

技術ノート KGTN 2010092203

現象

プリンタドライバを多数の GG サーバにインストールしたい。

説明

このようなケースでは、「プリンタドライバを含む」マスタとなる GG サーバのイメージを作成し、それを展開するのが最善の方法です。プリンタドライバが追加された場合は、各 GG サーバ上で標準的な手順でインストールするか、または「新しいプリンタドライバを含む」マスタとなる GG サーバのイメージを作成し、それを展開します。

「自己責任でも構わないので、簡単にプリンタドライバを一括コピーしたい」ということであれば、下記のスクリプトを利用する方法があります。マスタとなるサーバにプリンタドライバをインストールし、

```
cscript pdr_server.vbs
```

を実行し、その後各 GG サーバで

```
cscript pdr_clinet.vbs -server <マスタとなるサーバの IP アドレス>
```

を実行することで、マスタとなるサーバのプリンタドライバが一括コピーされます。なお、このスクリプトはあくまでもサンプルであり、その内容や動作について如何なる保証もサポートも行いません。

FILE: pdr_server.vbs

```
option explicit
```

```
'_
```

```
'_ 定数
```

```
'_
```

```
const L_DriverRegistry = "HKLM\SYSTEM\CurrentControlSet\Control\Print\Environments\Windows NT x86"
```

```
const L_DriverFolder = "C:\WINDOWS\system32\spool\drivers\w32x86"
```

```
'_
```

```
'_ Usage: cscript pdr_server.vbs
```

```
'_
```

```
main
```

```
'-----  
'-----                メインルーチン                -----  
'-----
```

```
sub main
```

```
    dim iRetVal
```

```
    iRetVal = gg_ExportRegistry(L_DriverRegistry, L_DriverFolder & "%printer_registry.reg")
```

```
    wscript.echo "iRetVal = " & iRetVal
```

```
end sub
```

```
'-----  
'----- ローカルコンピュータの指定レジストリのエクスポート -----  
'-----
```

```
function gg_ExportRegistry(strKey, strFile)
```

```
    on error resume next
```

```
    dim iResult
```

```
    dim strCommand
```

```
    dim objShell
```

```
    '-----
```

```
    iResult = -999
```

```
    strCommand = "reg export "" & strKey & "" "" & strFile & "" /y"
```

```
    set objShell = WScript.CreateObject("WScript.Shell")
```

```
    objShell.Run strCommand, 7, true
```

```
    '-----
```

```
    iResult = 0
```

```
    gg_ExportRegistry = iResult
```

```
end function
```

FILE: pdr_client.vbs

```
option explicit
```

```
'_
```

```
' - 定数
```

```
'_
```

```
const kErrorSuccess = 0
```

```
const kErrorFailure = 1
```

```
'_---
```

```
const kStartSpooler = "spooler/start"
```

```
const kStopSpooler = "spooler/stop"
```

```
const kCopyFolder = "folder/copy"
```

```
const kExportRegistry = "registry/export"
```

```
const kImportRegistry = "registry/import"
```

```
const kExportPrinterRegistry = "registry/export/printer"
```

```
const kExecuteBatch = "batch/execute"
```

```
'_---
```

```
const L_Empty_Text = ""
```

```
const L_Space_Text = " "
```

```
const L_Error_Text = "エラー"
```

```
const L_Success_Text = "成功"
```

```
const L_Failed_Text = "失敗"
```

```
const L_Hex_Text = "0x"
```

```
'_---
```

```
const L_DriverRegistry = "HKLM\SYSTEM\CurrentControlSet\Control\Print\Environments\Windows NT x86"
```

```
'_---
```

```
const ADS_SERVICE_STOPPED = 1
```

```
const ADS_SERVICE_START_PENDING = 2
```

```
const ADS_SERVICE_STOP_PENDING = 3
```

```
const ADS_SERVICE_RUNNING = 4
```

```
const ADS_SERVICE_CONTINUE_PENDING = 5
```

```
const ADS_SERVICE_PAUSE_PENDING = 6
```

```
const ADS_SERVICE_PAUSED = 7
```

```
const ADS_SERVICE_ERROR = 8
```

```
'_
```

```
' - Usage: cscript pdr_client.vbs -server <pdr_server>
```

```
'_
```

```
main
```

```
'_-----
```

```
'_-----
```

メインルーチン

```
'_-----
```

```
sub main
```

```
    dim bRetVal
```

```
    dim iRetVal
```

```
    dim strAction
```

```
    dim strServer
```

```
    dim strSrc
```

```
    dim strDst
```

```
    dim strCommand
```

```
'_-----
```

```

iRetVal = ParseCommandLine(strServer)
MsgBox "0) Parse Command : iRetVal = " & iRetVal & " , strServer = " & strServer
' ---
iRetVal = gg_ManageSpooler(".", kStopSpooler)
MsgBox "1) Stop Spooler : iRetVal = " & iRetVal
if iRetVal <> 0 and iRetVal <> 1 then
    wscript.echo "Print Spooler を停止できません"
end if
' ---
strSrc = "%%" & strServer & "%print$%w32x86"
strDst = "C:\WINDOWS\system32\spool\drivers\%w32x86"
iRetVal = gg_CopyFolder(strSrc, strDst)
MsgBox "3) Copy Files : iRetVal = " & iRetVal
' ---
iRetVal = gg_ImportRegistry(strDst & "%printer_registry.reg")
MsgBox "4) Import Registry : iRetVal = " & iRetVal
' ---
iRetVal = gg_ManageSpooler(".", kStartSpooler)
MsgBox "5) Start Spooler : iRetVal = " & iRetVal
if iRetVal <> 0 and iRetVal <> 1 then
    wscript.echo "Print Spooler を開始できません"
end if
end sub

```

```

' -----
' ---                               ---
' -----

```

```

function gg_ManageSpooler(strComputer, strOperation)
    dim iResult
    dim objService
    ' ---
    iResult = -999
    strComputer = RemoveBackslashes(strComputer)
    set objService = GetObject("WinNT:///" & strComputer & "/Spooler,Service")
    ' ---
    select case (strOperation)
        case kStartSpooler
            select case (objService.Status)
                case ADS_SERVICE_STOPPED
                    objService.Start
                    while objService.Status <> ADS_SERVICE_RUNNING: wend
                    iResult = 0
                case ADS_SERVICE_RUNNING
                    iResult = 1
            end select
        case kStopSpooler
            select case (objService.Status)
                case ADS_SERVICE_RUNNING
                    objService.Stop
                    while objService.Status <> ADS_SERVICE_STOPPED: wend
                    iResult = 0
                case ADS_SERVICE_STOPPED

```

```
        iResult = 1
    end select
end select
gg_ManageSpooler = iResult
end function
```

```
,-----  
,---                フォルダのコピー                ---  
,-----
```

```
function gg_CopyFolder(strSrc, strDst)
    on error resume next
    dim iResult
    dim objFSO
    dim objFolder
    '---
    iResult = -999
    set objFSO = WScript.CreateObject("Scripting.FileSystemObject")
    set objFolder = objFSO.GetFolder(strSrc)
    objFSO.CopyFolder objFolder.path, strDst, true
    '---
    if Err = kErrorSuccess then
        iResult = 0
    else
        wscript.echo "gg_CopyFolder : Err = " & Err.Description
        iResult = -1
    end if
    Err.Clear
    gg_CopyFolder = iResult
end function
```

```
,-----  
,---                ローカルコンピュータの指定レジストリのインポート                ---  
,-----
```

```
function gg_ImportRegistry(strFile)
    on error resume next
    dim iResult
    dim strCommand
    dim objShell
    '---
    iResult = -999
    strCommand = "reg import "" & strFile & """"
    set objShell = WScript.CreateObject("WScript.Shell")
    objShell.Run strCommand, 7, true
    '---
    iResult = 0
    gg_ImportRegistry = iResult
end function
```

```
'-----  
'--- コマンドラインのパージャー ---  
'-----
```

```
function ParseCommandLine(strServer)  
  on error resume next  
  dim oArgs  
  dim iIndex  
'---  
  strAction = "(none)"  
  iIndex = 0  
  set oArgs = wscript.Arguments  
'---  
  while iIndex < oArgs.Count  
    select case LCase(oArgs(iIndex))  
      case "--spooler"  
        iIndex = iIndex + 1  
        select case LCase(oArgs(iIndex))  
          case "start"  
            strAction = kStartSpooler  
          case "stop"  
            strAction = kStopSpooler  
        end select  
      case "--server"  
        iIndex = iIndex + 1  
        strServer = RemoveBackslashes(oArgs(iIndex))  
      case else  
        exit function  
    end select  
    iIndex = iIndex + 1  
  wend  
  ParseCommandLine = 0  
end function
```

```
'---
```

```
function RemoveBackslashes(strServer)  
  dim strRet  
  strRet = strServer  
  if Left(strServer, 2) = "¥¥" and Len(strServer) > 2 then  
    strRet = Mid(strServer, 3)  
  end if  
  RemoveBackslashes = strRet  
end function
```

Last reviewed: Sep 22, 2010
Status: DRAFT
Ref: NONE
Copyright © 2010 kitASP Corporation