ethtool is the primary means in Linux kernel-based operating systems (primarily Linux and Android) for displaying and modifying the parameters of network interface controllers (NICs) and their associated device driver software from application programs running in userspace.

ethtool consists of two components, an API within the Linux kernel through which NICs can send and receive parameters through their device driver software, and a userspace API based on the Linux SIOCETHTOOL ioctl mechanism through which application programs can communicate with the kernel to send and receive NIC and NIC driver parameters.

Most Linux distributions provide a standard utility program called ethtool that can be used from a shell to control, or gather information from NICs using the ethtool userspace API. In the Information technology community, the term ethtool is usually used to refer to this utility program.

The ethtool userspace API can be accessed from programs written in the C and C++ programming languages through the C standard library or C++ standard library respectively.

Several scripting languages such as Perl and Python provide ethtool API bindings that allow programmers using these languages to write scripts that can control NIC's.

The macOS and FreeBSD operating systems provide utility programs that have a user interface similar to the Linux ethtool utility, but that use fundamentally different APIs to communicate with their operating system kernels and NIC's.

Usage

The command is useful for:

Identification and diagnosis of Ethernet devices

Extended Ethernet devices statistics

Control speed, duplex, autonegotiation and flow control for Ethernet devices

Control checksum offload and other hardware offload features

Control DMA ring sizes and interrupt moderation

Control receive queue selection for multiqueue devices

Upgrade firmware in flash memory

Examples

To display the current parameters of the first network port (eth0):

$ ethtool eth0

Settings for eth0:

Supported ports: [ TP MII ]

Supported link modes: 10baseT/Half 10baseT/Full

100baseT/Half 100baseT/Full

Supports auto-negotiation: Yes

Advertised link modes: 10baseT/Half 10baseT/Full

100baseT/Half 100baseT/Full

Advertised auto-negotiation: No

Speed: 100Mb/s

Duplex: Full

Port: MII

PHYAD: 1

Transceiver: internal

Auto-negotiation: off

Supports Wake-on: g

Wake-on: g

Current message level: 0x00000007 (7)

Link detected: yes

[1]

To configure it with the 1000Mb/s speed and duplex, in 1000BASE-T:

$ ethtool -s eth0 speed 1000 duplex full autoneg off

To let the link light of the device eth0 flash for two minutes:

$ ethtool -p eth0 120

To print the driver info of the interface eth0:

$ ethtool -i eth0

driver: mlx5\_core

version: 4.9-2.2.4

firmware-version: 14.28.2006 (MT\_2420110034)

expansion-rom-version:

bus-info: 0000:65:00.1

supports-statistics: yes

supports-test: yes

supports-eeprom-access: no

supports-register-dump: no

supports-priv-flags: yes

mii-tool

mii-tool is an older program performing a similar function to ethtool. Since 2003,[2] it's considered obsolete and replaced by ethtool.

Example

To display the current parameters:

$ mii-tool

eth0: no autonegotiation, 100baseTx-HD, link ok

To force the network speed to 1 Gbit/s, and the duplex in full on the port 1 (eth0):

$ mii-tool -F 1000baseTx-FD eth0

$ mii-tool

eth0: 1 000 Mbit, full duplex, link ok

See also

Free and open-source software portal

iproute2

References

"9 Linux ethtool Examples to Manipulate Ethernet Card (NIC Card)". www.thegeekstuff.com. 2010-10-28. Retrieved 2020-11-14.

"Ethernet Auto-negotiation and Duplexing on Linux". lists.olug.org. September 2003.

External links

Official website

Ethtool examples

Net Tools on sourceforge.net