Ejemplo: Examinando las configuraciones del Firewall

Wednesday, September 9, 2020 8:45 PM

Investigación Previa

Para este ejemplo vamos a explorar los comandos de PowerShell que estén relacionados con el Firewall, los cuales nos permitan obtener información relacionada con la configuración del firewall en nuestro sistema y que nos permitan ver posibles filtros que tengamos en nuestra configuración.

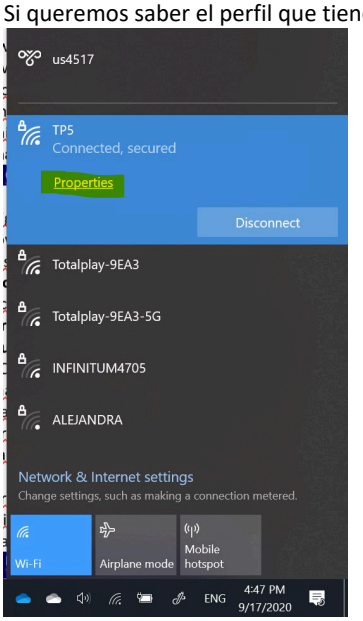
En PowerShell existe un módulo para configurar el Firewall de Windows Defender, **NetSecurity**. Para construir nuestro script lo primero que debemos hacer es ver todos los comandos disponibles, para eso usaremos el cmdlet **Get-Command**:

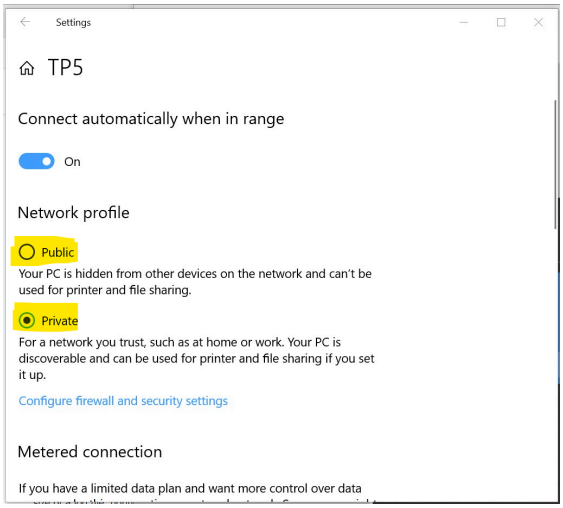
```
PS > Get-Command -Module NetSecurity
```

Investigando como funciona el Firewall de Windows, encontramos que hay 3 tipos de perfiles de red:

- **Domain** - Para equipos en el directorio activo
- **Private** - Para redes de casa o corporativas
- **Public** - Para redes públicas (como la de la FCFM)

El comando Set-NetFirewallProfile nos permite cambiar las opciones de perfil.
Si queremos saber el perfil que tiene nuestra red:





El perfil de la red se nos pregunta la primera vez que nos conectamos a ella. A modo de recomendación, todas las redes que no sean las de nuestra casa o de nuestra organización (o si no confiamos en la seguridad de nuestra organización), deben ser consideradas como redes públicas, para tener mayores restricciones de seguridad.

Si queremos activar los perfiles, podemos hacer lo siguiente:

- Para activar todos los perfiles

```
PS > Set-NetFirewallProfile -All -Enable True
```

- Para activar solo uno

```
PS > Set-NetFirewallProfile -Profile Public -Enabled True
```

Si queremos deshabilitarlo, podemos cambiar el valor del parámetro -Enabled por False.

Para ver el perfil de nuestra red actual podemos usar el comando:

```
PS > Get-NetConnectionProfile
```

Para cambiar el tipo de red:

```
PS > Set-NetConnectionProfile -Name "nombre" -NetworkCategory Public
```

Si queremos accesar las reglas de bloqueo:

```
PS> Get-NetFirewallRule -Action Block -Enabled True
```

Si quieres permitir ciertas conexión por puertos:

```
PS> New-NetFirewallRule -DisplayName 'HTTP-Inbound' -Profile @('Domain',
'Private') -Direction Inbound -Action Allow -Protocol TCP -LocalPort @('80',
'443')
```

En general, los comandos que nos permitirán administrar las reglas de Firewall son:

- New-NetFirewallRule
- Copy-NetFirewallRule
- Disable-NetFirewallRule
- Enable-NetFirewallRule
- Get-NetFirewallRule
- Remove-NetFirewallRule
- Rename-NetFirewallRule
- Set-NetFirewallRule
- Show-NetFirewallRule

Diseño de las funciones

Para los fines didácticos de este ejercicio y buscando que no tengamos problemas por usar nombres de funciones que ya existan, vamos a diseñar funciones con nombres en español.

Además, al no estar en equipos que tengan un controlador de dominios, vamos a considerar solo el uso de 2 perfiles: Público y privado.

Requerimos funciones para:

- Ver el estatus de un perfil específico en el Firewall - El parámetro de entrada será el perfil que queremos ver y podrá ser Public o Private.
- Cambiar el estatus de los perfiles - Mismo parámetro que el anterior
- Ver el perfil de nuestra red
- Cambiar nuestra red a otro tipo de perfil
- Ver las reglas de bloqueo
- Agregar regla de bloqueo de entrada para un puerto
- Eliminar regla de bloqueo

Ver-StatusPerfil

```
function Ver-StatusPerfil{
    param([Parameter(Mandatory)] [ValidateSet("Public","Private")] [string] $perfil)
    $status = Get-NetFirewallProfile -Name $perfil Write-Host "Perfil:" $perfil
    if($status.enabled){
        Write-Host "Status: Activado"
    } else{
        Write-Host "Status: Desactivado"
    }
}
```

Cambiar-StatusPerfil

```
function Cambiar-StatusPerfil{
    param([Parameter(Mandatory)] [ValidateSet("Public","Private")] [string] $perfil)
    $status = Get-NetFirewallProfile -Name $perfil
    Write-Host "Perfil:" $perfil
    if($status.enabled){
        Write-Host "Status actual: Activado"
```

```
        $opc = Read-Host -Promt "Deseas desactivarlo? [Y] Si [N] No"
        if ($opc -eq "Y"){
            Set-NetFirewallProfile -Name $perfil -Enabled False
        }
    } else{
        Write-Host "Status: Desactivado"
        $opc = Read-Host -Promt "Deseas activarlo? [Y] Si [N] No"
        if ($opc -eq "Y"){
            Write-Host "Activando perfil"
            Set-NetFirewallProfile -Name $perfil -Enabled True
        }
    }
    Ver-StatusPerfil -perfil $perfil
}
```

Ver-PerfilRedActual

```
function Ver-PerfilRedActual{
    $perfilRed = Get-NetConnectionProfile
    Write-Host "Nombre de red:" $perfilRed.Name
    Write-Host "Perfil de red:" $perfilRed.NetworkCategory
}
```

Cambiar-PerfilRedActual

```
function Cambiar-PerfilRedActual{
    $perfilRed = Get-NetConnectionProfile
    if($perfilRed.NetworkCategory -eq "Public"){
        Write-Host "El perfil actual es público"
        $opc = Read-Host -Prompt "Quieres cambiar a privado? [Y] Si [N] No"
        if($opc -eq "Y"){
            Set-NetConnectionProfile -Name $perfilRed.Name -NetworkCategory Private
            Write-Host "Perfil cambiado"
        }
    } else{
        Write-Host "El perfil actual es privado"
        $opc = Read-Host -Prompt "Quieres cambiar a público? [Y] Si [N] No"
        if($opc -eq "Y"){
            Set-NetConnectionProfile -Name $perfilRed.Name -NetworkCategory Public
            Write-Host "Perfil cambiado"
        }
    }
    Ver-PerfilRedActual
}
```

Ver-ReglasBloqueo

```
function Ver-ReglasBloqueo{
    if(Get-NetFirewallRule -Action Block -Enabled True -ErrorAction SilentlyContinue){
        Get-NetFirewallRule -Action Block -Enabled True
    } else{
        Write-Host "No hay reglas definidas aún"
    }
}
```

Agregar-ReglasBloqueo

```
function Agregar-ReglasBloqueo{
    $puerto = Read-Host -Prompt "Cuál puerto quieres bloquear?"
    New-NetFirewallRule -DisplayName "Puerto-Entrada-$puerto" -Profile "Public" -Direction
    Inbound -Action Block -Protocol TCP -LocalPort $puerto
}
```

Eliminar-ReglasBloqueo

```
function Eliminar-ReglasBloqueo{
    $reglas = Get-NetFirewallRule -Action Block -Enabled True
    Write-Host "Reglas actuales"
    foreach($regla in $reglas){
        Write-Host "Regla:" $regla.DisplayName
        Write-Host "Perfil:" $regla.Profile
        Write-Host "ID:" $regla.Name
        $opc = Read-Host -Prompt "Deseas eliminar esta regla [Y] Si [N] No"
        if($opc -eq "Y"){
            Remove-NetFirewallRule -ID $regla.name
            break
        }
    }
}
```

Módulos

- Cuántos módulos sería recomendable crear?
- Qué nombre(s) le(s) pondrías?

El script

- Requerimos crear un script con menús que haga uso de todas las funciones que tenemos
- Cómo podemos importar los Módulos? Se importan en automático?

Siguientes pasos?

Pasos faltantes se harán en el ejercicio del lunes, incluyendo agregar funciones y módulos nuevos.

Referencias

- https://www.jesusninoc.com/07/09/9-gestion-de-la-red-en-powershell/

- http://woshub.com/manage-windows-firewall-powershell/