

```
# 16 Marzo 2015 Script per eseguire il backup dell'infrastruttura Hyper-V a partire da un elenco di VM contenuto in
# Vmlist.txt ed invio di notifiche sullo stato del backup in mail
# questa versione crea un VHDX per ogni sessione di backup eseguito i backup e salva le vm all'interno
# è possibile specificare in n° di giorni di cui conservare il backup n modo distinto dal n° di giorni di cui conservare i log
```

```
$Vhdbackupdrive = "\\server" # Drive (oppure nome di rete) in cui è contenuto il VHDX per il salvataggio
$Vhdbackupfolder = "share" # percorso del server (oppure Share) dove è contenuto il VHDX per il backup giornaliero
$drivebck = "q:" # Drive su cui eseguire il salvataggio (Unità Mount del VHDX quotidiano)
Show-Get-Date
$DateVhd = Show-Tostring("yyyy-MM-dd-HH-mm-ss")
$VmlistBackup = Get-Content C:\Windows\ServerBackup\Vmlist.txt
$VMOBJect = Get-Content C:\Windows\ServerBackup\Vmlist.txt | Measure-Object -Line
$ObjECT.VMNumber = $VMOBJect.Lines
$NomeHost = Get-WMIObject win32_ComputerSystem | Select-Object -ExpandProperty name # Recupero il nome Host su cui è eseguito lo script
$Logpath = "C:\Windows\ServerBackup\Log" # Percorso del file di Log
$Logfile = (Get-Date -Format "yyyy-MM-dd") + "-bck-" + $NomeHost + ".log" # Composizione del nome file di log a partire dalla data corrente
$LogDir = "C:\Windows\Logs\WindowsServerBackup\" # Cartella di sistema dove Windows Server Backup archivia i Log
$PreviousEventsDay = -1 # numero di giorni precedenti alla data corrente per la rilevazione nel registro di eventi classificati come errore
$DateToCompare = (Get-Date).AddDays($PreviousEventsDay) # Variabile data riportata al valore odierno diminuito del n° di giorni di cui si vuole avere il report degli eventi
$SbckPath = $Vhdbackupdrive + $Vhdbackupfolder # Costruisco il path in cui sono contenuti i VHDX di Backup
$SbckCopiesRetained = (Get-Date).AddDays(-3) # numero di Giorni di cui si vuole mantenere il Backup Online. Modificare il valore AddDays(n)
$LogCopiesRetained = (Get-Date).AddDays(-10) # numero di Giorni di cui si vogliono mantenere i Log. Modificare il valore AddDays(n)

#dichiarazione della funzione per invio mail
function SendMail ($Event)
{
    #Generazione della Mail
    #Variabili di impostazione
    $MailFrom = $NomeHost + "@dominio.it"
    $MailTo = "notifiche.backup@dominio.it"
    $Subject = $MessageObject
    $SMTPServer = "mail.dominio.it"
    $SMTPPort = "25"
    $UserName = "hypervtest00" # decommentare se il mailserver richiede autenticazione per la Spedizione
    $Password = "1qwertzuiop" # decommentare se il mailserver richiede autenticazione per la Spedizione
    #Composizione del Messaggio
    $Message = New-Object System.Net.Mail.MailMessage
    $Message.Subject = $Subject
    $Message.Body = $Event
    $Message.To.Add($MailTo)
    $Message.From = $MailFrom
# Iterazione della variabile $VhdbackupLogFiles per allegare ogni file di log alla mail
foreach ($VhdbackupLogFile in $VhdbackupLogFiles)
{
    $Message.Attachments.Add($LogDir + $VhdbackupLogFile)
}
# Alliego il file di log generato per l'intero processo di backup
$Message.Attachments.Add($Logpath + $LogFile)
#Istanza oggetto SMTPCLIENT
$SMTP = New-Object System.Net.Mail.SmtpClient($SMTPServer, $SMTPPort);
$SMTP.EnableSsl = $False # decommentare se il mailserver richiede autenticazione su canale cifrato
$SMTP.Credentials = New-Object System.Net.NetworkCredential($UserName, $Password);
$SMTP.Send($Message)
#Includere sempre il comando sottostante in quanto Powershell mantiene il file allegato in lock finche' è istanziato il processo PS
$Message.Dispose()
}

##### NON MODIFICARE VARIABILI AL DI SOTTO DI QUESTA LINEA #####

# creo il VHDX per il giorno di copia
$DiskPath = ($Vhdbackupdrive + $Vhdbackupfolder + $DateVhd + "-" + $NomeHost + ".vhdx")
New-VHD -Path $DiskPath -Dynamic -SizeBytes 100Gb
Mount-DiskImage -ImagePath $DiskPath
$DiskNumber = (Get-DiskImage -ImagePath $DiskPath).Number
Initialize-Disk -Number $DiskNumber -PartitionStyle MBR
New-Partition -DiskNumber $DiskNumber -UseMaximumSize -DriveLetter Q | Format-Volume -Filesystem NTFS -NewFileSystemLabel $DateVhd Confirm:$False

# Test controllo presenza disco VHDX montato (decommentare per simulazione errore)
# Dismount-DiskImage -ImagePath $DiskPath

# Controllo la presenza del drive per i backup
$DriveOK = Test-Path -Path $DriveBck # Se ok procedo con il backup
if ($DriveOK -eq $true)
{
    Write-Host "Drive backup presente"
}
# Se non presente allego il file con l'elenco VM non copiate ed esco con notifica in Mail
else {
    $MessageObject = " Backup Hyper-V " + $NomeHost + " ATTENZIONE!!! procedura copie non eseguita "
    $LogDir = "c:\windows\serverbackup\" $VhdbackupLogFiles = "Vmlist.txt"
    SendMail " drive di backup non pronto o non esistente elenco delle VM non salvate Allegato alla Mail"
    exit
}

# Iterazione della variabile ed esecuzione del comando di backup VM per ognuna
foreach ($VMachine in $VmlistBackup)
{
    $Policy = New-WBPolicy
    $VirtualMachines = Get-WBVirtualMachine | ? VMName -eq $VMachine
    Add-WBVirtualMachine -Policy $Policy -VirtualMachines $VirtualMachines
    $Destinazione = New-WBBackupTarget -VolumePath $DriveBck -verbose
    Get-Date -Format dd-MM-yyyy--HH-mm | Out-File ($Logpath + $LogFile) -Append
    Add-Content -path ($Logpath + $LogFile) -value (" ##### INIZIO COPIA VM " + ($VMachine))
    Add-WBBackupTarget -Policy $Policy -Target $Destinazione
    # attivo il transcript dell'output del comando sul file di log per il CMDLET start-wbbackup
    Start-Transcript -Path ($Logpath + $LogFile) -Append -NoClobber
    Start-WBBackup -Policy $Policy
    # Fermo la redirection dell'output
    Stop-Transcript
    get-wbsummary | Out-File ($Logpath + $LogFile) -Append
    Add-Content -path ($Logpath + $LogFile) -value (" ##### FINE COPIA VM " + ($VMachine))
    Add-Content -path ($Logpath + $LogFile) -value (" ")
    Add-Content -path ($Logpath + $LogFile) -value (" ")
}

# Esego la disattivazione del VHD
Dismount-DiskImage -ImagePath $DiskPath -Verbose | Out-File -FilePath ($Logpath + $LogFile) -Append
# Eliminazione backup obsoleti e registrazione sul file di Log
Get-ChildItem -Path $SbckPath -Recurse -Force | Where-Object { $_.PSIsContainer -and $_.CreationTime -lt $SbckCopiesRetained } | Remove-Item -Force -Verbose
# Eliminazione Log obsoleti e registrazione sul file di Log
Get-ChildItem -Path $Logpath -Recurse -Force | Where-Object { $_.PSIsContainer -and $_.CreationTime -lt $LogCopiesRetained } | Remove-Item -Force -Verbose
#invio delle Notifiche dopo controllo registro eventi e recupero dei due ultimi file di log generati #interrogazione Registro Eventi
$EventsFromDate = (get-date).AddDays($PreviousEventsDay)
$HyperVErr = Get-WinEvent -ErrorAction SilentlyContinue -FilterHashtable @{logname='Microsoft-Windows-Backup';Level=2 ; StartTime=$EventsFromDate }
#Level=4 Information
#Level=2 Error

# Individuazione dei file di log per l'invio in allegato alla mail di report
# Mantenere la rilevazione dei file DOPO l'esecuzione del backup in quanto i files vengono generati ad avvio backup avvenuto se rilevati prima viene inviato un file della versione precedente
$VhdbackupLogFiles = @(Get-ChildItem $LogDir | Where-Object { $_.Extension -eq ".log" } | Where-Object { $_.LastWriteTime -gt $DateToCompare})

if ($HyperVErr -eq $null)
{
    $HyperVErr = $NomeHost + " Non ci sono eventi di Backup classificati come malfunzionamento all'interno del Registro Eventi negli ultimi " + $PreviousEventsDay + " giorni"
    $MessageObject = $NomeHost + " Backup Hyper-V terminato correttamente "
    SendMail $HyperVErr
    Write-Host $HyperVErr
}
else {
    $EventIDMessage = $HyperVErr.Message
    $MessageObject = $NomeHost + " Backup Hyper-V terminato con errori "
    SendMail $EventIDMessage
    Write-Host $EventIDMessage
}
}
```