

iBtTraceTool commands:

tracestart <scenario> <options> / < single options for scenario based logging >
tracestop

triage

GetLinkInfo

piconet

otpver

<scenario> - boot

<single options for scenario based logging > - install, bsod, sx, yb, bluetooth

<options> - ibtusb, ibtuart, ixbtusb, hci, acpi, moto, usbbl, msusb, ibtsiva, ibtver, msft, usb3win7,
iusb, firmware, ibtbp, hid, ibtpci

Description:

ibtusb	- Enables tracing for Intel Bluetooth USB Filter driver
ibtuart	- Enables tracing for Intel Bluetooth UART driver
ixbtusb	- Enables tracing for ixbtusb driver
hci	- Enables hci tracing only to use with btetlparse
acpi	- Enables ACPI tracing on Windows 10
usbbl	- Enables tracing for bootloader driver
msusb	- Enables tracing for Microsoft USB stack
msuart	- Enables tracing for Microsoft UART stack
ibtsiva	- Enables tracing for iBTSiva Service (ibtsiva)
ibtver	- Enables tracing for the ibtVer application
msft	- Enable tracing for Microsoft BT stack (for opening support case)
firmware	- Enables tracing for Intel Bluetooth Firmware
ibtbpl	- Enables tracing for Intel ibtbpl.exe (iBT Binary Patch Loader application)
hid	- Enables tracing for Microsoft HID stack
ibtpci	- Enables tracing for Intel Bluetooth PCI driver
ibtrstd	- Enables tracing for Intel Bluetooth Reset Device driver
sercx2	- Enables tracing for sercx2 driver
lpssuart	- Enables tracing for LPSS UART driver

uart driver - Enables tracing for Intel Bluetooth UART driver, Microsoft UART stack, and LPSS UART

Win7 Specific options:

usb3win7 - Enables tracing for Intel USB3 Controller driver (Windows 7)

iusb - Enables tracing for Intel USB3 Controller driver and
Intel Bluetooth USB Filter driver (Windows 7- interlaced logs)

moto - Enables tracing for Motorola Bluetooth stack only

Log files location:

Log files are stored under "Intel_Bluetooth_Traces" folder on the Desktop if not specified otherwise.

To change the logs folder, provide the new path using the tracestart command as following:

```
tracestart <trace scenario> <trace option> -path <New path to store logs>
```

LogSize parameter:

The default log files size is set to 128MB. In case the user wants to get larger etl file, logSize parameter can be passed with the desired value

LogSize can be applied to msft, ibtusb, ibtpci, ibtuart, msusb options

nocirc option:

The default mode is set to circular. That means that once the file reaches the maximum size, the oldest events are replaced with the incoming events. Setting the "nocirc" disables the circular file mode and stops logging once the file reaches the maximum size.

See examples below!

Usage Examples:

`tracestart`

Enables default tracing for normal scenario

`tracestart -path D:\MyLogs`

Enables default tracing for normal scenario but stores the logs in the new path

"D:\MyLogs" instead of the default path.

`tracestart -nocirc`

Disables default file mode. Ibttracetool will stop logging once the file reaches the maximum size.

`tracestart boot iusb -path D:\MyLogs`

Enables complete tracing for boot scenario and usb options,

and stores the logs in the new path "D:\MyLogs" instead of the default path.

Requires shutdown /p or restart

`tracestart boot ibtusb`

Enables boot scenario for ibtusb

Requires shutdown /p or restart

`tracestart sx OR tracestart yb`

Enables logging for sx or yb scenarios

Starts Intel WRT Reporting Tool to collect additional firmware logs only on Win10 systems

tracestart install

Enables logging for install BT driver scenario

tracestart bluetooth

Enables logging for bluetooth related scenarios

tracestart ibtuart

Enables tracing for ibtuart option

tracestart ibtusb

Enables tracing for ibtusb option

tracestart ibtusb -logSize 200

Enables tracing for ibtusb option and the size for the etl file will be around 200MB

tracestart ibtuart firmware

Enables ibtuart tracing and default firmware tracing

tracestart firmware (oi=ahb, at=yes, om=dc_new, tsp=hci)

Enables custom firmware tracing

Starts Intel WRT Reporting Tool to collect default FW traces

tracestart firmware (tsp=llc,tsp=hci)

Enables default firmware tracing with two specified trace selection patterns

Starts Intel WRT Reporting Tool to collect default FW traces

tracestart ibtpci

Enables tracing for ibtpci option

triage

Collects triage and first failure logs and saves them to the desktop

Also collects driver and KMDF IFR logs

otpver

Returns the format, content, and OTP versions of a BT device

otpver D:\Otpver.txt

Returns the format, content, and OTP versions of a BT device

Redirects the output to a txt file in the specified path D:\Otpver.txt.

getLinkInfo

Obtains a specific device's piconet information

piconet

Gets Bluetooth piconet info for connected devices

tracestop

Stops the ongoing tracing for all specified options

Notes on Firmware Tracing:

Bluetooth firmware supports only a few different debug tracing methods.

However, the release firmware supports only one method of tracing called Binary tracing.

To configure tracing, refer to the following parameters:

output interface (oi) (default=vs_hci):

vs_hci

mem_store

i2c

uart

ahb

add timestamp (at) (default=no):

yes

no

overflow mode (om) (default=ow_old):

dc_new

ow_old

trace selection pattern (tsp) (default=lc):

tl/tld

hci

lm

lec

llc

sp

lc

bc

Multiple trace selection patterns can be used. Correct usage requires that arguments are passed with comma separation, and no spaces.

Arguments are not required; if they are omitted or incorrectly entered, default values will be set.

Further Firmware Trace Details:

1. op codes - HCI_Intel_FW_Trace_Config - ocf = 7C, ogf = FC
2. trace state - 0x00 disables tracing and 0x01 enables tracing
3. debug output interface - 0x01 for HCI VS Event, 0x02 for Memory Storage, 0x04 for Emulated I2C,
0x08 for Debug UART FIFO, 0x10 for AHB Bus
4. timestamp - 0x00 does not insert timestamp, 0x01 inserts timestamp
5. overflow mode - 0x00 overwrites oldest traces, 0x01 discards newest traces
6. trace selection pattern - 0x00 trace selection pattern off, 0xFF trace selection pattern on
7. Intel WRT Reporting Tool must be installed on the system in order to get WRT logs