

**NAME**

ifconfig – configure a network interface

**SYNOPSIS**

```
ifconfig [-v] [-a] [-s] [interface]
ifconfig [-v] interface [atype] options | address ...
```

**DESCRIPTION**

**Ifconfig** is used to configure the kernel-resident network interfaces. It is used at boot time to set up interfaces as necessary. After that, it is usually only needed when debugging or when system tuning is needed.

If no arguments are given, **ifconfig** displays the status of the currently active interfaces. If a single **interface** argument is given, it displays the status of the given interface only; if a single **-a** argument is given, it displays the status of all interfaces, even those that are down. Otherwise, it configures an interface.

**Address Families**

If the first argument after the interface name is recognized as the name of a supported address family, that address family is used for decoding and displaying all protocol addresses. Currently supported address families include **inet** (TCP/IP, default), **inet6** (IPv6), **ax25** (AMPR Packet Radio), **ddp** (Appletalk Phase 2), **ipx** (Novell IPX) and **netrom** (AMPR Packet radio). All numbers supplied as parts in IPv4 dotted decimal notation may be decimal, octal, or hexadecimal, as specified in the ISO C standard (that is, a leading 0x or 0X implies hexadecimal; otherwise, a leading '0' implies octal; otherwise, the number is interpreted as decimal). Use of hexadecimal and octal numbers is not RFC-compliant and therefore its use is discouraged.

**OPTIONS**

- a** display all interfaces which are currently available, even if down
- s** display a short list (like netstat -i)
- v** be more verbose for some error conditions

**interface**

The name of the interface. This is usually a driver name followed by a unit number, for example **eth0** for the first Ethernet interface. If your kernel supports alias interfaces, you can specify them with syntax like **eth0:0** for the first alias of eth0. You can use them to assign more addresses. To delete an alias interface use **ifconfig eth0:0 down**. Note: for every scope (i.e. same net with address/netmask combination) all aliases are deleted, if you delete the first (primary).

- up** This flag causes the interface to be activated. It is implicitly specified if an address is assigned to the interface; you can suppress this behavior when using an alias interface by appending an **-** to the alias (e.g. **eth0:0-**). It is also suppressed when using the IPv4 0.0.0.0 address as the kernel will use this to implicitly delete alias interfaces.

- down** This flag causes the driver for this interface to be shut down.

- [-]arp** Enable or disable the use of the ARP protocol on this interface.

**[-]promisc**

Enable or disable the **promiscuous** mode of the interface. If selected, all packets on the network will be received by the interface.

**[-]allmulti**

Enable or disable **all-multicast** mode. If selected, all multicast packets on the network will be received by the interface.

- mtu N** This parameter sets the Maximum Transfer Unit (MTU) of an interface.

**dstaddr addr**

Set the remote IP address for a point-to-point link (such as PPP). This keyword is now obsolete; use the **pointopoint** keyword instead.

**netmask addr**

Set the IP network mask for this interface. This value defaults to the usual class A, B or C network mask (as derived from the interface IP address), but it can be set to any value.

**add addr/prefixlen**

Add an IPv6 address to an interface.

**del addr/prefixlen**

Remove an IPv6 address from an interface.

**tunnel ::aa.bb.cc.dd**

Create a new SIT (IPv6-in-IPv4) device, tunnelling to the given destination.

**irq addr**

Set the interrupt line used by this device. Not all devices can dynamically change their IRQ setting.

**io\_addr addr**

Set the start address in I/O space for this device.

**mem\_start addr**

Set the start address for shared memory used by this device. Only a few devices need this.

**media type**

Set the physical port or medium type to be used by the device. Not all devices can change this setting, and those that can vary in what values they support. Typical values for **type** are **10base2** (thin Ethernet), **10baseT** (twisted-pair 10Mbps Ethernet), **AUI** (external transceiver) and so on. The special medium type of **auto** can be used to tell the driver to auto-sense the media. Again, not all drivers can do this.

**[–]broadcast [addr]**

If the address argument is given, set the protocol broadcast address for this interface. Otherwise, set (or clear) the **IFF\_BROADCAST** flag for the interface.

**[–]pointopoint [addr]**

This keyword enables the **point-to-point** mode of an interface, meaning that it is a direct link between two machines with nobody else listening on it.

If the address argument is also given, set the protocol address of the other side of the link, just like the obsolete **dstaddr** keyword does. Otherwise, set or clear the **IFF\_POINTOPOINT** flag for the interface.

**hw class address**

Set the hardware address of this interface, if the device driver supports this operation. The keyword must be followed by the name of the hardware class and the printable ASCII equivalent of the hardware address. Hardware classes currently supported include **ether** (Ethernet), **ax25** (AMPR AX.25), **ARCnet** and **netrom** (AMPR NET/ROM).

**multicast**

Set the multicast flag on the interface. This should not normally be needed as the drivers set the flag correctly themselves.

**address**

The IP address to be assigned to this interface.

**txqueuelen length**

Set the length of the transmit queue of the device. It is useful to set this to small values for slower devices with a high latency (modem links, ISDN) to prevent fast bulk transfers from disturbing interactive traffic like telnet too much.

**NOTES**

Since kernel release 2.2 there are no explicit interface statistics for alias interfaces anymore. The statistics printed for the original address are shared with all alias addresses on the same device. If you want per-address statistics you should add explicit accounting rules for the address using the **iptables(8)** command.

Interrupt problems with Ethernet device drivers fail with EAGAIN (*SIOCSIFFLAGS: Resource temporarily unavailable*) it is most likely a interrupt conflict. See <http://www.scyld.com/expert/irq-conflict.html> for more information.

## FILES

`/proc/net/dev`  
`/proc/net/if_inet6`

## BUGS

Ifconfig uses the ioctl access method to get the full address information, which limits hardware addresses to 8 bytes. Because Infiniband hardware address has 20 bytes, only the first 8 bytes are displayed correctly. Please use **ip link** command from **iproute2** package to display link layer informations including the hardware address.

While appletalk DDP and IPX addresses will be displayed they cannot be altered by this command.

## SEE ALSO

`route(8)`, `netstat(8)`, `arp(8)`, `rarp(8)`, `iptables(8)`, `ifup(8)`, `interfaces(5)`.  
<http://physics.nist.gov/cuu/Units/binary.html> - Prefixes for binary multiples

## AUTHORS

Fred N. van Kempen, <waltje@u.walt.nl.mugnet.org>  
Alan Cox, <Alan.Cox@linux.org>  
Phil Blundell, <Philip.Blundell@pobox.com>  
Andi Kleen  
Bernd Eckenfels, <net-tools@lina.inka.de>