



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 'Add-NetEventWFPCaptureProvider'**

**PS C:\Users\wahid> Get-Help Add-NetEventWFPCaptureProvider**

#### NAME

Add-NetEventWFPCaptureProvider

#### SYNOPSIS

Creates a WFP capture provider.

#### SYNTAX

```
Add-NetEventWFPCaptureProvider [-SessionName] <String> [[-Level] <Byte>]
[[-MatchAnyKeyword] <UInt64>] [[-MatchAllKeyword] <UInt64>]
[[-CaptureLayerSet] {IPv4Inbound | IPv4Outbound | IPv6Inbound | IPv6Outbound}]
[[-IPAddresses] <String[]>] [[-TCPPorts] <UInt16[]>] [[-UDPPorts] <UInt16[]>]
[-AsJob] [-CimSession <CimSession[]>] [-Confirm] [-ThrottleLimit <Int32>]
[-WhatIf] [<CommonParameters>]
```

#### DESCRIPTION

The Add-NetEventWFPCaptureProvider cmdlet creates a Windows Firewall Platform (WFP) capture provider for network events. The WFP captures events directly from the network stack. You can capture traffic in network tunnels and from the loopback adapter. Unlike a NetEventPacketCapture provider, the

NetEventWFPCaptureProvider captures network traffic above the IP layer.

A computer typically supports only one packet capture provider. If there is an existing provider on the current computer, remove it before you run this cmdlet.

## PARAMETERS

**-AsJob** [<SwitchParameter>]

Runs the cmdlet as a background job. Use this parameter to run commands that take a long time to complete.

**-CaptureLayerSet** <WFPCaptureSet>

Specifies a WFP capture set, which designates the layers and directions to filter. The acceptable values for this parameter are:

- IPv4Inbound

- IPv4Outbound

- IPv6Inbound

- IPv6Outbound

You can locally OR the direction and IP layer pairs together. For instance, you could capture incoming loopback traffic from IPv6 to avoid seeing duplicate traffic received by the loopback interface.

**-CimSession** <CimSession[]>

Runs the cmdlet in a remote session or on a remote computer. Enter a computer name or a session object, such as the output of a New-CimSession

(<https://go.microsoft.com/fwlink/p/?LinkId=227967>) or

[Get-CimSession](<https://go.microsoft.com/fwlink/p/?LinkId=227966>)cmdlet.

The default is the current session on the local computer.

-Confirm [<SwitchParameter>]

Prompts you for confirmation before running the cmdlet.

-IPAddresses <String[]>

Specifies an array of IP addresses. The provider filters for and logs network traffic that matches the IP addresses that this parameter specifies. The provider joins multiple addresses by using logical ORs.

-Level <Byte>

Specifies the Event Tracing for Windows (ETW) event error levels that NetEventWFPCaptureProvider returns. Use a level of detail specifier as a filter the type of error events that are logged. The default value for this parameter is 0x4, for informational events. The acceptable values for this parameter are:

- 0x5. Verbose - 0x4. Informational - 0x3. Warning - 0x2. Error - 0x1.

Critical - 0x0. LogAlways

The provider must log the event if the value of the event is less than or equal to the value of this parameter. Lower level events up to and including the specified level are logged.

-MatchAllKeyword <UInt64>

Specifies a keyword bitmask that restricts the events that the provider logs.

-MatchAnyKeyword <UInt64>

Specifies keywords as a set of hexadecimal values. Keywords are flags that you can combine to generate hexadecimal values that enable the provider to

write one or more events for which it is instrumented, if a match is found. Use a set of hexadecimal values for keywords instead of the keyword names, and apply a filter to write ETW events for keyword matches. For more information, see `EnableTraceEx2` function ([https://msdn.microsoft.com/en-us/library/windows/desktop/dd392305\(v=vs.85\)\)](https://msdn.microsoft.com/en-us/library/windows/desktop/dd392305(v=vs.85))) in the Microsoft Developer Network library.

**-SessionName <String>**

Specifies the name of the session that is associated with the `NetEventWFPCaptureProvider`. This parameter has the same value as the `Name` parameter for the `New-NetEventSession` cmdlet.

**-TCPPorts <UInt16[]>**

Specifies an array of TCP ports. The provider filters for and logs network traffic that matches the ports that this parameter specifies. The provider joins multiple port numbers with logical ORs.

**-ThrottleLimit <Int32>**

Specifies the maximum number of concurrent operations that can be established to run the cmdlet. If this parameter is omitted or a value of `0` is entered, then Windows PowerShell calculates an optimum throttle limit for the cmdlet based on the number of CIM cmdlets that are running on the computer. The throttle limit applies only to the current cmdlet, not to the session or to the computer.

**-UDPPorts <UInt16[]>**

Specifies an array of UDP ports. The provider filters for and logs network traffic that matches the ports that this parameter specifies. The provider joins multiple port numbers with logical ORs.

**-WhatIf [<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 \(https://go.microsoft.com/fwlink/?LinkID=113216\)](https://go.microsoft.com/fwlink/?LinkID=113216).

----- Example 1: Create a WFP capture provider -----

```
PS C:\>New-NetEventSession -Name "WFPCapture" -CaptureMode RealtimeLocal
-LocalFilePath "C:\users\DavidChew\Documents\wfpdata.etl"
PS C:\> Add-NetEventWFPCaptureProvider -SessionName "WFPCapture"
PS C:\> Start-NetEventSession -Name "WFPCapture"
```

The first command creates a network event session by using the `New-NetEventSession` cmdlet. The command also assigns the name `WFPCapture` to the session.

The second command uses the current cmdlet to create a WFP capture provider for the session named `WFPCapture`.

The final command starts the event tracing session named `WFPCapture`.

## REMARKS

To see the examples, type: `"get-help Add-NetEventWFPCaptureProvider -examples"`.

For more information, type: `"get-help Add-NetEventWFPCaptureProvider -detailed"`.

For technical information, type: `"get-help Add-NetEventWFPCaptureProvider -full"`.

For online help, type: `"get-help Add-NetEventWFPCaptureProvider -online"`