

Guía de estudio - Optimización de PowerShell Remoting



¡Hola! Te damos la bienvenida a esta nueva guía de estudio.

¿En qué consiste esta guía?

PowerShell Remoting es una herramienta esencial para la administración remota de servidores y equipos en entornos empresariales. Sin embargo, para maximizar su eficiencia y seguridad, es fundamental realizar ajustes en la configuración de red y reforzar las políticas de seguridad.

Esta guía abordará:

- Ajustes en la configuración de red para mejorar el rendimiento de PowerShell Remoting.
- Medidas de seguridad recomendadas para proteger las conexiones remotas.
- Solución de problemas comunes en la optimización de PowerShell Remoting.

¡Vamos con todo!



Tabla de contenidos

Guía de estudio - Optimización de PowerShell Remoting	1
¿En qué consiste esta guía?	1
Tabla de contenidos	2
Ajustes en la configuración de red para mejorar el rendimiento	3
Optimización de WinRM	3
Configuración de firewall para PowerShell Remoting	3
Actividad guiada: Optimización de configuración de red	4
Medidas de seguridad para PowerShell Remoting	4
Configuración de autenticación segura	4
Restringir accesos no autorizados	4
Uso de certificados SSL para PowerShell Remoting	5
Solución de problemas comunes	5
Error: No se puede establecer conexión con el servidor remoto	5
Error: Acceso denegado	5
¡Manos a la obra! - Solución de problemas	6



¡Comencemos!

Ajustes en la configuración de red para mejorar el rendimiento

Optimización de WinRM

WinRM (Windows Remote Management) es el servicio que permite la ejecución de comandos remotos en PowerShell. Para mejorar su rendimiento, es recomendable aumentar el número máximo de conexiones concurrentes y la cantidad de datos permitidos por sesión.

- **Aumentar el límite de sesiones simultáneas:**

```
winrm set winrm/config/service '@{MaxConcurrentOperationsPerUser="100"}'
```

- **Ajustar el tamaño máximo de datos por sesión:**

```
winrm set winrm/config '@{MaxEnvelopeSizekb="8192"}'
```

- **Verificar la configuración actual de WinRM:**

```
winrm get winrm/config
```

Configuración de firewall para PowerShell Remoting

Si las conexiones remotas son lentas o no funcionan correctamente, es importante asegurarse de que los puertos adecuados estén abiertos.

- **Abrir el puerto estándar de PowerShell Remoting (HTTP - 5985):**

```
New-NetFirewallRule -Name "PSRemoting" -DisplayName "PowerShell  
Remoting" -Enabled True -Direction Inbound -Protocol TCP -Action Allow  
-LocalPort 5985
```

- **Para conexiones seguras (HTTPS - 5986):**

```
New-NetFirewallRule -Name "PSRemoting-SSL" -DisplayName "PowerShell  
Remoting Secure" -Enabled True -Direction Inbound -Protocol TCP -Action  
Allow -LocalPort 5986
```

- **Verificar reglas activas del firewall relacionadas con PowerShell Remoting:**

```
Get-NetFirewallRule | Where-Object { $_.DisplayName -like "*PowerShell*"  
}
```



Actividad guiada: Optimización de configuración de red

1. Inicializar la máquina virtual con Windows Server
2. Verificar la configuración actual de WinRM.
3. Aumentar el número de sesiones concurrentes y el tamaño máximo de datos por sesión.
4. Abrir los puertos adecuados en el firewall.
5. Comprobar su funcionamiento.

Medidas de seguridad para PowerShell Remoting

Configuración de autenticación segura

Para evitar accesos no autorizados, se recomienda habilitar métodos de autenticación segura en PowerShell Remoting.

- **Configurar autenticación basada en Kerberos:**

```
winrm set winrm/config/service '@{AllowUnencrypted="false"}'
```

- **Deshabilitar autenticación básica (solo permitir autenticación segura):**

```
winrm set winrm/config/service/Auth '@{Basic="false"}'
```

Restringir accesos no autorizados

Para mayor control, se recomienda permitir sólo conexiones desde direcciones IP específicas.

- **Permitir conexiones sólo desde una IP específica:**

```
Set-Item WSMan:\localhost\Client\TrustedHosts -Value "192.168.1.100"
```

- **Para permitir varias IPs:**

```
Set-Item WSMan:\localhost\Client\TrustedHosts -Value  
"192.168.1.100,192.168.1.101"
```

- **Verificar la configuración de hosts confiables:**

```
Get-Item WSMAN:\localhost\Client\TrustedHosts
```

Uso de certificados SSL para PowerShell Remoting

Para entornos donde la seguridad es crítica, se recomienda el uso de conexiones cifradas mediante SSL.

- **Crear un listener HTTPS para PowerShell Remoting:**

```
winrm create winrm/config/Listener?Address=*&Transport=HTTPS  
'@{Port="5986";CertificateThumbprint="ABC123DEF456"}'
```

- **Verificar la configuración de listeners en WinRM:**

```
winrm enumerate winrm/config/listener
```

Solución de problemas comunes

Error: No se puede establecer conexión con el servidor remoto

Si PowerShell Remoting no responde, verificar:

- Que WinRM está en ejecución:

```
Get-Service -Name WinRM
```

- Que el puerto 5985 o 5986 está abierto en el firewall.
- Que el servidor remoto está en la lista de hosts confiables:

```
Get-Item WSMAN:\localhost\Client\TrustedHosts
```

Error: Acceso denegado

Si se recibe un mensaje de acceso denegado:

- Verificar credenciales de usuario:

```
Test-WSMan -ComputerName ServidorRemoto -Credential Administrador
```

- Agregar el usuario a los permisos de PowerShell Remoting:

```
Set-PSSessionConfiguration -Name Microsoft.PowerShell  
-ShowSecurityDescriptorUI
```

¡Manos a la obra! - Solución de problemas

1. Inicializar la máquina virtual con Windows Server
2. Identificar y solucionar un problema de conexión con Test-WSMan.
3. Verificar y modificar los permisos de usuario en PowerShell Remoting.
4. Comprobar la configuración de firewall para garantizar que PowerShell Remoting funcione correctamente.
5. Verificar la conexión remota desde Powershell posicionado en un cliente desktop

Reflexiona:

- ¿Cómo afecta el tamaño del MaxEnvelopeSizekb al rendimiento de PowerShell Remoting?
- ¿Por qué es recomendable usar autenticación basada en Kerberos en entornos empresariales?
- ¿Cuáles son las ventajas de utilizar conexiones seguras con SSL en PowerShell Remoting?
- ¿Cómo se pueden mitigar los riesgos de seguridad al habilitar la administración remota?
- ¿Qué problemas pueden surgir al configurar reglas de firewall para PowerShell Remoting?



La optimización de PowerShell Remoting permite mejorar su rendimiento y seguridad en entornos empresariales. A través de ajustes en WinRM, firewall y autenticación, los administradores pueden garantizar conexiones eficientes y seguras a servidores remotos.