Some things to know about the Sysprep process on Windows 7

In the following article, I will show you the basics of Sysprep creation in Windows 7. There have been a lot of tutorials on how to do this on Microsoft Windows XP, but to apply it. Use with Windows 7 is not a simple thing.

In the following article, I will show you the basics of Sysprep creation in Windows 7. There have been a lot of tutorials on how to do this on Microsoft Windows XP, but to apply it. Use with Windows 7 is not a simple thing.

The necessary tools in this test:

1. The WAIK (Windows Automated Installation Kit) is installed on a computer not used to create an image file.If you want to create the**unattend.xml** file during the entire sysprep process, we will need to need the **image.wim**fileon the installation disc of Windows 7.
2. Mysysprep2 Tool

Basically, our implementation process will be based on the following order: after installing Windows 7 on the computer, log in using the domain account that has been assigned Administrator rights (and has been assigned to the Administrators group). This account is not allowed to use any scripts or policies, in this test, we use the domain account (named imageprep), and the software to be installed will be stored in the DFS folder split. share. When executing the system in this way, all applications that want to repair will easily find the original installation file. After that, we will install all the necessary programs, update Windows, create local accounts, set up profile mode . at will. This entire profile becomes the default profile of the computer later in the image file.

When everything is well prepared, we will have to create an **unattend.xml** answer file using the **Windows System Image Manager** tool **(Windows SIM)** on another computer with WAIK installed. After successfully installing, please run Windows SIM from the Start Menu and follow these steps:

1. Insert the Windows 7 installation disc into the DVD drive
2. Start Windows SIM, from the menu you select**Tools> Create Catalog** , the system will require the **install.wim**fileon the original installation disk, just point to the sources folder.When the catalog creation process is complete, the system will be ready to set up and configure the answer file.

One of our requirements is that sysprep will ask the user to initialize the computer name before assigning it to the generic domain in the system. However, this process raises another problem: when using an **unattend.xml** and **Windows sysprep** answer file, if the user does not specifically declare the computer name in the file, the system will automatically assign that computer to the domain by a random string of names. Specifically, if you delete the part from the answer file, the user will be asked by the system to enter this parameter when the answer file is approved during the sysprep process, but if so, it is too late, at that time. , the computer has been assigned

to the domain by a random name, and therefore may cause problems later on. But with **Mysysprep2** , users can give the computer a name and assign it to the domain after processing the **unattend.xml** file. Besides, **Mysysprep** should be placed in the **C: WindowsSystem32Sysprep** folder, next to **sysprep.exe** (not deleted or renamed **sysprep.exe**).

Below is the unattend.xml file used in this test:

first
2
3
4
5 true
6
7
8 0
9
ten
11
twelfth
13 no
14 Your Company
15 Your Domain
16 true
17 true
18 false
19 true
20 false
21 true
22 false
23 Low
24 false
25 about: blank
26 true
27 true
28 0
29 true
30 true
False 31
32 false
33 Medium
34
35
36 0d81a790-d84a-407f-a5e0-92464dcf560c
37
38
39 false
40
41

42
43 32
44 96
45 1366
46 60
47 768
48
49 false
50 true
51 false
52 false
53 Organization Name
54 Registered Owner
55 false
56 Eastern Standard Time
57% Please input a computer name (0.15)%
58
59
60 1
sixty one
62
63
sixty four
65 Your Domain
66 Some Password
67 Some User
68
69 OU = Re-Imaged, OU = YourComputers, DC = YourCompany, DC = Com
70% Please select a domain to join {Your Domain}%
71
72
seventy three
74
75
76 EN-US
77 EN-US
78 EN-US
79 EN-US
80
81
82
83 true
84 Other
85 3
86 true
eighty seven
88 false
89 Eastern Standard Time

90 false
91 false
92 false
ninety three
ninety four
95 The Password
96 false
97
98
99
100
101 The Password
102 false
103
104 Temp Local Account
105 TempUser
106 Users
107 TempUser
108
109
110
111
112 ClearType
113
114
115
116
117

First, we will learn about the two statements below:

% Please input m?t tên ng??i dùng (0.15)%
% Please select a domain to Join {your.domain.com}%

There is another specific component when using mysysprep2, which is that the system will display different messages in the SIM section but will completely ignore this information, the part **(0,15)** in will automatically delete extra characters if the computer name is longer than 15 characters. Besides, we also see:

true

The above command will ask sysprep to copy the entire profile information that the user created earlier to the default profile. In addition, the user must specify the OU to assign the computer, otherwise the component will not be assigned to the system's domain:

OU = Test, DC = Your, DC = Domain, DC = Here

Another point to note with the **unattend.xml** file is that we have to create a local user account during sysprep processing. Specifically, when a user installs Windows 7 using a disk, it is necessary to create a local user to log in, and this process continues when using the **attend.xml** and **sysprep** files. In this test, we created a local user

account named **TempUser** - which will be deleted when we first log in to Windows. The purpose of this account is to create just the process of creating local users when sysprep works.

In addition, we can also use another way to implement this process, although not specifically presented but quite effective. Specifically, on the computer we perform the image file creation process, open the folder **C: WindowsSteup** , create a new folder named Scripts here. Inside that, we create a **SetupComplete.cmd** file - which starts automatically before the Windows login screen starts. And this is also the step that TempUser account is deleted.

To delete this **TempUser** account, please add the following command to **SetupComplete.cmd:**

net user TempUser / delete

And activate the Administrator account with the command:

net Administrator / active user: yes

Although you may have enabled the Administrator account on the computer used to create the image before using sysprep, the account is still disabled. To delete the unattend.xml file, add the following line of code to the SetupComplete.cmd file:

del C: WindowsSystem32Sysprepunattend.xml

On the other hand, the user's password after being declared in the xml file is encrypted, but in essence, the file does not exist in this process. On the other hand, users can adjust their answer file according to their needs. After completing the **unattend.xml** file, copy to the **C: WindowsSystem32Sysprep** folder **,** and we are ready to run sysprep on the computer. Inside the newly created profile section, use the Command Prompt, point to the **C: WindowsSystem32Sysprep** folder and type the command below:

mysysprep.exe generalize oobe shutdown unattend: unattend.xml

The computer will turn itself off after sysprep is complete. And this is also the time to create the image of the hard drive, each has its own way to complete this. At this test, we used the Norton Ghost utility, and as a result, there were 2 partitions, C of the operating system, and D, which contained the user's data. In the next process, applying the image file back to the computer, we will be prompted to initialize the computer and domain name. The Domain Name section will automatically display after completing:

% Please select a domain to Join {your.domain.com}%

In case the system has multiple domains, we will use the dropdown menu structure at this step:

% Please select m?t mi?n ?? Join {domain1; domain2; domain3}%

The computer will complete the installation process and can continue with the familiar boot Welcome screen. When all steps are successful, we can log in with the domain account.

Some common problems and solutions:

After completing the step of deleting the **unattend.xml** file, we need to ensure that the entire hardware of the system remains stable when using sysprep:

true

When we set the True value at this step, Plug and Play devices will be fully installed with drivers on the computer that applies the image file. And these hardware devices do not need to be reinstalled during the creation and configuration process, these information will be stored in: **x86_Microsoft-Windows-PnpSysprep.**

Create parameters using the energy level in the image file and initialize this parameter in the unattend.xml file, specifically in the **Control Panel> Power Options section** . When we want to perform this process, we will need the GUID parameter, and in order to find this parameter we must use the following command in Command Prompt:

PowerCfg -List

and find where the newly created plan is located. The last thing is to assign this GUID parameter to the unattend.xml file to become the default power usage and consumption level in the system, which will be stored in the **x86_Microsoft-Windows-powercpl__neutral section.**

Split the drive into **DriverStore** in the image file, we can use the pnputil.exe utility. Specifically, in case the system already has a driver of a certain peripheral device such as a printer or scan, we don't want other user accounts to receive notifications every time we install the driver, we will assign the This parameter enters **DriverStore** in the Windows image file. For example: **pnputil.exe -a HP0001.inf** . To find out more about this syntax, please refer to the tutorial directly from Microsoft here.

To turn off the notification of **Action Center: Set backup** , we will create a new file **DisableBackupMonitoring.reg** with **Notepad** and use the code below:

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINESOFTWAREMicrosoftWindowsCurrentVersionWindowsBackup]
"DisableMonitoring" = dword: 00000001

Below is a code used to remove the default shortcuts on the taskbar when the user logs in to the system for the first time. You can refer to this code here, technically this script file will be copied to the **C: WindowsSystem32SysprepCustom** folder via a batch file that is replaced in the startup section of the user profile. After that, the batch file will be automatically deleted and the process will only happen on the first login of the user:

```
1 Option Explicit
2
3 Const CSIDL_STARTMENU = & HB
4 Const CSIDL_COMMON_PROGRAMS = & H17
5
6 Dim objShell, objFSO
7 Dim objCurrentUserStartFolder
8 Dim strCurrentUserStartFolderPath
9 Dim objAllUsersProgramsFolder
```

```
Dim strAllUsersProgramsPath
Dim objFolder
Dim objFolderItem
Dim colVerbs
Dim objVerb

Set objShell = CreateObject ("Shell.Application")
Set objFSO = CreateObject ("Scripting.FileSystemObject")

Set objCurrentUserStartFolder = objShell.NameSpace (CSIDL_STARTMENU)
strCurrentUserStartFolderPath = objCurrentUserStartFolder.Self.Path

Set objAllUsersProgramsFolder = objShell.NameSpace (CSIDL_COMMON_PROGRAMS)
strAllUsersProgramsPath = objAllUsersProgramsFolder.Self.Path

" " " " " " " " " " " " " " " " " " " " Unpin Shortcuts " " " " " " " " " " " " " " " " " " " " " " " " " " " "

'*** All Users Shortcuts ****

'Internet Explorer
If objFSO.FileExists (strAllUsersProgramsPath & "Network & InternetMicrosoftInternet Explorer.lnk") Then
Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Network & InternetMicrosoft")
Set objFolderItem = objFolder.ParseName ("Internet Explorer.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
Next
End If

'Mozilla Firefox
If objFSO.FileExists (strAllUsersProgramsPath & "Network & InternetMozilla FirefoxMozilla Firefox.lnk") Then
Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Network & InternetMozilla Firefox")
Set objFolderItem = objFolder.ParseName ("Mozilla Firefox.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
Replace If (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
Next
End If

'Windows Explorer
If objFSO.FileExists (strAllUsersProgramsPath & "AccessoriesWindows Explorer.lnk") Then
Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Accessories")
Set objFolderItem = objFolder.ParseName ("Windows Explorer.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
```

```
56 Next
57 End If
58
59 'Windows Media Player
60 If objFSO.FileExists (strAllUsersProgramsPath & "Network & InternetMicrosoftWindows Media Player.lnk") Then
61 Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Network & InternetMicrosoft")
62 Set objFolderItem = objFolder.ParseName ("Windows Media Player.lnk")
63 Set colVerbs = objFolderItem.Verbs
64 For Each objVerb In colVerbs
65 If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
66 Next
67 End If
68
69 '*** Current Users Shortcuts ****
70
71 'Internet Explorer
72 If objFSO.FileExists (strCurrentUserStartFolderPath & "ProgramsInternet Explorer.lnk") Then
73 Set objFolder = objShell.Namespace (strCurrentUserStartFolderPath & "Programs")
74 Set objFolderItem = objFolder.ParseName ("Internet Explorer.lnk")
75 Set colVerbs = objFolderItem.Verbs
76 For Each objVerb In colVerbs
77 If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
78 Next
79 End If
80
81 'Mozilla Firefox
82 If objFSO.FileExists (strCurrentUserStartFolderPath & "ProgramsMozilla FirefoxMozilla Firefox.lnk") Then
83 Set objFolder = objShell.Namespace (strCurrentUserStartFolderPath & "ProgramsMozilla Firefox")
84 Set objFolderItem = objFolder.ParseName ("Mozilla Firefox.lnk")
85 Set colVerbs = objFolderItem.Verbs
86 For Each objVerb In colVerbs
87 If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
88 Next
89 End If
90
91 'Windows Explorer
92 If objFSO.FileExists (strCurrentUserStartFolderPath & "ProgramsAccessoriesWindows Explorer.lnk") Then
93 Set objFolder = objShell.Namespace (strCurrentUserStartFolderPath & "ProgramsAccessories")
94 Set objFolderItem = objFolder.ParseName ("Windows Explorer.lnk")
95 Set colVerbs = objFolderItem.Verbs
96 For Each objVerb In colVerbs
97 If Replace (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
98 Next
99 End If
100
101 'Windows Media Player
```

```
If objFSO.FileExists (strCurrentUserStartFolderPath & "ProgramsWindows Media Player.lnk") Then
Set objFolder = objShell.Namespace (strCurrentUserStartFolderPath & "Programs")
Set objFolderItem = objFolder.ParseName ("Windows Media Player.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
Replace If (objVerb.name, "&", "") = "Unpin from Taskbar" Then objVerb.DoIt
Next
End If

" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " '

" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " 'Pin Shortcuts' " " " " " " " " " " " " " " " " " " " " " " " " " " " " " '

'Internet Explorer - All Users
If objFSO.FileExists (strAllUsersProgramsPath & "Network & InternetMicrosoftInternet Explorer.lnk") Then
Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Network & InternetMicrosoft")
Set objFolderItem = objFolder.ParseName ("Internet Explorer.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
If Replace (objVerb.name, "&", "") = "Pin to Taskbar" Then objVerb.DoIt
Next
End If

'Mozilla Firefox - All Users
If objFSO.FileExists (strAllUsersProgramsPath & "Network & InternetMozilla FirefoxMozilla Firefox.lnk") Then
Set objFolder = objShell.Namespace (strAllUsersProgramsPath & "Network & InternetMozilla Firefox")
Set objFolderItem = objFolder.ParseName ("Mozilla Firefox.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
Replace (objVerb.name, "&", "") = "Pin to Taskbar" Then objVerb.DoIt
Next
End If

'Windows Explorer - Current User
If objFSO.FileExists (strCurrentUserStartFolderPath & "ProgramsAccessoriesWindows Explorer.lnk") Then
Set objFolder = objShell.Namespace (strCurrentUserStartFolderPath & "ProgramsAccessories")
Set objFolderItem = objFolder.ParseName ("Windows Explorer.lnk")
Set colVerbs = objFolderItem.Verbs
For Each objVerb In colVerbs
If Replace (objVerb.name, "&", "") = "Pin to Taskbar" Then objVerb.DoIt
Next
End If

" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " '
```

After setting up the **Network Location** section in the **unattend.xml** file, but the user still receives the warning information on the first login, install the following hotfix on the computer used to make the image: http:/// support .microsoft.com/kb/2028749

IE's homepage address is reset after using sysprep: fix by assigning IE's own settings to the unattend.xml file in **x86_Microsoft-Windows-IE-InternetExplorer__neutral_31bf3856ad364e35_nonSxS.**

The user loses the Aero theme after using sysprep: use the following command in the Command Prompt:

WinSAT prepop

The above process will create the WinSAT **prepop .xml** file and store it in the **% WINDIR% performancewinsatdatastore directory** and allow the user to keep the theme settings after using sysprep.

Good luck!