

The Unix `ifconfig` command and The Windows `ipconfig` command

Hans Kruse and Carl Bruggeman

Jan 1, 2007

The `ifconfig` command

`ifconfig` is the Swiss Army Knife of network interface configuration. With `ifconfig` you can set the various properties of network interfaces, such as the address, network mask, and many more. `ifconfig` is normally used by low-level operating system scripts to configure network interfaces as the system starts up, or as transient interfaces such as wireless or USB devices are brought up and down. `ifconfig` only changes the in-memory settings of an interface, it does not change configuration files on disk.

What does the `ifconfig` output mean?

`ifconfig` can also be used to examine the current settings of a network interface in detail. Here is an example from Mac OS X:

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    inet6 fe80::216:cbff:fe08:2aee%en1 prefixlen 64 scopeid 0x5
    inet 192.168.0.17 netmask 0xfffff00 broadcast 192.168.0.255
    ether 00:16:cb:08:2a:ee
    media: autoselect status: active
    supported media: autoselect
```

lo0: This is the “loop-back” interface. Every Unix system has one, and you want to leave it alone. It is used for a variety of internal operating system functions.

gif0: You will see a number of special-purpose network interfaces. Some support specialized system functions, and some are associated with devices like serial ports and Firewire ports.

en1: In this example, this is the only “real” network interface. Look for a number of items here:

flags – in this line look for the `UP` in the list of flags. If you do not see this flag, no packets will travel through this interface.

inet6 – this is the IPv6 configuration information, you will see more of this later in the quarter.

`inet` – here is the IPv4 information, the address, the network mask (in hexadecimal), and the broadcast address (the address with all 1's in the host portion).

`ether` – this is the hardware address, we will talk about that soon.

ifconfig on different Unix flavors

Because `ifconfig` has to be capable of manipulating all aspects of the network interfaces, its options differ a little from system to system. This document contains the `man` pages for both OS X and Linux. We will use only a small subset of the capabilities of `ifconfig` in the lab, so the main difference you will see is in the format of the `ifconfig` output.

The ipconfig command on Windows

Windows does not have an `ifconfig` command (there is a facility called `netsh` which we will use later in the quarter). To display interface parameters, you use the `ipconfig` command, usually in the form `ipconfig /all`. While the format is very different, you will see similar information as in the `ifconfig` example above. `ipconfig` can manipulate the DHCP behavior of an interface, we will see that in a few weeks.

The ifconfig man page from Mac OS X

IFCONFIG(8) BSD System Manager's Manual IFCONFIG(8)

NAME

ifconfig -- configure network interface parameters

SYNOPSIS

```
ifconfig [-L] [-m] interface [create] [address_family]
  [address[/prefixlength] [dest_address]] [parameters]
ifconfig interface destroy
ifconfig -a [-L] [-d] [-m] [-u] [address_family]
ifconfig -l [-d] [-u] [address_family]
ifconfig [-L] [-b] [-d] [-m] [-u]
ifconfig interface vlan vlan-tag vlandev iface
ifconfig interface -vlandev iface
ifconfig interface bonddev iface
ifconfig interface -bonddev iface
```

DESCRIPTION

Ifconfig is used to assign an address to a network interface and/or configure network interface parameters. Ifconfig must be used at boot time to define the network address of each interface present on a machine; it may also be used at a later time to redefine an interface's address or other operating parameters.

The following options are available:

address

For the DARPA-Internet family, the address is either a host name present in the host name data base, hosts(5), or a DARPA Internet address expressed in the Internet standard 'dot notation'.

It is also possible to use the CIDR notation (also known as the slash notation) to include the netmask. That is, one can specify an address like 192.168.0.1/16.

address_family

Specify the address family which affects interpretation of the remaining parameters. Since an interface can receive transmissions in differing protocols with different naming schemes, specifying the address family is recommended. The address or protocol families currently supported are 'inet', 'inet6',

dest_address

Specify the address of the correspondent on the other end of a point to point link.

interface

This parameter is a string of the form 'name unit', for example, 'en0'.

`iface` This parameter has the same encoding as the interface parameter.

The following parameters may be set with `ifconfig`:

`add` Another name for the alias parameter. Introduced for compatibility with BSD/OS.

`alias` Establish an additional network address for this interface. This is sometimes useful when changing network numbers, and one wishes to accept packets addressed to the old interface. If the address is on the same subnet as the first network address for this interface, a netmask of `0xffffffff` has to be specified.

`-alias` Remove the network address specified. This would be used if you incorrectly specified an alias, or it was no longer needed. If you have incorrectly set an NS address having the side effect of specifying the host portion, removing all NS addresses will allow you to respecify the host portion.

`anycast`
(Inet6 only.) Specify that the address configured is an anycast address. Based on the current specification, only routers may configure anycast addresses. Anycast address will not be used as source address of any of outgoing IPv6 packets.

`arp` Enable the use of the Address Resolution Protocol (`arp(4)`) in mapping between network level addresses and link level addresses (default). This is currently implemented for mapping between DARPA Internet addresses and IEEE 802 48-bit MAC addresses (Ethernet, FDDI, and Token Ring addresses).

`-arp` Disable the use of the Address Resolution Protocol (`arp(4)`).

`bonddev iface`
If the interface is a bond pseudo device, associate physical interface `iface` with it. The bond pseudo device conforms to the IEEE 802.3ad Link Aggregation specification.

If this is the first physical interface to be associated with the bond interface, the bond interface inherits the ethernet address from the physical interface. Physical interfaces that are added to the bond have their ethernet address re-programmed so that all members of the bond have the same ethernet address. If the physical interface is subsequently removed from the bond using `-bonddev`, a new ethernet address is chosen from the remaining interfaces, and all interfaces are re-programmed again with the new ethernet address. If no remaining interfaces exist, the bond interface's ethernet address is cleared.

If the specified physical interface `iface` is not capable of hav-

ing its ethernet address re-programmed, the `bonddev` command will fail.

Once the physical interface `iface` is successfully associated with the bond interface, all received packets are diverted to the bond interface. The physical interface is no longer useable on its own, and remains that way until it is removed from the bond using `-bonddev`.

It is possible that the specified interface `iface` is not capable of aggregating, and may remain unused until the operating conditions change.

The link status of the bond interface depends on the state of link aggregation. If no active partner is detected, the link status will remain inactive.

To monitor the 802.3ad Link Aggregation state, use the `-b` option.

A physical interface that is associated with a vlan pseudo device cannot at the same time be associated with a bond pseudo device. A physical interface cannot be associated with more than one bond pseudo device at the same time.

It is not possible to associate a bond with pseudo interfaces such as vlan. Only physical ethernet interfaces may be associated with a bond.

`-bonddev iface`

If the interface is a bond pseudo device, disassociate the physical interface `iface` from it. Before the interface is removed from the bond, the bond device announces to the link partner that the interface is now individual and no longer aggregatable. If the physical `iface` is the last interface in the bond, the bond interface clears its link address.

`broadcast`

(Inet only.) Specify the address to use to represent broadcasts to the network. The default broadcast address is the address with a host part of all 1's.

`debug` Enable driver dependent debugging code; usually, this turns on extra console error logging.

`-debug` Disable driver dependent debugging code.

`delete` Another name for the `-alias` parameter.

`down` Mark an interface 'down'. When an interface is marked 'down', the system will not attempt to transmit messages through that interface. If possible, the interface will be reset

to disable reception as well. This action does not automatically disable routes using the interface.

`ether` Another name for the `lladdr` parameter.

`lladdr addr`

Set the link-level address on an interface. This can be used to e.g. set a new MAC address on an ethernet interface, though the mechanism used is not ethernet-specific. The address `addr` is specified as a series of colon-separated hex digits. If the interface is already up when this option is used, it will be briefly brought down and then brought back up again in order to ensure that the receive filter in the underlying ethernet hardware is properly reprogrammed.

`media type`

If the driver supports the media selection system, set the media type of the interface to `type`. Some interfaces support the mutually exclusive use of one of several different physical media connectors. For example, a 10Mb/s Ethernet interface might support the use of either AUI or twisted pair connectors. Setting the media type to `‘‘10base5/AUI’’` would change the currently active connector to the AUI port. Setting it to `‘‘10baseT/UTP’’` would activate twisted pair. Refer to the interfaces’ driver specific documentation or man page for a complete list of the available types.

`mediaopt opts`

If the driver supports the media selection system, set the specified media options on the interface. The `opts` argument is a comma delimited list of options to apply to the interface. Refer to the interfaces’ driver specific man page for a complete list of available options.

`-mediaopt opts`

If the driver supports the media selection system, disable the specified media options on the interface.

`tunnel src_addr dest_addr`

(IP tunnel devices only.) Configure the physical source and destination address for IP tunnel interfaces (`gif(4)`). The arguments `src_addr` and `dest_addr` are interpreted as the outer source/destination for the encapsulating IPv4/IPv6 header.

`deletetunnel`

Unconfigure the physical source and destination address for IP tunnel interfaces previously configured with `tunnel`.

`create` Create the specified network pseudo-device. If the interface is given without a unit number, try to create a new device with an arbitrary unit number. If creation of an arbitrary device is

successful, the new device name is printed to standard output.

destroy

Destroy the specified network pseudo-device.

plumb Another name for the create parameter. Included for Solaris compatibility.

unplumb

Another name for the destroy parameter. Included for Solaris compatibility.

metric n

Set the routing metric of the interface to n, default 0. The routing metric is used by the routing protocol (routed(8)). Higher metrics have the effect of making a route less favorable; metrics are counted as addition hops to the destination network or host.

mtu n Set the maximum transmission unit of the interface to n, default is interface specific. The MTU is used to limit the size of packets that are transmitted on an interface. Not all interfaces support setting the MTU, and some interfaces have range restrictions.

netmask mask

(Inet only.) Specify how much of the address to reserve for subdividing networks into sub-networks. The mask includes the network part of the local address and the subnet part, which is taken from the host field of the address. The mask can be specified as a single hexadecimal number with a leading '0x', with a dot-notation Internet address, or with a pseudo-network name listed in the network table networks(5). The mask contains 1's for the bit positions in the 32-bit address which are to be used for the network and subnet parts, and 0's for the host part. The mask should contain at least the standard network portion, and the subnet field should be contiguous with the network portion.

The netmask can also be specified in CIDR notation after the address. See the address option above for more information.

prefixlen len

(Inet6 only.) Specify that len bits are reserved for subdividing networks into sub-networks. The len must be integer, and for syntactical reason it must be between 0 to 128. It is almost always 64 under the current IPv6 assignment rule. If the parameter is omitted, 64 is used.

remove Another name for the -alias parameter. Introduced for compatibility with BSD/OS.

link[0-2]

Enable special processing of the link level of the interface. These three options are interface specific in actual effect, however, they are in general used to select special modes of operation. An example of this is to enable SLIP compression, or to select the connector type for some Ethernet cards. Refer to the man page for the specific driver for more information.

-link[0-2]

Disable special processing at the link level with the specified interface.

up Mark an interface ‘‘up’’. This may be used to enable an interface after an ‘‘ifconfig down’’. It happens automatically when setting the first address on an interface. If the interface was reset when previously marked down, the hardware will be re-initialized.

vlan vlan_tag vlandev iface

If the interface is a vlan pseudo interface, set its vlan tag value to vlan_tag and associate it with the physical interface iface.

The vlan_tag value is a 16-bit number that is used to create an 802.1Q vlan header for packets sent from the vlan interface.

A packet that is transmitted through the vlan interface is sent using the specified physical interface iface with 802.1Q vlan encapsulation with the specified vlan_tag. A packet with 802.1Q encapsulation received by the physical interface is directed to the associated vlan interface with the matching vlan_tag. If there is no matching vlan interface, the packet is dropped.

The vlan interface is assigned a copy of the parent interface’s flags and the parent’s ethernet address. If the vlan interface already has a physical interface associated with it, this command will fail. To change the association to another physical interface, the existing association must be cleared first using -vlandev.

If the physical interface supports 802.1Q VLAN tagging in hardware, the vlan pseudo interface does not itself insert or remove the 802.1Q encapsulation header. Instead, the vlan_tag is passed out of band from the packet data.

A physical interface that is associated with a bond pseudo device cannot at the same time be associated with a vlan interface. However, a physical interface can be associated with multiple vlan interfaces at the same time, as long as each of the vlan_tag values are unique.

`-vlandev iface`

If the driver is a vlan pseudo device, disassociate the physical interface `iface` from it. This breaks the link between the vlan interface and its parent, clears its vlan tag, flags and its link address.

Ifconfig displays the current configuration for a network interface when no optional parameters are supplied. If a protocol family is specified, ifconfig will report only the details specific to that protocol family.

If the driver supports the media selection system, the supported media list will be included in the output, regardless of whether the `-m` flag is passed or not.

The `-b` option passed before the interface name will print the link aggregation state for bond pseudo devices.

If `-L` flag is supplied, address lifetime is displayed for IPv6 addresses, as time offset string.

Optionally, the `-a` flag may be used instead of an interface name. This flag instructs ifconfig to display information about all interfaces in the system. The `-d` flag limits this to interfaces that are down, and `-u` limits this to interfaces that are up. When no arguments are given, `-a` is implied.

The `-l` flag may be used to list all available interfaces on the system, with no other additional information. Use of this flag is mutually exclusive with all other flags and commands, except for `-d` (only list interfaces that are down) and `-u` (only list interfaces that are up).

Only the super-user may modify the configuration of a network interface.

NOTES

The media selection system is relatively new and only some drivers support it (or have need for it).

DIAGNOSTICS

Messages indicating the specified interface does not exist, the requested address is unknown, or the user is not privileged and tried to alter an interface's configuration.

BUGS

IPv6 link-local addresses are required for several basic communication between IPv6 node. If they are deleted by ifconfig manually, the kernel might show very strange behavior. So, such manual deletions are strongly discouraged.

SEE ALSO

`netstat(1)`, `netintro(4)`, `rc(8)`, `routed(8)`

HISTORY

The `ifconfig` command appeared in 4.2BSD.

BSD July 15, 2004 BSD

The ifconfig man page from Linux

IFCONFIG(8) Linux Programmers Manual IFCONFIG(8)

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).

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 eth0:0 for the first alias of eth0. You can use them to assign a second address. To delete an alias interface use ifconfig eth0:0 down 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.

`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.

`metric N`
This parameter sets the interface metric.

`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 `pointpoint` 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 ipchains(8) or iptables(8) command.

Since net-tools 1.60-4 ifconfig is printing byte counters and human readable counters with IEC 60027-2 units. So 1 KiB are 2¹⁰ byte. Note, the numbers are truncated to one decimal (which can be quite a large error if you consider 0.1 PiB is 112.589.990.684.262 bytes :)

Interrupt problems with Ethernet device drivers fail with EAGAIN (SIOC SIIFLAGS: 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/socket
/proc/net/dev
/proc/net/if_inet6

BUGS

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), ipchains(8), iptables(8)
<http://physics.nist.gov/cuu/Units/binary.html> - Prefixes for binary multiples

AUTHORS

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

net-tools 2005-06-30 IFCONFIG(8)