



python



PowerShell

FPDF Library  
PDF generator

*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***PowerShell Get-Help on command 'Enable-PSRemoting'***

***PS C:\Users\wahid> Get-Help Enable-PSRemoting***

#### NAME

Enable-PSRemoting

#### SYNOPSIS

Configures the computer to receive remote commands.

#### SYNTAX

Enable-PSRemoting [-Force] [-SkipNetworkProfileCheck] [-Confirm] [-WhatIf]  
[<CommonParameters>]

#### DESCRIPTION

The `Enable-PSRemoting` cmdlet configures the computer to receive PowerShell remote commands that are sent by using the WS-Management technology.

PowerShell remoting is enabled by default on Windows Server 2012. You can use `Enable-PSRemoting` to enable PowerShell remoting on other supported versions of Windows and to re-enable remoting on Windows Server 2012 if it becomes disabled.

You have to run this command only one time on each computer that will receive commands. You do not have to run it on computers that only send commands. Because the configuration starts listeners, it is prudent to run it only where it is needed.

Beginning in PowerShell 3.0, the ``Enable-PSRemoting`` cmdlet can enable PowerShell remoting on client versions of Windows when the computer is on a public network. For more information, see the description of the `SkipNetworkProfileCheck` parameter.

The ``Enable-PSRemoting`` cmdlet performs the following operations:

- Runs the `Set-WSManQuickConfig`

(`./Microsoft.WSMan.Management/Set-WSManQuickConfig.md`)cmdlet, which performs the following tasks:

- Starts the WinRM service.
- Sets the startup type on the WinRM service to Automatic.
- Creates a listener to accept requests on any IP address.
- Enables a firewall exception for WS-Management communications.
- Registers the `Microsoft.PowerShell` and `Microsoft.PowerShell.Workflow` session configurations, if they are not already registered.
- Registers the `Microsoft.PowerShell32` session configuration on 64-bit computers, if it is not already registered.
- Enables all session configurations.
- Changes the security descriptor of all session configurations to allow remote access.
- Restarts the WinRM service to make the preceding changes effective.

To run this cmdlet on the Windows platform, start PowerShell by using the Run as administrator option. This does not apply to Linux or MacOS versions of PowerShell.

> [!CAUTION] > On systems that have both PowerShell 3.0 and PowerShell 2.0, do not use > PowerShell 2.0 to run the ``Enable-PSRemoting`` and ``Disable-PSRemoting`` cmdlets. The commands > might appear to succeed, but the remoting is not configured correctly. Remote commands and later > attempts to

enable and disable remoting, are likely to fail.

## PARAMETERS

`-Force <System.Management.Automation.SwitchParameter>`

Forces the command to run without asking for user confirmation.

`-SkipNetworkProfileCheck <System.Management.Automation.SwitchParameter>`

Indicates that this cmdlet enables remoting on client versions of the Windows operating system when the computer is on a public network. This parameter enables a firewall rule for public networks that allows remote access only from computers in the same local subnet.

This parameter does not affect server versions of the Windows operating system, which, by default, have a local subnet firewall rule for public networks. If the local subnet firewall rule is disabled on a server version, ``Enable-PSRemoting`` re-enables it, regardless of the value of this parameter.

To remove the local subnet restriction and enable remote access from all locations on public networks, use the ``Set-NetFirewallRule`` cmdlet in the NetSecurity module.

This parameter was introduced in PowerShell 3.0.

`-Confirm <System.Management.Automation.SwitchParameter>`

Prompts you for confirmation before running the cmdlet.

`-WhatIf <System.Management.Automation.SwitchParameter>`

Shows what would happen if the cmdlet runs. The cmdlet is not run.

`<CommonParameters>`

This cmdlet supports the common parameters: Verbose, Debug,

ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see about\_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

-- Example 1: Configure a computer to receive remote commands --

```
Enable-PSRemoting
```

Example 2: Configure a computer to receive remote commands without a confirmation prompt

```
Enable-PSRemoting -Force
```

----- Example 3: Allow remote access on clients -----

```
Get-NetFirewallRule -Name 'WINRM*' | Select-Object -Property Name
```

```
Name
```

```
----
```

```
WINRM-HTTP-In-TCP-NoScope
```

```
WINRM-HTTP-In-TCP
```

```
WINRM-HTTP-Compat-In-TCP-NoScope
```

```
WINRM-HTTP-Compat-In-TCP
```

```
Enable-PSRemoting -SkipNetworkProfileCheck -Force
```

```
Set-NetFirewallRule -Name 'WINRM-HTTP-In-TCP' -RemoteAddress Any
```

By default, `Enable-PSRemoting` creates network rules that allow remote access from private and domain networks. The command uses the SkipNetworkProfileCheck parameter to allow remote access from public networks in the same local subnet. The command specifies the Force parameter to suppress confirmation

messages.

The `SkipNetworkProfileCheck` parameter does not affect server versions of the Windows operating system, which allow remote access from public networks in the same local subnet by default.

The `Set-NetFirewallRule` cmdlet in the NetSecurity module adds a firewall rule that allows remote access from public networks from any remote location. This includes locations in different subnets.`

> [!NOTE] > The name of the firewall rule can be different depending on the version of Windows. Use the > `Get-NetFirewallRule` cmdlet to list the names of the rules on your system.`

#### REMARKS

To see the examples, type: `"get-help Enable-PSRemoting -examples"`.

For more information, type: `"get-help Enable-PSRemoting -detailed"`.

For technical information, type: `"get-help Enable-PSRemoting -full"`.

For online help, type: `"get-help Enable-PSRemoting -online"`