

Guía de estudio - Conexión y ejecución remota en PowerShell



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

¿En qué consiste esta guía?

PowerShell permite la administración remota de servidores y equipos a través de PowerShell Remoting, lo que facilita la ejecución de comandos y la gestión de sistemas sin necesidad de acceso físico. Dos de los cmdlets más utilizados para la ejecución remota son Enter-PSSession y Invoke-Command.

Esta guía cubre:

- Uso de Enter-PSSession para acceso interactivo.
- Uso de Invoke-Command para ejecutar comandos en servidores remotos.
- Configuraciones avanzadas y solución de problemas.

¡Vamos con todo!



Tabla de contenidos

| | |
|---|----------|
| Guía de estudio - Conexión y ejecución remota en PowerShell | 1 |
| ¿En qué consiste esta guía? | 1 |
| Tabla de contenidos | 2 |
| Uso de Enter-PSSession para acceso interactivo | 3 |
| Requisitos previos | 3 |
| Iniciar una sesión interactiva | 3 |
| Ejecutar comandos en la sesión interactiva | 3 |
| Actividad guiada: Configuración y uso de Enter-PSSession | 4 |
| Uso de Invoke-Command para la ejecución de comandos en servidores remotos | 4 |
| Ejecutar un comando en un servidor remoto | 4 |
| Ejecutar múltiples comandos | 4 |
| Ejecutar comandos en múltiples servidores | 4 |
| Uso de credenciales alternativas | 5 |
| Actividad guiada: Uso de Invoke-Command | 5 |
| Configuración avanzada y solución de problemas | 5 |
| Verificar la configuración de PowerShell Remoting | 5 |
| Configuración de Firewall | 5 |
| Errores comunes y soluciones | 6 |
| ¡Manos a la obra! - Configuración y solución de problemas | 6 |



¡Comencemos!

Uso de Enter-PSSession para acceso interactivo

El cmdlet Enter-PSSession permite abrir una sesión interactiva con un servidor remoto, donde se pueden ejecutar comandos como si se estuviera trabajando localmente en ese sistema.

Requisitos previos

Antes de usar Enter-PSSession, es necesario:

- Tener habilitado PowerShell Remoting en el servidor remoto (Enable-PSRemoting -Force).
- Asegurarse de que el servicio WinRM está activo (Start-Service WinRM).
- Configurar el servidor remoto como host confiable si no está en el mismo dominio:

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

Iniciar una sesión interactiva

Para conectarse a un servidor remoto:

```
Enter-PSSession -ComputerName ServidorRemoto -Credential Administrador
```

Este comando solicitará credenciales de inicio de sesión y abrirá una sesión interactiva en el servidor.

Ejecutar comandos en la sesión interactiva

Una vez dentro de la sesión, los comandos se ejecutan como si estuviéramos en la consola local:

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

Para salir de la sesión y volver a la máquina local:

```
Exit-PSSession
```



Actividad guiada: Configuración y uso de Enter-PSSession

1. Inicializar la máquina virtual Windows Server con dominio creado y otra máquina Windows Server Core unida al dominio.
2. Habilitar PowerShell Remoting.
3. En la máquina tutora, Agregar el servidor remoto a la lista de hosts confiables.
4. Iniciar una sesión interactiva con Enter-PSSession.
5. Ejecutar comandos en la sesión remota y salir con Exit-PSSession.

Uso de Invoke-Command para la ejecución de comandos en servidores remotos

El cmdlet Invoke-Command permite ejecutar comandos en uno o varios servidores remotos simultáneamente, sin necesidad de abrir una sesión interactiva.

Ejecutar un comando en un servidor remoto

```
Invoke-Command -ComputerName ServidorRemoto -ScriptBlock { Get-Service WinRM }
```

Este comando recupera el estado del servicio WinRM en el servidor remoto.

Ejecutar múltiples comandos

Si se necesita ejecutar varios comandos, se pueden agrupar dentro de ScriptBlock:

```
Invoke-Command -ComputerName ServidorRemoto -ScriptBlock {  
    Get-Date  
    Get-Process  
    Get-Service -Name W32Time  
}
```

Ejecutar comandos en múltiples servidores

PowerShell permite administrar múltiples servidores al mismo tiempo:

```
Invoke-Command -ComputerName Servidor1, Servidor2, Servidor3  
-ScriptBlock { Get-Service WinRM }
```

Uso de credenciales alternativas

Si se requiere especificar credenciales diferentes, se usa:

```
$Credenciales = Get-Credential  
Invoke-Command -ComputerName ServidorRemoto -Credential $Credenciales  
-ScriptBlock { Get-EventLog -LogName System -Newest 10 }
```



Actividad guiada: Uso de Invoke-Command

1. Ejecutar un comando en un servidor remoto con Invoke-Command.
2. Ejecutar múltiples comandos en la misma sesión.
3. Administrar servicios en múltiples servidores simultáneamente.

Configuración avanzada y solución de problemas

Verificar la configuración de PowerShell Remoting

Si hay problemas de conexión, verificar si PSRemoting está habilitado:

```
Get-PSSessionConfiguration
```

Para habilitarlo si es necesario:

```
Enable-PSRemoting -Force
```

Configuración de Firewall

Si el firewall bloquea las conexiones remotas, agregar una regla para permitir PowerShell Remoting:

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

Errores comunes y soluciones

| Error | Solución |
|---|---|
| Acceso denegado al intentar conectar. | Verificar permisos de usuario y credenciales. |
| El servidor no responde. | Asegurar que WinRM está en ejecución (Start-Service WinRM). |
| No se puede establecer una sesión remota. | Agregar el servidor a TrustedHosts. |
| Firewall bloquea la conexión. | Configurar una regla de firewall para WinRM. |

Tabla 01. Errores comunes y soluciones.

Fuente: ADL.



¡Manos a la obra! - Configuración y solución de problemas

1. Verificar si PowerShell Remoting está habilitado en un servidor remoto.
2. Configurar el firewall para permitir conexiones remotas.
3. Solucionar problemas de conexión mediante Get-PSSessionConfiguration.

Reflexiona:

- ¿Cuál es la diferencia entre Enter-PSSession e Invoke-Command?
- ¿Cuándo es más conveniente usar Enter-PSSession en lugar de Invoke-Command?
- ¿Cómo se pueden mejorar la seguridad y el control de acceso en sesiones remotas de PowerShell?
- ¿Qué errores comunes pueden surgir en PowerShell Remoting y cómo solucionarlos?
- ¿Por qué es importante configurar adecuadamente el firewall al habilitar PSRemoting?



PowerShell Remoting es una herramienta esencial para la administración remota de servidores. Con Enter-PSSession, los administradores pueden trabajar en servidores remotos de forma interactiva, mientras que Invoke-Command permite ejecutar scripts y comandos en múltiples servidores simultáneamente.