

Netsh, como configurar parámetros de red con scripts en Windows

En Windows también se puede configurar la red a través de comandos. Estos comandos también se pueden incluir en scripts.

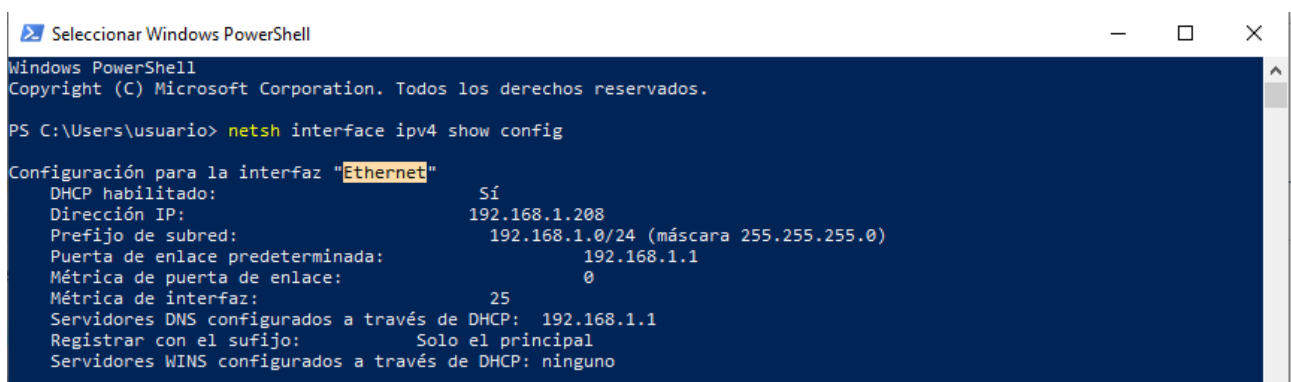
Abrimos un Powershell(Inicio->Windows PowerShell) para utilizar el comando netsh y crear scripts. Hay dos tipos de PowerShell:

- de entorno de línea de comandos, **PowerShell**
- de entorno de Shell interactivo **PowerShell ISE** que proporciona una interfaz gráfica de usuario para crear, modificar, depurar y ejecutar scripts. Sin embargo, es importante tener en cuenta que PowerShell ISE ya no recibe actualizaciones y Microsoft está recomendando el uso de Visual Studio Code como un entorno de desarrollo integrado, más moderno y avanzado.

Puedes probar estos comandos en una máquina virtual Windows 10.

netsh

- **Visualizar** la configuración de red
netsh interface ipv4 show config



```
Selecionar Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\usuario> netsh interface ipv4 show config

Configuración para la interfaz "Ethernet"
DHCP habilitado:                Sí
Dirección IP:                   192.168.1.208
Prefijo de subred:              192.168.1.0/24 (máscara 255.255.255.0)
Puerta de enlace predeterminada: 192.168.1.1
Métrica de puerta de enlace:    0
Métrica de interfaz:            25
Servidores DNS configurados a través de DHCP: 192.168.1.1
Registrar con el sufijo:        Solo el principal
Servidores WINS configurados a través de DHCP: ninguno
```

En la primera línea figura el nombre de la interfaz, "Ethernet"

```
netsh interface ipv4 show config name="Ethernet 5"
Get-NetIPAddress
Get-NetIPAddress -AddressFamily IPv4
```

```
Windows PowerShell
PS C:\Users\usuario> Get-NetIPAddress -AddressFamily IPv4

IPAddress      : 192.168.56.1
InterfaceIndex  : 16
InterfaceAlias  : Ethernet 2
AddressFamily   : IPv4
Type            : Unicast
PrefixLength    : 24
PrefixOrigin    : Manual
SuffixOrigin     : Manual
AddressState     : Preferred
ValidLifetime   : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
SkipAsSource     : False
PolicyStore     : ActiveStore

IPAddress      : 192.168.1.208
InterfaceIndex  : 17
InterfaceAlias  : Ethernet
AddressFamily   : IPv4
Type            : Unicast
PrefixLength    : 24
PrefixOrigin    : Dhcp
SuffixOrigin     : Dhcp
AddressState     : Preferred
ValidLifetime   : 00:53:29
PreferredLifetime : 00:53:29
SkipAsSource     : False
PolicyStore     : ActiveStore
```

Parecido a la primera imagen pero con otro formato.

Get-NetIPAddress -AddressFamily IPv4 -InterfaceAlias 'Ethernet 5'

- **Poner dirección IPv4 estática** a una interface
netsh interface ipv4 set address name="Nombre de la interfaz" static Dirección_IP
Máscara de Subred GATEWAY
Ejemplo:
netsh interface ipv4 set address name="Ethernet 5" static 192.168.1.44
255.255.255.0 192.168.1.1
- **Poner dirección IPv4 dinámica** a una interface
netsh interface ip4 set address name="Nombre de la interfaz" source=dhcp
Ejemplo:
netsh interface ip4 set address name="Ethernet 5" source=dhcp
- **Poner DNS estático.**
netsh interface ipv4 set dns name="Nombre_Interfaz" static Dirección IP DNS
Ejemplo: netsh interface ipv4 set dns name="Ethernet 5" static 8.8.8.8
- **Poner DNS estático secundario.**
netsh interface ipv4 set dnsservers name="Nombre_Interfaz" static Dirección IP
DNS
Ejemplo: netsh interface ipv4 add dnsservers name="Ethernet 5" 8.8.4.4 index=2
- **Poner DNS por DHCP:**
netsh interface ipv4 set dnsserver name="Nombre_Interfaz" source=dhcp
netsh interface ipv4 set dnsserver name="Ethernet 5" source=dhcp
- **Habilitar/Deshabilitar una conexión de red**
netsh interface set interface name="Ethernet 5" admin=ENABLED
netsh interface set interface name="Ethernet 5" admin=DISABLED

Scripts:

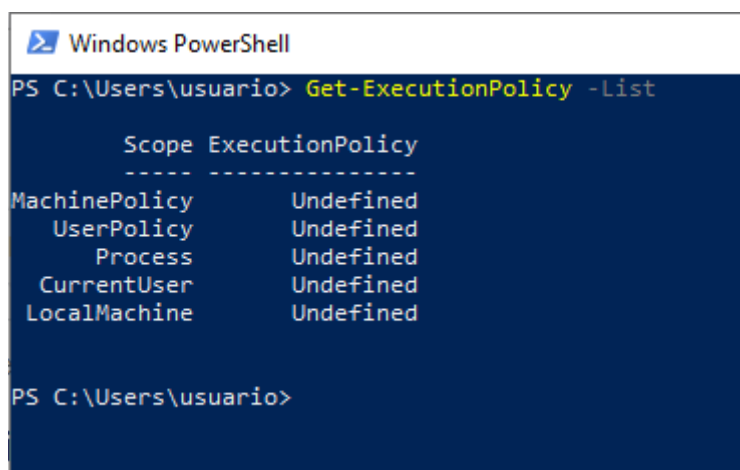
Editamos el archivo casa.ps1(La extensión debe empezar por "ps" al tratarse de un script)

casa.ps1 //configura la red cuando estoy en casa
`$ip="192.168.1.101" #variable con nombre "ip" que contiene como valor la cadena "192.168.1.101". Esto es útil por que para cambiar el valor no hay que ir cambiandolo por los comandos.`
`$masc="255.255.255.0"`
`$gw="192.168.1.1"`
`$dns1="8.8.8.8"`
`$dns2="8.8.4.4"`
`netsh interface ipv4 set address name="Ethernet 5" static $ip $masc $gw`
`netsh interface ipv4 set dnsservers name="Ethernet 5" static $dns1`
`netsh interface ipv4 add dnsservers name="Ethernet 5" $dns2 index=2`
`Get-NetIPAddress -AddressFamily IPv4 -InterfaceAlias "ethernet 5"`

dhcp.ps1 //configura la red para que tome la IP y DNS por dhcp
`netsh interface ipv4 set address name="Ethernet 5" source=dhcp`
`netsh interface ipv4 set dnsserver name="Ethernet 5" source=dhcp`
`netsh interface ipv4 show config name="Ethernet 5"`

Para que estos scrips se puedan ejecutar es necesario habilitar la ejecución de scrips en alguno de los siguientes ámbitos(**scopes**.)Para obtener los ámbitos posibles ejecutar el comando :

Get-ExecutionPolicy -List



```

Windows PowerShell
PS C:\Users\usuario> Get-ExecutionPolicy -List

Scope ExecutionPolicy
-----
MachinePolicy Undefined
UserPolicy Undefined
Process Undefined
CurrentUser Undefined
LocalMachine Undefined

PS C:\Users\usuario>

```

A estos ámbitos se le pueden aplicar distintos tipos de **políticas**:

Restricted: No carga archivos de configuracion ni ejecuta scripts. La configuracion predeterminada es "restricted".

Allsigned: Requiere que todos los scripts y archivos de configuracion esten firmados por un editor de confianza incluidos los scripts que escribamos en el equipo local.

Remotesigned: Requiere que todos los scripts y archivos de configuracion descargados de internet esten firmados por un editor de confianza.

Unrestricted: Carga todos los archivos de configuración y ejecuta todos los scripts. Si ejecutamos un script sin firmar descargado de internet, nos pedirá permiso antes de ejecutarlo.

Para cambiar los permisos del ámbito que deseemos es necesario ejecutar el siguiente comando.

Para aplicar la política Unrestricted al ámbito LocalMachine abre un PowerShell **como administrador** y ejecute el comando:

Set-ExecutionPolicy -Scope LocalMachine Unrestricted



Puede ver como cambia la política del ámbito

