

Ejercicios de PowerShell: ordenar direcciones IP remotamente y crear un disco virtual (Invoke-Command)

Necesita winrm, el módulo TCPServer e Hyper-V habilitado

```
$menu=@"  
1 Ordenar IPs  
2 Crear VHD  
S Salir
```

```
Selecione una tarea o pulse S para salir  
"@
```

```
$equipos = Get-ADComputer -Filter * | select -ExpandProperty  
name
```

```
Function Invoke-Menu {  
[cmdletbinding()]  
Param(  
[Parameter(Position=0,Mandatory=$True,HelpMessage="Enter your  
menu text")]  
[ValidateNotNullOrEmpty()]  
[string]$Menu,  
[Parameter(Position=1)]  
[ValidateNotNullOrEmpty()]  
[string]$Title = "My Menu",  
[Alias("cls")]  
[switch]$ClearScreen  
)
```

```
#clear the screen if requested  
if ($ClearScreen) {  
Clear-Host  
}
```

```

#build the menu prompt
$menuPrompt = $title
#add a return
$menuprompt+="n"
#add an underline
$menuprompt+="-"*$title.Length
#add another return
$menuprompt+="n"
#add the menu
$menuPrompt+=$menu

Read-Host -Prompt $menuprompt

}

Do {
    Switch (Invoke-Menu -menu $menu -title "Tareas de
administración" -clear) {
        "1" {$i = 20
            $equipos | where {$_ -notmatch "SERVER"} | %{
                $ipn = "192.168.1." + $i++
                Write-Host "Cambiando IP de $_" -ForegroundColor
Green
                sleep -seconds 2
                Enter-PSSession -ComputerName $_
                Invoke-Command -ComputerName $_ -ScriptBlock {
                    $ipa = Get-NetIPAddress -InterfaceIndex 7
-AddressFamily IPv4 | select -ExpandProperty IPAddress
                    New-NetIPAddress -IPAddress $using:ipn -
AddressFamily IPv4 -InterfaceIndex 7 -PrefixLength 24
                    Remove-NetIPAddress -IPAddress "$ipa" -
Confirm:$False
                } -AsJob
                Exit-PSSession
            }
            Clear-DnsClientCache
            Read-Host "IPs cambiadas con éxito, presione
ENTER para volver al menú"
        }
        "2" {Write-Host "Por favor, especifique los parámetros
deseados" -ForegroundColor Green
    }
}

```

```
sleep -seconds 2
```

```
    $cliente = Read-Host "Introduzca el equipo donde  
quiere crear el VHD"
```

```
    $nombre = Read-Host "Introduzca el nombre del  
disco"
```

```
    $archivo = "C:\\" + $nombre + ".vhd"
```

```
    $letra = 65..90 | % {[char]$_} | Where-Object {$_ -  
notin (Get-Partition | select -ExpandProperty DriveLetter)} |  
Get-Random
```

```
    $capacidad = Read-Host "Introduzca la capacidad en  
MB"
```

```
    $secpasswd = ConvertTo-SecureString  
"Escarabajo.11" -AsPlainText -Force
```

```
    $mycreds = New-Object  
System.Management.Automation.PSCredential  
("red\administrador", $secpasswd)
```

```
Invoke-TCPServer -Computername $cliente -Port 1655  
-Credential ($mycreds) -Verbose
```

```
    $command = @"
```

```
        New-VHD -Path $archivo -SizeBytes  
([int]$capacidad* 1MB)
```

```
    Mount-VHD $archivo -Passthru | Initialize-Disk -  
PassThru |
```

```
        New-Partition -UseMaximumSize -DriveLetter  
"$letra" |
```

```
        Format-Volume -FileSystem NTFS -NewFileSystemLabel  
$nombre -Confirm:0
```

```
    "@
```

```
    Send-Command -Computername $cliente -Port 1655 -  
Command "$command"
```

```
    -Verbose -Credential ($mycreds)
```

```
    Read-Host "VHD creado con éxito, presione ENTER para  
volver al menú"
```

```
    }
```

```
"S" {Write-Host "Adiós!" -ForegroundColor Magenta
```

```
Return
```

```
}
```

```
        Default {Write-Warning "Por favor, introduzca una opción  
válida"  
                sleep -milliseconds 2400}  
    }  
} While ($True)
```