

Administración de Sistemas Operativos	
Aplicación de scripting en SO PowerShell Examen	
Fecha 27 Octubre 2021	
El trabajo ha de ser perfectamente legible. Haz los recortes necesarios para justificar el trabajo, primero del script o comando y luego de su salida. Guárdalo en un pdf con encabezado tu nombre y fecha y súbelo a classroom ASO examen_bloque2. (no cumplir el formato resta 1 del total obtenido) Se valora que todos en los ejercicios se incluyan comentarios explicando el flujo del programa en su caso y/o se valorará la claridad y precisión así como la estética de los mismos.	

1. Tienes un equipo con 2 tarjetas de red (Puente/NAT y red interna1), y 2 discos duros, el primero con Windows 10 instalado.
Prepara el equipo mediante un script de comandos **PowerShell** de la siguiente manera:

- Nombre equipo windows10: ordenador1 (0,2 puntos)

```

Examen ASO David Prado [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Administrador: Windows PowerShell
PS C:\Windows\system32>
PS C:\Windows\system32> Rename-Computer ordenador1
ADVERTENCIA: Los cambios serán efectivos tras reiniciar el equipo DESKTOP-17C67D0.
PS C:\Windows\system32>
  
```

- Nombre de la interface de la tarjeta NAT: internet (0,2 puntos)
- Nombre de la interface de la tarjeta en red interna: red_LAN (0,2 puntos)
Mirando las MACs de las tarjetas de red puedo determinar cual es la interna y la externa

```

Examen ASO David Prado [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Administrador: Windows PowerShell
PS C:\Windows\system32> Get-NetAdapter

Name                InterfaceDescription          ifIndex Status      MacAddress
-----
Ethernet 2          Intel(R) PRO/1000 MT Desktop  3      Up          08-00-27-AA-4B-D6
Ethernet            Intel(R) PRO/1000 MT Desktop  4      Up          08-00-27-EA-4B-2D

PS C:\Windows\system32> Rename-NetAdapter -Name Ethernet -NewName internet
PS C:\Windows\system32> Rename-NetAdapter -Name 'Ethernet 2' -NewName red_lan
PS C:\Windows\system32> Get-NetAdapter

Name                InterfaceDescription          ifIndex Status      MacAddress
-----
red_lan             Intel(R) PRO/1000 MT Desktop  3      Up          08-00-27-AA-4B-D6
internet            Intel(R) PRO/1000 MT Desktop  4      Up          08-00-27-EA-4B-2D

PS C:\Windows\system32>
  
```

- Ponle la IP fija a la tarjeta red_LAN: 192.168.30.xx, donde xx es tu nº de equipo. (0,2puntos)

Con el comando **get-netipaddress** puedo saber el numero index de cada tarjeta de red

```

PS C:\Users\davidprado> Get-NetIPAddress

IPAddress           : fe80::7972:bbcb:913e:4de1%3
InterfaceIndex      : 3
InterfaceAlias       : red_lan
AddressFamily        : IPv6
Type                 : Unicast
PrefixLength         : 64
  
```

```
PS C:\Users\davidprado> New-NetIPAddress -InterfaceIndex 3 -IPAddress 192.168.30.218 -PrefixLength 24 -DefaultGateway 192.168.30.1

IPAddress      : 192.168.30.218
InterfaceIndex : 3
InterfaceAlias  : red_lan
AddressFamily   : IPv4
Type            : Unicast
```

- Crea la carpeta examen en el directorio Documentos. (0,2 puntos)

Examen ASO David Prado [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Administrador: Windows PowerShell

```
PS C:\Users\davidprado> New-Item -ItemType Directory -Path .\Documents\ -Name examen

Directorio: C:\Users\davidprado\Documents
```

Mode	LastWriteTime	Length	Name
d----	27/10/2021 13:03		examen

```
PS C:\Users\davidprado>
```

- Crea en el disco 2, 3 particiones con formato NTFS, de tamaño 20 GB cada uno. (0,2 puntos)

```
PS C:\Users\davidprado> New-Partition -DiskNumber 1 -Size 10GB -DriveLetter x

DiskPath: \\?\scsi#disk&ven_vbox&prod_harddisk#4&2617aeae&0&020000#{53f56307-b6bf-11d0-94f1-000000000000}

PartitionNumber DriveLetter Offset              Size Type
-----
2                x          135266304          10 GB Basic

PS C:\Users\davidprado> New-Partition -DiskNumber 2 -Size 10GB -DriveLetter g

DiskPath: \\?\scsi#disk&ven_vbox&prod_harddisk#4&2617aeae&0&030000#{53f56307-b6bf-11d0-94f1-000000000000}

PartitionNumber DriveLetter Offset              Size Type
-----
2                g          135266304          10 GB Basic

PS C:\Users\davidprado> Format-volume -DriveLetter x -FileSystem NTFS

DriveLetter FriendlyName FileSystemType DriveType HealthStatus OperationalStatus SizeRemainin
-----
x              NTFS          Fixed      Healthy      OK              9.96 G

PS C:\Users\davidprado> Format-volume -DriveLetter g -FileSystem NTFS

DriveLetter FriendlyName FileSystemType DriveType HealthStatus OperationalStatus SizeRemainin
-----
g              NTFS          Fixed      Healthy      OK              9.96 G
```

- Crea en el disco 1, dos particiones de 30 GB, una con formato exfat y otra NTFS con letras de unidad M y otra N. (0,2 puntos)

Yo las he creado en el disco 2 y 3 ya que en el dsico 1 no me dejaba a pesar de que tenia 30 gigas libres

```
PS C:\Users\davidprado> New-Partition -DiskNumber 1 -Size 5GB -DriveLetter m

DiskPath: \\?\scsi#disk&ven_vbox&prod_harddisk#4&2617aeae&0&020000#{53f56307-b6bf-11d0-94f1-000000000000}

PartitionNumber DriveLetter Offset              Size Type
-----
3                M          10872684544          5 GB Basic

PS C:\Users\davidprado> New-Partition -DiskNumber 2 -Size 5GB -DriveLetter n

DiskPath: \\?\scsi#disk&ven_vbox&prod_harddisk#4&2617aeae&0&030000#{53f56307-b6bf-11d0-94f2-000000000000}

PartitionNumber DriveLetter Offset              Size Type
-----
3                N          10872684544          5 GB Basic
```

```
PS C:\Users\davidprado> Format-Volume -DriveLetter m -FileSystem exFAT
```

DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
M		Unknown	Fixed	Healthy	OK	5 GB	5 GB

```
PS C:\Users\davidprado> Format-Volume -DriveLetter m -FileSystem NTFS
```

DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
M		NTFS	Fixed	Healthy	OK	4.97 GB	5 GB

2. Si tenemos en cuenta la siguiente clasificación de ordenadores de sobremesa según sus usos:

- *Aplicaciones de oficina*: la mayoría, tanto en el ámbito doméstico como laboral, se utilizan principalmente para ejecutar aplicaciones de oficina, como tratamiento de textos, navegar o correo electrónico. procesador Corie i3 y una **memoria RAM 4G**
- *Videojuegos*: los ordenadores de sobremesa para jugar deben ser potentes en todos los aspectos, procesador, tarjeta gráfica, tarjeta de sonido, disco duro y pantalla. Disponen del hardware más potente y están optimizados para obtener el máximo rendimiento del procesador y la tarjeta gráfica. una buena tarjeta gráfica y una memoria RAM de al menos **8GB a DDR3**, además de un buen sistema de refrigeración que evita el sobrecalentamiento de tu equipo.
- *Multimedia*: para actividades multimedia conviene priorizar un disco duro grande y una elevada conectividad, con USB, HDMI, ranura para tarjetas de memoria, Ethernet, Wi-Fi, Bluetooth, DVI y grabadora DVD. PC sobremesa con un buen procesador, una **gran memoria RAM de al menos 8GB** pero si tu presupuesto te lo permite, mejor de **16GB o 32GB** y dos o cuatro módulos y una tarjeta gráfica muy potente para soportar las exigencias propias del uso que vas a dar a tu equipo

Se pide crear un script que me muestre, indicando lo q se va a visionar, la siguiente información concreta de tu equipo:

- Tamaño del disco1
- Tipo de disco duro del disco1
- Marca o fabricante del procesador
- Velocidad de trabajo
- Tamaño de la memoria RAM máxima que soporta
- Tamaño de memoria RAM efectiva
- Modelo/fabricante de placa base `wmic baseboard`
- Tipo de tarjeta gráfica `wmic path win32_VideoController`
- Tipo de tarjeta de sonido
- Tipo de refrigeración
- `Get-WmiObject Win32_Fan`
- `Get-WmiObject Win32_HeatPipe`
- `Get-WmiObject Win32_Refrigeration`
- `Get-WmiObject Win32_TemperatureProbe`

(Concreta con información relevante que tú valores como técnico) Sin ninguna línea irrelevante (serán penalizadas), en función de esa información y mande la información a un archivo txt de nombre `mi_equipo.txt` y te indique para lo que mejor esté orientado (oficina, multimedia o juegos). (2 puntos comandos exactos+ testeo y script)

```

1  #ejercicio 2 por david prado
2
3  Write-Host "Este equipo tiene"
4
5  write-host "Tamaño del disco1"
6  Get-Disk -Number 0 | Select-Object friendlyname, size
7
8  write-host "Tipo de disco duro del disco1"
9
10 (get-physicaldisk -SerialNumber (Get-Disk -Number 0 | Select-Object serialnumber).serialnumber).mediatype
11
12 write-host "Marca o fabricante del procesador"
13
14 if ((Get-wmiobject win32_processor) -match "intel")
15 {
16     {Write-Host "el procesador es intel"}
17 }
18 else
19 {
20     {Write-Host "el procesador es amd"}
21 }
22
23 write-host "Velocidad de trabajo del procesador representada en Hz"
24 (Get-wmiobject win32_processor).maxclockspeed
25
26 write-host "Tamaño de memoria RAM efectiva"
27 Get-wmiobject Win32_PhysicalMemory
28
29 write-host "Modelo/fabricante de placa base wmic baseboard"
30 Get-wmiobject Win32_BaseBoard | Select-Object manufacturer
31 write-host "Tipo de tarjeta gráfica"
32 Get-wmiobject win32_VideoController | Select-Object caption
33 write-host "Tipo de tarjeta de sonido"
34 Get-wmiobject Win32_SoundDevice

```

3. Se pide un script, que a partir de una lista (puede ser un csv/ o un txt) con el nombre, apellido1, edad (introduce 2 mayores de edad y 3 menores de edad) y sexo (hombre/mujer) de 5 personas, genere:
 - Una carpeta para cada usuario, llamada nombre.Apellido1
 - Cree un usuario local para cada uno, con la contraseña pepe@2 (1,2 puntos)

Y por otro lado genera otro tipo menú, que te haga lo siguiente:

Listar sólo los nombres de la lista.

- Listar los mayores de 18 años.
- Listar la cantidad de menores.
- Salir.

(1.2 puntos)

4. Se pide un script que me convierta una imagen .bmp, usando parámetros posicionales, en .jpg. y luego me liste todas las imágenes jpg, de la carpeta imágenes del equipo. (1 punto)
5. Se pide un script que te dé distintas opciones a realizar sobre directorios:

Crear, mover, borrar, ver el contenido (distinguiendo entre directorios y ficheros), copiar un directorio, a partir de una ruta y el nombre de una carpeta, primero habrá de testear si existe. Cada vez que pulse una opción tenemos que introducir los parámetros necesarios de esa opción,

(1,4 puntos)

```
ej 5 por david prado.ps1 X Sin título2.ps1*
1 #ej5 por david prado
2
3 $rutadirectorio=Read-Host "escribe la ruta de un directorio"
4 if ( test-path -path $rutadirectorio)
5 {
6     write-host '1. Mover'
7     write-host '2. Borrar'
8     write-host '3. ver el contenido'
9     write-host '4. copiarle en otro sitio'
10    write-host '5. salir'
11
12    $opcion=Read-Host 'que quieres hacer?'
13    switch ($opcion){
14    1{
15
16        $destino=read-host "introduce la ruta de destino del directorio"
17        Move-Item -Path $rutadirectorio -Destination $destino
18
19    }
20    #####
21    2{
22
23        Write-Host " se ha eliminado el directorio" $rutadirectorio
24        Remove-Item $rutadirectorio -Recurse
25    }
26    #####
27    3{
28
29        Write-Host 'este es el contenido de' $rutadirectorio
30        Get-ChildItem -Path $rutadirectorio
31    }
32    }
33    #####
34    4{
35
36        $destino=Read-Host "donde quieres copiar este directorio?"
37        Copy-Item -Path $rutadirectorio -Destination $destino -Recurse
38        Write-Host "el directorio se ha movido a " $destino
39
40    }
41
42    #####
43    5{
44        return
45    }
46    }
47 }
48
9 else
0 {
1 $crear=read-host "no existe, quieres crearlo s\n"
2 if ( $crear -eq 's')
3
4 {
5     Write-Host "la carpeta de ha creado"
6     New-Item -ItemType Directory -Path $rutadirectorio
7 }
8 else
9
0 {echo 'no'}
1
2 }
3
```

Para hacer la muestra del script he creado el directorio hola dentro de descargas

```

ej 5 por david prado.ps1 X Sin título2.ps1*
PS C:\Users\davidprado\Downloads> C:\Users\davidprado\Documents\ej 5 por david prado.ps1
escribe la ruta de un directorio: C:\Users\davidprado\Downloads\hola
1. Mover
2. Borrar
3. ver el contenido
4. copiarle en otro sitio
5. salir
que quieres hacer?: 3
este es el contenido de C:\Users\davidprado\Downloads\hola

    Directorio: C:\Users\davidprado\Downloads\hola

Mode                LastWriteTime         Length Name
----                -
-a-----         27/10/2021    13:38             0 hola2.txt

PS C:\Users\davidprado\Downloads> C:\Users\davidprado\Documents\ej 5 por david prado.ps1
escribe la ruta de un directorio: C:\Users\davidprado\Downloads\hola
1. Mover
2. Borrar
3. ver el contenido
4. copiarle en otro sitio
5. salir
que quieres hacer?: 1
introduce la ruta de destino del directorio: C:\Users\davidprado\Documents

PS C:\Users\davidprado\Downloads> C:\Users\davidprado\Documents\ej 5 por david prado.ps1
escribe la ruta de un directorio: C:\Users\davidprado\Documents\hola
1. Mover
2. Borrar
3. ver el contenido
4. copiarle en otro sitio
5. salir
que quieres hacer?: 2
se ha eliminado el directorio C:\Users\davidprado\Documents\hola

PS C:\Users\davidprado\Downloads> ls

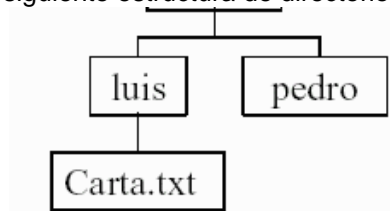
PS C:\Users\davidprado\Downloads> ls C:\Users\davidprado\Documents

    Directorio: C:\Users\davidprado\Documents

Mode                LastWriteTime         Length Name
----                -
d-----         27/10/2021    13:03             examen
-a-----         27/10/2021    13:38          1311 ej 5 por david prado.ps1

```

6. Crea la siguiente estructura de directorios en tu directorio de trabajo:



Donde Luis y Pedro son directorios y Carta.txt un documento de texto de la carpeta Luis. (0,3 puntos)

```

PS C:\Users\davidprado> New-Item -Type Directory -Name luis _
PS C:\Users\davidprado> New-Item -Type Directory -Name pedro_
PS C:\Users\davidprado> New-Item -Type file -path .\luis\carta.txt

```

```

PS C:\Users\davidprado> ls

    Directorio: C:\Users\davidprado

Mode                LastWriteTime         Length Name
----                -
d-r---            14/09/2021         9:31      Contacts
d-r---            14/09/2021         9:31      Desktop
d-r---            27/10/2021        13:53      Documents
d-r---            27/10/2021        13:39      Downloads
d-r---            14/09/2021         9:31      Favorites
d-r---            14/09/2021         9:31      Links
d-----          27/10/2021        14:01      luis
d-r---            14/09/2021         9:31      Music
d-r---            27/10/2021        11:42      OneDrive
d-----          27/10/2021        14:01      pedro
d-r---            14/09/2021         9:32      Pictures
d-r---            14/09/2021         9:31      Saved Games
d-r---            14/09/2021         9:32      Searches
d-r---            14/09/2021         9:31      Videos

PS C:\Users\davidprado> ls -Path .\luis\

    Directorio: C:\Users\davidprado\luis

Mode                LastWriteTime         Length Name
----                -
-a----          27/10/2021        14:01           0 carta.txt

PS C:\Users\davidprado>

```

- Crea la siguiente línea al documento carta.txt: "Los cmdlets que comienzan con start son:"(0,2 puntos)

```

PS C:\Users\davidprado> Add-Content -Path .\luis\carta.txt -Value "Los cmdlets que comienzan con start son:"
PS C:\Users\davidprado>

```

- Añade el listado con los cmdlets que comienzan con start al documento carta.txt. (0,2 puntos)

```

PS C:\Users\davidprado> $cmdl=get-command -CommandType Cmdlet -name start*
PS C:\Users\davidprado> Add-Content -Path .\luis\carta.txt -Value $cmdl
PS C:\Users\davidprado> Get-Content -Path .\luis\carta.txt
Los cmdlets que comienzan con start son:
Start-BitsTransfer
Start-DscConfiguration
Start-DtcDiagnosticResourceManager
Start-Job
Start-Process
Start-Service
Start-Sleep
Start-Transaction
Start-Transcript
PS C:\Users\davidprado>

```

- Obtén el tamaño del documento carta.txt (0,2 puntos)

```
1 Write-Host "carta pesa..."
2 (gci -Path c:\users\davidprado\luis\carta.txt | Select-Object length).length
3 Write-Host "bytes"
4
```

PS C:\Users\davidprado\Downloads> C:\Users\davidprado\Documents\Sin título5.ps1
carta pesa...
213
bytes

PS C:\Users\davidprado\Downloads>

TEST (1 punto: las bien +0.1, las mal -0.05. Sombrea en amarillo la opción elegida)

1. Para completar un cmdlet en la siguiente línea usas:

- a) F7
- b) Tabulador**
- c) -(guión)
- d) `(acento grave)

2. Powershell está incorporada en todas las versiones de Windows:

- a) Sólo en los Windows server
- b) Cada versión de Windows tiene su versión.**
- c) Todos los Windows tiene la versión 5 o superior de powershell.
- d) Las versiones de powershell no son actualizables.

3. Cuando intentas ejecutar un script por primera vez con PowerShell:

- a) has de ejecutarlo como administrador y habilitar la ejecución de scripts con set-executionPolicy Unrestricted.**
- b) Se ejecuta como cualquier script.
- c) Debes de importar el modulo de Active Directory.
- d) Debes de habilitarlo con get-member.

4. Para crear el fichero texto1.txt en el directorio D:\Datos\textos. Cual no es correcto:

- a) New-Item "D:\Datos\textos" texto1.txt file
- b) Write-host " " > D:\Datos\Textos\texto1.txt
- c) New-Item -ItemType file -path "D:\Datos\textos"**
- d) New-Item file "D:\Datos\textos"**

5. Para crear la variable mareo con el valor esto es lo que hay, cuál de las siguientes no es correcta:

- a) \$mareo="esto es lo que hay"**
- b) New-variable -name mareo -value "esto es lo que hay"
- c) Set-variable -name mareo -value "esto es lo que hay"
- d) Get-variable -name mareo -value "esto es lo que hay"

6. El identificado del primer parámetro posicional es:

- a) %1
- b) Args[0]**
- c) \$_
- d) \$?

7. El primer valor del array \$array es:

- a) \$array[0]**
- b) \$array[1]
- c) Args[0]
- d) \$1

8. Para obtener solamente la lista únicamente de las propiedades del objeto get-childitem, escribimos:

- a) PS>get-childitem | get-member
- b) PS>get-childitem -property
- c) PS>get-childitem | get-member | where-object{\$_.membertype -eq "property"}**
- d) PS>get-childitem | where-object{\$_.membertype -eq "property"}

9. La expresión `get-service | where-object{$.estatus -eq "stopped"}`

- a) Lista los servicios que están parados.
- b) Lista los procesos que contengan la palabra stopped.
- c) Lista los procesos que están parados.
- d) Lista los procesos que no están parados.

10. La expresión `get-childitem | where-object{$_.length -ge 1000 }`

- a) Muestra los ficheros que tienen 1000MB o más del directorio actual.
- b) Muestra los ficheros que tienen 1000MB o menos del directorio actual.
- c) Muestra los ficheros que tienen 1000 bytes o más del directorio actual.
- d) Muestra los ficheros que tienen 1000 Bytes o más del directorio raíz.