ef,47,1a,\
10,a5,f1,02,60,8c,9e,eb,ac,0c,00,00,00,50,00,00,00,a6,6a,63,28,3d,95,d2,11,\
b5,d6,00,c0,4f,d9,18,d0,0b,00,00,00,78,00,00,00,30,f1,25,b7,ef,47,1a,10,a5,\
f1,02,60,8c,9e,eb,ac,0e,00,00,00,78,00,00,00
"Sort"=hex:00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,01,00,00,00,30,f1,\
25,b7,ef,47,1a,10,a5,f1,02,60,8c,9e,eb,ac,0a,00,00,00,01,00,00,00
"GroupView"=dword:00000000
"GroupByKey:FMTID"="{00000000-0000-0000-0000-000000000000}"
"GroupByKey:PID"=dword:00000000
"GroupByDirection"=dword:00000001
    
```

[HKEY_CURRENT_USER\Control Panel\Desktop\WindowMetrics]

```

"BorderWidth"=-15"
"CaptionFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,00,\
00,00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
    
```

```
"CaptionHeight"="-315"
"CaptionWidth"="-315"
"IconFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,00,00,\
00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
"IconTitleWrap"="1"
"MenuFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,00,00,\
00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
"MenuHeight"="-285"
"MenuWidth"="-285"
"MessageFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,00,\
00,00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
"ScrollHeight"="-255"
"ScrollWidth"="-255"
"SmCaptionFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,\
00,00,00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
"SmCaptionHeight"="-255"
"SmCaptionWidth"="-255"
"StatusFont"=hex:f4,ff,ff,ff,00,00,00,00,00,00,00,00,00,00,00,90,01,00,00,\
00,00,00,01,00,00,05,00,53,00,65,00,67,00,6f,00,65,00,20,00,55,00,49,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
"Shell Icon Size"="32"
"AppliedDPI"=dword:00000060
"PaddedBorderWidth"="-60"
"IconSpacing"="-1395"
"IconVerticalSpacing"="-1125"
"MinAnimate"="1"
```

Delete.REG

Windows Registry Editor Version 5.00

```
[-HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\Bags]
[-HKEY_CURRENT_USER\Software\Microsoft\Windows\Shell\BagMRU]
[-HKEY_CURRENT_USER\Control Panel\Desktop\WindowMetrics]
```

ImportXMLSchTasks.CMD

C:\VBS\SmallIconDesk\nXML.VBS

```
C:\windows\system32\SchTasks.exe /Create /TN "IconSmallSize" /XML
"C:\VBS\SmallIconDesk\IconSmallSizeDef.xml" /f
```

nXML.VBS

```
'-----DEBUT-----
' Identification de la date et heure actuel du PC
'-----
```

On Error Resume Next

```

strComputer = "."
Set objWMIService = GetObject("winmgmts:" _
& "{impersonationLevel=impersonate}!\\" & strComputer & "\root\cimv2")

Dim vDate, vHeure

Set collItems = objWMIService.ExecQuery("Select * from Win32_LocalTime")

For Each objItem in collItems

' Identification de la date actuel
vDate = objItem.Year & "-" & Right("00" & objItem.Month,2) & "-" & Right("00" & objItem.Day,2)
'Wscript.Echo "Date actuel : " & vDate

' Mise en place et formatage HH:MM:SS
' -----

' Identification de l'heure actuelle
vH = Right("00" & objItem.Hour,2)
'Wscript.Echo "Heure Actuelle : " & vH
' Réduction de 2 heure à l'heure actuelle (dû à l'ajout automatiquement, de 2 heure dans la console
Planificateur des tâches )
vHeureMoins = Right("00" & objItem.Hour - 2,2)
'Wscript.Echo "Heure Actuelle : " & vH & "          " & "Heure - 1 pour le Planificateur des tâches : " &
vHeureMoins

' Identification de la minute actuelle
vMinuteActuel = Right("00" & objItem.Minute,2)
' Ajout de 2 minutes à la minute actuelle
vMinutePlus = Right("00" & objItem.Minute + 3,2)
'Wscript.Echo "Minute Actuelle : " & vMinuteActuel & "          " & "Minute + 3 pour le Planificateur des tâches
: " & vMinutePlus

' Identification de la seconde actuelle
vS = Right("00" & objItem.Second,2)
'Wscript.Echo "Seconde Actuelle : " & vS

' Horaire d'exécution dans le Planificateur des Tâches
vHeure = vHeureMoins & ":" & vMinutePlus & ":" & vS
'Wscript.Echo "Horaire d'exécution dans le Planificateur des Tâches : " & vHeure

Next

'-----
' Identification de la date et heure actuel du PC
'-----FIN-----

'-----DEBUT-----
' Lecture_RemplacementDATEHEURE_Ecriture
'-----

on error resume next

'Nouvelle valeur à injecter
nValeur = vDate & "T" & vHeure & "Z"

'Initialisation du Pasreur XMLDOM de Microsoft
Set xmlDoc = CreateObject("Microsoft.XMLDOM")

Charger tout le fichier en mémoire
xmlDoc.Load "C:\VBS\SmallIconDesk\IconSmallSize.xml"

'On récupère tous les noeuds Triggres
Set oValeur = xmlDoc.selectSingleNode("//Triggers/TimeTrigger/StartBoundary")

'On récupère les informations sur StartBoundary
oValeur.text = nValeur
'MsgBox "StartBoundary : " & nValeur

```

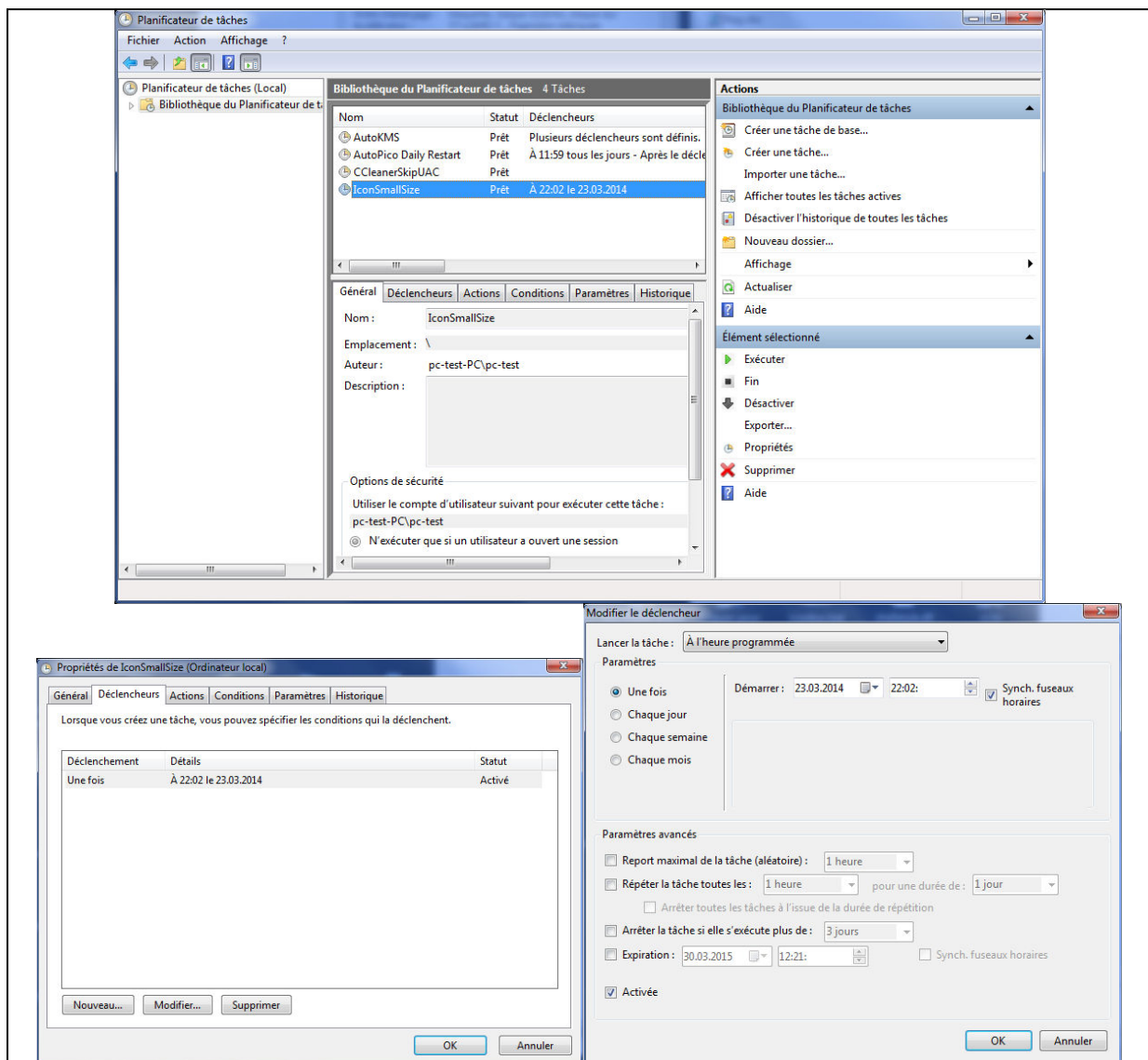
```
strResult = xmlDoc.save("C:\VBS\SmallIconDesk\IconSmallSizeDef.xml")

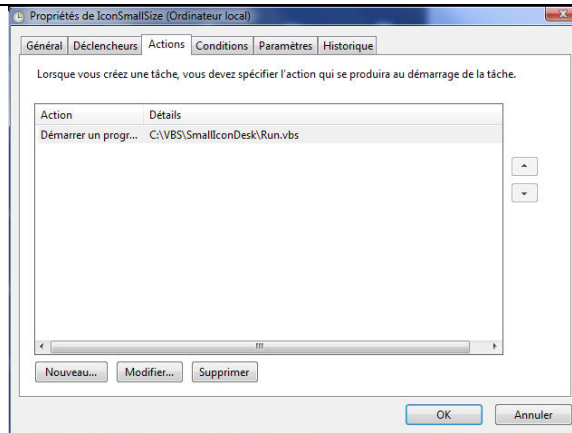
'Fermeture du Pasreur XMLDOM de Microsoft
Set xmlDoc = Nothing

'-----
' Lecture_RemplacementDATEHEURE_Ecriture
'-----FIN-----
```

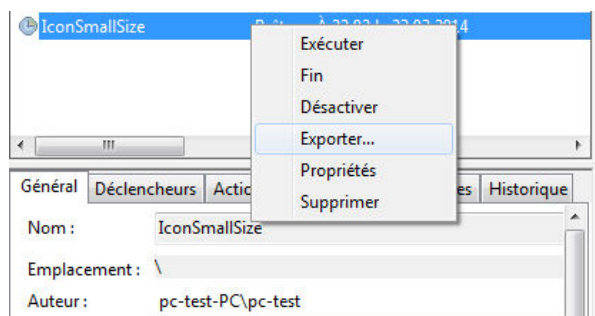
Pour mettre en place l'affichage de "Petites Icônes Bureau", il faut passer par la console "Planificateur de tâches" situé dans les "Outils systèmes". Pour cela nous avons besoin du fichier type XML pour faire notre planification automatiquement.

Ouvrir la console "Planificateur de tâches" pour créer la tâche "IconSmallSize"





Puis effectuer l'exportation du fichier XML sous le Nom : **IconSmallSize.XML**



```
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2014-03-22T22:19:06.5669</Date>
    <Author>pc-test-PC\pc-test</Author>
  </RegistrationInfo>
  <Triggers>
    <TimeTrigger>
      <StartBoundary>2014-03-22T22:25:06.5669</StartBoundary>
      <Enabled>true</Enabled>
    </TimeTrigger>
  </Triggers>
  <Principals>
    <Principal id="Author">
      <UserId>pc-test-PC\pc-test</UserId>
      <LogonType>InteractiveToken</LogonType>
      <RunLevel>LeastPrivilege</RunLevel>
    </Principal>
  </Principals>
  <Settings>
    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
    <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries>
    <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
    <AllowHardTerminate>true</AllowHardTerminate>
    <StartWhenAvailable>>false</StartWhenAvailable>
    <RunOnlyIfNetworkAvailable>>false</RunOnlyIfNetworkAvailable>
    <IdleSettings>
      <StopOnIdleEnd>true</StopOnIdleEnd>
      <RestartOnIdle>>false</RestartOnIdle>
    </IdleSettings>
    <AllowStartOnDemand>true</AllowStartOnDemand>
    <Enabled>true</Enabled>
    <Hidden>>false</Hidden>
    <RunOnlyIfIdle>>false</RunOnlyIfIdle>
    <WakeToRun>>false</WakeToRun>
    <ExecutionTimeLimit>P3D</ExecutionTimeLimit>
    <Priority>7</Priority>
  </Settings>
</Task>
```

```

</Settings>
<Actions Context="Author">
  <Exec>
    <Command>C:\VBS\SmallIconDesk\Run.vbs</Command>
  </Exec>
</Actions>
</Task>

```

Maintenant, il faut manuellement indiquer le Nom de l'ordinateur et le Nom de session du labo, sur les variables <Author> <UserId>. Pour la variable <StartBoundary>, indique la valeur DATEHEURE.

IconSmallSize.xml

```

<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2014-03-22T22:19:06.5669</Date>
    <Author>W7PROSP1X64FR\master</Author>
  </RegistrationInfo>
  <Triggers>
    <TimeTrigger>
      <StartBoundary>DATEHEURE</StartBoundary>
      <Enabled>true</Enabled>
    </TimeTrigger>
  </Triggers>
  <Principals>
    <Principal id="Author">
      <UserId>W7PROSP1X64FR\master</UserId>
      <LogonType>InteractiveToken</LogonType>
      <RunLevel>LeastPrivilege</RunLevel>
    </Principal>
  </Principals>
  <Settings>
    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
    <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries>
    <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
    <AllowHardTerminate>true</AllowHardTerminate>
    <StartWhenAvailable>>false</StartWhenAvailable>
    <RunOnlyIfNetworkAvailable>>false</RunOnlyIfNetworkAvailable>
    <IdleSettings>
      <StopOnIdleEnd>true</StopOnIdleEnd>
      <RestartOnIdle>>false</RestartOnIdle>
    </IdleSettings>
    <AllowStartOnDemand>true</AllowStartOnDemand>
    <Enabled>true</Enabled>
    <Hidden>>false</Hidden>
    <RunOnlyIfIdle>>false</RunOnlyIfIdle>
    <WakeToRun>>false</WakeToRun>
    <ExecutionTimeLimit>P3D</ExecutionTimeLimit>
    <Priority>7</Priority>
  </Settings>
  <Actions Context="Author">
    <Exec>
      <Command>C:\VBS\SmallIconDesk\Run.vbs</Command>
    </Exec>
  </Actions>
</Task>

```

Run.VBS

```
Set WshShell = CreateObject("Wscript.Shell")

Set ofs = CreateObject("Scripting.FileSystemObject")
strScriptDir = ofs.GetParentFolderName(Wscript.ScriptFullName)

' Suppression des Clefs suivantes - Bags BagMRU WindowMetrics
sREGFile1 = strScriptDir & "\\Delete.reg"
WshShell.Run "C:\Windows\regedit.exe /s " & sREGFile1 , 0, True
wscript.sleep 500

' Ecriture des clefs Icones Small - Bags BagMRU WindowMetrics
sREGFile2 = strScriptDir & "\\SmallIcon.reg"
WshShell.Run "C:\Windows\regedit.exe /s " & sREGFile2 , 0, True
wscript.sleep 500

' Désactivation puis Activation du Process Explorer.Exe
strFile1 = strScriptDir & "\\ReStartExploreWin.vbs"
WshShell.Run "C:\Windows\System32\cscript.exe " & Chr(34) & strFile1 & Chr(34)
```

ReStartExplorerWin.VBS

```
On Error Resume Next

' Kill Explorer.exe

strComputer = "."

Set objWMIService = GetObject("winmgmts:" _
    & "{impersonationLevel=impersonate}!\\" & strComputer & "\root\cimv2")

Set colProcessList = objWMIService.ExecQuery _
    ("Select * from Win32_Process Where Name = 'explorer.exe'")

For Each objProcess in colProcessList
    objProcess.Terminate(1)
Next

' Launch Explorer.exe

Set objShell = CreateObject("Wscript.Shell")

objShell.Run "explorer.exe"

Set objShell = Nothing

Wscript.exit
```



❖ Scripts : Désactiver UAC - User Account Control

ImportXMLSchTasks.CMD

C:\VBS\UAC\nXML.VBS

C:\windows\system32\SchTasks.exe /Create /TN "UACDisable" /XML "C:\VBS\UAC\UACDef.xml" /f

nXML.VBS

```
'-----DEBUT-----
' Identification de la date et heure actuel du PC
'-----

On Error Resume Next

strComputer = "."
Set objWMIService = GetObject("winmgmts:" _
& "{impersonationLevel=impersonate}!\\" & strComputer & "\root\cimv2")

Dim vDate, vHeure

Set collItems = objWMIService.ExecQuery("Select * from Win32_LocalTime")

For Each objItem in collItems

' Identification de la date actuel
vDate = objItem.Year & "-" & Right("00" & objItem.Month,2) & "-" & Right("00" & objItem.Day,2)
'Wscript.Echo "Date actuel : " & vDate

' Mise en place et formatage HH:MM:SS
'-----

' Identification de l'heure actuelle
vH = Right("00" & objItem.Hour,2)
'Wscript.Echo "Heure Actuelle : " & vH
' Réduction de 2 heures à l'heure actuelle (dû à l'ajout automatiquement, de 2 heure dans la console
Planificateur des tâches )
vHeureMoins = Right("00" & objItem.Hour - 2,2)
'Wscript.Echo "Heure Actuelle : " & vH & " " & "Heure - 1 pour le Planificateur des tâches : " &
vHeureMoins

' Identification de la minute actuelle
vMinuteActuel = Right("00" & objItem.Minute,2)
```



```
' Ajout de 2 minutes à la minute actuelle
vMinutePlus = Right("00" & objItem.Minute + 1,2)
'Wscript.Echo "Minute Actuelle : " & vMinuteActuel & " " & "Minute + 1 pour le Planificateur des tâches
: " & vMinutePlus

' Identification de la seconde actuelle
vS = Right("00" & objItem.Hour,2)
'Wscript.Echo "Seconde Actuelle : " & vS

' Horaire d'exécution dans le Planificateur des Tâches
vHeure = vHeureMoins & ":" & vMinutePlus & ":" & vS
'Wscript.Echo "Horaire d'exécution dans le Planificateur des Tâches : " & vHeure

Next

'-----
' Identification de la date et heure actuel du PC
'-----FIN-----

'-----DEBUT-----
' Lecture_ReplacementDATEHEURE_Ecriture
'-----

on error resume next

'Nouvelle valeur à injecter
nValeur = vDate & "T" & vHeure & "Z"

'Initialisation du Pasreur XMLDOM de Microsoft
Set xmlDoc = CreateObject("Microsoft.XMLDOM")

Charger tout le fichier en mémoire
xmlDoc.Load "C:\VBS\UAC\UAC.xml"

'On récupère tous les noeuds Triggres
Set oValeur = xmlDoc.selectSingleNode("//Triggers/TimeTrigger/StartBoundary")

'On récupère les informations sur StartBoundary
oValeur.text = nValeur
'MsgBox "StartBoundary : " & nValeur

strResult = xmlDoc.save("C:\VBS\UAC\UACDef.xml")

'Fermeture du Pasreur XMLDOM de Microsoft
Set xmlDoc = Nothing

'-----
' Lecture_ReplacementDATEHEURE_Ecriture
'-----FIN-----
```

Pour mettre en place la désactivation de "UAC", il faut passer par la console "Planificateur de tâches" situé dans les "Outils systèmes". Pour cela nous avons besoin du fichier type XML pour faire notre planification automatiquement.

➔ Voir la section [Scripts : Petites Icônes Desktop]

UAC.XML

```
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2014-03-22T22:19:06.5669</Date>
    <Author>W7PROSP1X64FR\master</Author>
  </RegistrationInfo>
  <Triggers>
    <TimeTrigger>
      <StartBoundary>DATEHEURE</StartBoundary>
      <Enabled>true</Enabled>
```

```

</TimeTrigger>
</Triggers>
<Principals>
  <Principal id="Author">
    <UserId>W7PROSP1X64FR\master</UserId>
    <LogonType>InteractiveToken</LogonType>
    <RunLevel>LeastPrivilege</RunLevel>
  </Principal>
</Principals>
<Settings>
  <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
  <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries>
  <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
  <AllowHardTerminate>true</AllowHardTerminate>
  <StartWhenAvailable>false</StartWhenAvailable>
  <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
  <IdleSettings>
    <StopOnIdleEnd>true</StopOnIdleEnd>
    <RestartOnIdle>false</RestartOnIdle>
  </IdleSettings>
  <AllowStartOnDemand>true</AllowStartOnDemand>
  <Enabled>true</Enabled>
  <Hidden>false</Hidden>
  <RunOnlyIfIdle>false</RunOnlyIfIdle>
  <WakeToRun>false</WakeToRun>
  <ExecutionTimeLimit>P3D</ExecutionTimeLimit>
  <Priority>7</Priority>
</Settings>
<Actions Context="Author">
  <Exec>
    <Command>C:\VBS\UAC\Run.vbs</Command>
  </Exec>
</Actions>
</Task>

```

Run.VBS

```

Set WshShell = CreateObject("Wscript.Shell")

Set ofs = CreateObject("Scripting.FileSystemObject")
strScriptDir = ofs.GetParentFolderName(Wscript.ScriptFullName)

' Désactivation UAC via REG
strFile1 = strScriptDir & "\Reg.vbs"
WshShell.Run "C:\Windows\System32\cmd.exe " & Chr(34) & strFile1 & Chr(34)

' Pause de 1 secondes
wscript.sleep 1000

strFile4 = strScriptDir & "\Reboot.vbs"
WshShell.Run "C:\Windows\System32\cmd.exe " & Chr(34) & strFile4 & Chr(34)

```

REG.VBS

```

Set WshShell = CreateObject("Wscript.Shell")

sREGFile = "C:\VBS\UAC\UAC.reg"
WshShell.Run "C:\Windows\regedit.exe /s " & sREGFile , 0, True

' Pause 1 seconde
wscript.sleep 1000

```

UAC.REG

```
Windows Registry Editor Version 5.00

;           Windows 7 Professionnel Service Pack 1

;=====

;           Sécurité

;=====
;Contrôle de Compte d'Utilisateur UAP - UAC [LUA] User Account Protection
;Comptes Utilisateurs\Modifier les paramètres de contrôle de compte utilisateur
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System]
; 0 Enlever les privilèges sans invite utilisateur
; 1 Demande d'informations d'identification sur le bureau sécurisé
; 2 Demande de consentement sur le bureau sécurisé
; 3 Demande d'informations d'identification
; 4 Demande de Consentement
; 5 Demande de Consentement pour les binaires non windows

;Exécuter les comptes d'administrateurs en mode d'approbation d'administrateur
"EnableLUA"=dword:00000000

;Comportement de l'invite d'élévation pour les administrateurs en mode d'approbation Administrateur
"ConsentPromptBehaviorAdmin"=dword:00000000

;Détecter les installations d'applications et demander l'élévation
"EnableInstallerDetection"=dword:00000000

;Autoriser les applications UIAccess à demander l'élévation sans utiliser le bureau sécurisé
"EnableUIADesktopToggle"=dword:00000000

;Passer au Bureau sécurisé lors d'une demande d'élévation
"PromptOnSecureDesktop"=dword:00000000

;Elever uniquement les exécutables signés et validés
"ValidateAdminCodeSignatures"=dword:00000000

;Utiliser le mode Approbation administrateur pour le compte Administrateur intégré
"FilterAdministratorToken"=dword:00000000

;Virtualiser les échecs d'écritures de fichiers et de Registre dans des emplacements définis par utilisateur
"EnableVirtualization"=dword:00000001

;Elever uniquement les applications UIAccess installées à des emplacements sécurisés
"EnableSecureUIAPaths"=dword:00000001
```

Reboot.VBS

```
'Fusion des fichiers registre
Set WshShell = CreateObject("Wscript.Shell")

Set ofs = CreateObject("Scripting.FileSystemObject")
strScriptDir = ofs.GetParentFolderName(Wscript.ScriptFullName)
strFile = strScriptDir & "\Ul.hta"
WshShell.Run "C:\Windows\System32\mshta.exe " & Chr(34) & strFile & Chr(34)
```

UI.HTA

```

<html>
<head>
<hta:application ShowInTaskbar="yes" sysMenu="no" border="none" contextMenu="yes" Id="install"/>

<script type='text/vbscript'>
Set WshShell = CreateObject("Wscript.Shell")

'-----
'Main
'-----
Sub Window_onload()
'-----
'Positionnement de la fenetre
'-----
Me.ResizeTo 700, 150
Me.MoveTo Screen.Width / 2 - 350, Screen.Height / 2 - 75

MajAffichage
End Sub

'-----
'Affichage du message de redemarrage
'-----
Sub MajAffichage

document.getElementById("CadreHeader").innerHTML = "<span style='font: 14 pt arial; font-weight: bold; color: Orange; vertical-align: text-bottom;'>_
& "Redemarrage de l'ordinateur dans 2 secondes..." & "</br>" & "...: UAC – Contrôle de Comptes
Utilisateur désactivé :... " & "</br>by " & chr(174) & "TechWan2003</span>"
Me.ResizeTo 700, 150
Me.MoveTo Screen.Width / 2 - 350, Screen.Height / 2 - 75
setTimeout "Redemarrage", 10000

End Sub

'-----
'Redemarrage
'-----
Sub Redemarrage
'On ferme la fenetre
Me.Close()

'Reboot la machine
Set objWMIService =
GetObject("winmgmts: {impersonationLevel=impersonate,(Shutdown)}!\\.\root\cimv2")
Set colOperatingSystems = objWMIService.ExecQuery("Select * from Win32_OperatingSystem")
For Each objOperatingSystem in colOperatingSystems
objOperatingSystem.Reboot()
Next
End Sub

</script>
</head>

<body scroll="no" bgcolor="#000000">

<div id="Cadre" align="center" style="font: 8pt arial; color: white; font-weight: bold; padding-top: 10px;">
<div id="CadreHeader" align="Center" style="border: 0px solid red; float: left; position: relative; left: 0px; top: 0px; width: 100%; font: 10pt arial; color: orange; font-weight: bold;"></div>
</div>

</body>
</html>

```

❖ Scripts : AutoUnattend.XML

AutoUnattend.XML

....

```
<settings pass="oobeSystem">
  <component name="Microsoft-Windows-Shell-Setup" processorArchitecture="amd64"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

....

```
  <FirstLogonCommands>
    <SynchronousCommand wcm:action="add">
      <Description>Désactivation Totale UAC</Description>
      <CommandLine>C:\VBS\UAC\ImportXMLSchTasks.CMD</CommandLine>
      <Order>1</Order>
    </SynchronousCommand>
    <SynchronousCommand wcm:action="add">
      <Description>Icon Small Size Desktop</Description>
      <CommandLine>C:\VBS\SmallIconDesk\ImportXMLSchTasks.CMD</CommandLine>
      <Order>2</Order>
    </SynchronousCommand>
  </FirstLogonCommands>
</component>
</settings>
<settings pass="offlineServicing">
  <component name="Microsoft-Windows-LUA-Settings" processorArchitecture="amd64"
publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS"
xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <EnableLUA>>false</EnableLUA>
  </component>
</settings>
</unattend>
```

La désactivation UAC via AutoUnattend.XML passe par le paramétrage suivant :

```
<EnableLUA>>false</EnableLUA>
```