

Microsoft netsh trace

Basic

Start the trace:

```
netsh trace start capture=yes Ethernet.Type=IPv4
```

Stop the trace:

```
netsh trace stop
```

Files will be created in %AppData%\Local\Temp\NetTraces

View ETL in Microsoft Network Monitor 3.4

If you load the ETL in Microsoft Network Monitor 3.4, you will see the following error in the packet display:

MicrosoftWindowsNDISPacketCapture: Windows stub parser: Requires full Common parsers. See the "How Do I Change Parser Set Options(Version 3.3 or before) or Configure Parser Profile (Version 3.4)" help topic for tips on loading these parser sets.

To fix this you need to change the active Parser Profile. Click the Parser Profiles button on the top right of the screen, click Network Monitor Profiles, and then click Windows to set it as the active profile.

Details

```
C:\>netsh trace start help
```

start

Starts tracing.

```
Usage: trace start [sessionname=<sessionname>]
        [[scenario=]<scenario1,scenario2>]
        [[globalKeywords=]keywords] [[globalLevel=]level]
        [[capture=]yes|no] [[capturetype=]physical|vmswitch|both]
        [[report=]yes|no|disabled] [[persistent=]yes|no]
        [[traceFile=]path\filename] [[maxSize=]filemaxsize]
        [[fileMode=]single|circular|append] [[overwrite=]yes|no]
        [[correlation=]yes|no|disabled] [capturefilters]
        [[provider=]providerIdOrName] [[keywords=]keywordMaskOrSet]
        [[level=]level] [bufferSize=<bufferSize>]
        [[[provider=]provider2IdOrName] [[providerFilter=]yes|no]]
        [[keywords=]keyword2MaskOrSet] [[perfMerge=]yes|no]
        [[level=]level2] ...
```

Defaults:

```
capture=no (specifies whether packet capture is enabled
            in addition to trace events)
capturetype=physical (specifies whether packet capture needs to be
                     enabled for physical network adapters only, virtual switch
                     only, or both physical network adapters and virtual switch)
report=no (specifies whether a complementing report will be generated
          along with the trace file)
persistent=no (specifies whether the tracing session continues
              across reboots, and is on until netsh trace stop is issued)
maxSize=250 MB (specifies the maximum trace file size, 0=no maximum)
bufferSize=512 (specifies trace buffer size in KB, min 4, max 16384)
fileMode=circular
overwrite=yes (specifies whether an existing trace output file will
              be overwritten)
correlation=disabled (specifies whether related events will be
                     correlated and grouped together)
perfMerge=yes (specifies whether performance metadata is merged
              into trace)
```

`traceFile=%LOCALAPPDATA%\Temp\NetTraces\[sessionname]NetTrace.etl`
(specifies location of the output file)
`providerFilter=no` (specifies whether provider filter is enabled)
`sessionname=''` (specifies a name for the trace session so that
simultaneous traces can be collected.

Provider keywords default to all and level to 255 unless otherwise specified.

For example:

```
netsh trace start scenario=InternetClient capture=yes
```

Starts tracing for the InternetClient scenario and dependent providers
with packet capture enabled for physical network adapters only.
Tracing will stop when the "netsh trace stop" command is issued
or when the system reboots.
Default location and name will be used for the output file. If an old
file exists, it will be overwritten.

```
netsh trace start provider=microsoft-windows-wlan-autoconfig  
keywords=state,ut:authentication
```

Starts tracing for the microsoft-windows-wlan-autoconfig provider
Tracing will stop when the "netsh trace stop" command is issued
or when the system reboots.
Default location and name will be used for the output file. If an old
file exists, it will be overwritten.
Only events with keyword 'state' or 'ut:authentication' will be logged.

`netsh trace show provider` command can be used to display
supported keywords and levels.

Capture Filters:

Capture filters are only supported when capture is explicitly
enabled with `capture=yes`. Use '`netsh trace show CaptureFilterHelp`'
to display a list of supported capture filters and their usage.

Provider Filters:

Provider filters are supported by multiple providers and are enabled
with `providerFilter=Yes` after every provider.

Use 'netsh trace show ProviderFilterHelp' to display a list of supported provider filters for each provider and their usage.

capturefilterhelp

```
C:\>netsh trace show capturefilterhelp
```

Capture Filters:

Capture filters are only supported when capture is explicitly enabled with capture=yes. Supported capture filters are:

CaptureInterface=<interface name or GUID>

Enables packet capture for the specified interface name or GUID. Use 'netsh trace show interfaces' to list available interfaces.

e.g. CaptureInterface={716A7812-4AEE-4545-9D00-C10EFD223551}

e.g. CaptureInterface={!{716A7812-4AEE-4545-9D00-C10EFD223551}}

e.g. CaptureInterface="Local Area Connection"

Ethernet.Address=<MAC address>

Matches the specified filter against both source and destination MAC addresses.

e.g. Ethernet.Address=00-0D-56-1F-73-64

Ethernet.SourceAddress=<MAC address>

Matches the specified filter against source MAC addresses.

e.g. Ethernet.SourceAddress=00-0D-56-1F-73-64

Ethernet.DestinationAddress=<MAC address>

Matches the specified filter against destination MAC addresses.

e.g. Ethernet.DestinationAddress=00-0D-56-1F-73-64

Ethernet.Type=<ethertype>

Matches the specified filter against the MAC ethertype.

e.g. Ethernet.Type=IPv4

e.g. Ethernet.Type=NOT(0x86DD)

e.g. Ethernet.Type=(IPv4,IPv6)

Wifi.Type=<Management|Data>

Matches the specified filter against the Wifi type. Allowed values are 'Management' and 'Data'. If not specified, the Wifi.Type filter is not applied.

Note: This capture filter does not support ranges, lists or negation.

e.g. Wifi.Type=Management

Protocol=<protocol>

Matches the specified filter against the IP protocol.

e.g. Protocol=6

e.g. Protocol=!(TCP,UDP)

e.g. Protocol=(4-10)

IPv4.Address=<IPv4 address>

Matches the specified filter against both source and destination IPv4 addresses.

e.g. IPv4.Address=157.59.136.1

e.g. IPv4.Address=!(157.59.136.1)

e.g. IPv4.Address=(157.59.136.1,157.59.136.11)

IPv4.SourceAddress=<IPv4 address>

Matches the specified filter against source IPv4 addresses.

e.g. IPv4.SourceAddress=157.59.136.1

IPv4.DestinationAddress=<IPv4 address>

Matches the specified filter against destination IPv4 addresses.

e.g. IPv4.DestinationAddress=157.59.136.1

IPv6.Address=<IPv6 address>

Matches the specified filter against both source and destination IPv6 addresses.

e.g. IPv6.Address=fe80::5038:3c4:35de:f4c3\%8

e.g. IPv6.Address=!(fe80::5038:3c4:35de:f4c3\%8)

IPv6.SourceAddress=<IPv6 address>

Matches the specified filter against source IPv6 addresses.

e.g. IPv6.SourceAddress=fe80::5038:3c4:35de:f4c3\%8

IPv6.DestinationAddress=<IPv6 address>

Matches the specified filter against destination IPv6 addresses.

e.g. `IPv6.DestinationAddress=fe80::5038:3c4:35de:f4c3\%8`

`CustomMac=<type(offset,value)>`

Matches the specified filter against the value at the specified offset starting with the MAC header.

Note: This capture filter does not support ranges, lists or negation.

e.g. `CustomMac=UINT8(0x1,0x23)`

e.g. `CustomMac=ASCIISTRING(3,test)`

e.g. `CustomMac=UNICODESTRING(2,test)`

`CustomIp=<type(offset,value)>`

Matches the specified filter against the value at the specified offset starting with the IP header.

Note: This capture filter does not support ranges, lists or negation.

e.g. `CustomIp=UINT16(4,0x3201)`

e.g. `CustomIp=UINT32(0x2,18932)`

`CaptureMultiLayer=<yes|no>`

Enables multi-layer packet capture.

Note: This capture filter does not support ranges, lists or negation.

`PacketTruncateBytes=<value>`

Captures only the the specified number of bytes of each packet.

Note: This capture filter does not support ranges, lists or negation.

e.g. `PacketTruncateBytes=40`

Note:

Multiple filters may be used together. However the same filter may not be repeated.

e.g. `'netsh trace start capture=yes Ethernet.Type=IPv4
IPv4.Address=157.59.136.1'`

Filters need to be explicitly stated when required. If a filter is not specified, it is treated as "don't-care".

e.g. `'netsh trace start capture=yes IPv4.SourceAddress=157.59.136.1'`

This will capture IPv4 packets only from 157.59.136.1, and it will also capture packets with non-IPv4 Ethernet Types, since the Ethernet.Type filter is not explicitly specified.

e.g. `'netsh trace start capture=yes IPv4.SourceAddress=157.59.136.1
Ethernet.Type=IPv4'`

This will capture IPv4 packets only from 157.59.136.1. Packets with other Ethernet Types will be discarded since an explicit filter has been specified.

Capture filters support ranges, lists and negation (unless stated otherwise).

e.g. Range: 'netsh trace start capture=yes Ethernet.Type=IPv4 Protocol=(4-10)'

This will capture IPv4 packets with protocols between 4 and 10 inclusive.

e.g. List: 'netsh trace start capture=yes Ethernet.Type=(IPv4,IPv6)'

This will capture only IPv4 and IPv6 packets.

e.g. Negation: 'netsh trace start capture=yes Ethernet.Type=!IPv4'

This will capture all non-IPv4 packets.

Negation may be combined with lists in some cases.

e.g. 'netsh trace start capture=yes Ethernet.Type=!(IPv4,IPv6)'

This will capture all non-IPv4 and non-IPv6 packets.

'NOT' can be used instead of '!' to indicate negation. This requires parentheses to be present around the values to be negated.

e.g. 'netsh trace start capture=yes Ethernet.Type=NOT(IPv4)'

Scenarios

```
C:\>netsh trace show scenarios
```

Available scenarios (24):

```
-----
AddressAcquisition      : Troubleshoot address acquisition related issues
AddressAcquisitionServer : Troubleshoot address acquisition server related issues
DirectAccess            : Troubleshoot DirectAccess related issues
DirectAccessServer      :
FileSharing              : Troubleshoot common file and printer sharing problems
ICS                      : Troubleshoot internet connection sharing related issues
InternetClient           : Troubleshoot web connectivity issues
InternetServer           : Troubleshoot server-side web connectivity issues
L2SEC                    : Troubleshoot layer 2 authentication related issues
LAN                      : Troubleshoot wired LAN related issues
Layer2                   : Troubleshoot layer 2 connectivity related issues
```

MBN	: Troubleshoot mobile broadband related issues
NDIS	: Troubleshoot network adapter related issues
NetConnection	: Troubleshoot network connection related issues
NetworkSnapshot	: Collect the current network state of the system
P2P-Grouping	: Troubleshoot Peer-to-Peer Grouping related issues
P2P-PNRP	: Troubleshoot Peer Name Resolution Protocol (PNRP) related issues
RemoteAssistance	: Troubleshoot Windows Remote Assistance related issues
Virtualization	:
VPNServer	: Troubleshoot VPN related issues
WCN	: Troubleshoot Windows Connect Now related issues
WFP-IPsec	: Troubleshoot Windows Filtering Platform and IPsec related issues
WLAN	: Troubleshoot wireless LAN related issues
XboxMultiplayer	: Troubleshoot Xbox Live Multiplayer connectivity-related issues

-end

Revision #2

Created 21 December 2023 21:39:45 by bluecrow76

Updated 21 December 2023 22:25:45 by bluecrow76