NAME

ifconfig – configure network interface parameters

SYNOPSIS

/sbin/ifconfig *interface* [*address_family*] [*address* [*dest_address*]] [**up**] [**down**]
[**auto-revarp**] [**netmask** *mask*] [**broadcast** *address*] [**metric** *n*] [**mtu** *n*]
[**trailers** | **−trailers**] [**private** | **−private**] [**arp** | **−arp**] [**plumb**] [**unplumb**]

/usr/sbin/ifconfig *interface* [*address_family*] [*address* [*dest_address*]] [**up**] [**down**]
[**auto-revarp**] [**netmask** *mask*] [**broadcast** *address*] [**metric** *n*] [**mtu** *n*]
[**trailers** | **−trailers**] [**private** | **−private**] [**arp** | **−arp**] [**plumb**] [**unplumb**]

AVAILABILITY

SUNWcsu

DESCRIPTION

ifconfig is used to assign an address to a network interface and/or to 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. Used without options, **ifconfig** displays the current configuration for a network interface. If a protocol family is specified, **ifconfig** will report only the details specific to that protocol family. Only the super-user may modify the configuration of a network interface.

The *interface* parameter is a string of the form *name physical-unit*, for example **le0** or **ie1**, or of the form *name physical-unit* : *logical-unit*, for example **le0:1**. Three special interface names, −**a**, −**ad** and −**au**, are reserved and refer to all or a subset of the interfaces in the system. If one of these interface names is given, the commands following it are applied to all of the interfaces that match:

−a Apply the commands to all interfaces in the system.

−ad Apply the commands to all ''down'' interfaces in the system.

−au Apply the commands to all ''up'' interfaces in the system.

Since an interface may receive transmissions in differing protocols, each of which may require separate naming schemes, the parameters and addresses are interpreted according to the rules of some address family, specified by the *address_family* parameter. The address families currently supported are **ether** and **inet**. If no address family is specified, **inet** is assumed.

For the TCP/IP family (**inet**), the address is either a host name present in the host name data base (see **hosts**(4)) or in the Network Information Service (NIS) map **hosts**, or a TCP/IP address expressed in the Internet standard "dot notation". Typically, an Internet address specified in dot notation will consist of your system's network number and the machine's unique host number. A typical Internet address is **192.9.200.44**, where **192.9.200** is the network number and **44** is the machine's host number.

For the **ether** address family, the address is an Ethernet address represented as *x*:*x*:*x*:*x*:*x*:*x* where *x* is a hexadecimal number between **0** and **FF**.

If the *dest_address* parameter is supplied in addition to the *address* parameter, it specifies the address of the correspondent on the other end of a point to point link.

OPTIONS

arp Enable the use of the Address Resolution Protocol in mapping between network level addresses and link level addresses (default). This is currently implemented for mapping between TCP/IP addresses and 10Mb/s Ethernet addresses.

–arp Disable the use of the Address Resolution Protocol.

auto-revarp Use the Reverse Address Resolution Protocol (RARP) to automatically acquire an address for this interface.

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.

plumb Open the device associated with the physical interface name and setup the streams needed for TCP/IP to use the device. Before this is done the interface will not show up in the output of **ifconfig –a**.

unplumb Destroy any streams associated with this device and close the device. After this command is executed the device name should not show up in the output of **ifconfig –a**.

private Tells the **in.routed** routing daemon that the interface should not be advertised.

–private Specify unadvertised interfaces.

trailers This flag used to cause a non-standard encapsulation of inet packets on certain link levels. Drivers supplied with this release no longer use this flag, but it is ignored for compatibility.

–trailers Disable the use of a "trailer" link level encapsulation.

up Mark an interface "up". This happens automatically when setting the first address on an interface. The **up** option enables an interface after an **ifconfig down**, reinitializing the hardware.

broadcast *address* (**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. A **+** (plus sign) given for the broadcast value causes the broadcast address to be reset to a default appropriate for the (possibly new) address and netmask. **Note**: The arguments of **ifconfig** are interpreted left to right, and therefore

ifconfig –a netmask + broadcast +

and

ifconfig –a broadcast + netmask +

may result in different values being assigned for the interfaces' broadcast addresses.

metric *n* — Set the routing metric of the interface to *n*, default **0**. The routing metric is used by the routing protocol 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*. For many types of networks the mtu has an upper limit, for example, **1500** for Ethernet.

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 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 mask can be specified in one of four ways: a) with a single hexadecimal number with a leading 0x. b) with a dot-notation address, c) with a '**+**' (plus sign) address, or d) with a pseudo host name/pseudo network name listed in the network table **networks**(4). If a '**+**' (plus sign) is given for the netmask value, the mask is looked up in the **netmasks database** using the interface network number as the key.

If a pseudo host name/pseudo network name is supplied as the netmask value, netmask data may be located in the **hosts** or **networks** table. **ifconfig** first looks up the name in the **hosts** table. If an entry is found, the host IP address is used as the netmask. If the entry isn't found there, **ifconfig** looks up the name in the **networks** table. (If the entry is found there, the IP network address is padded with a zero.) The **hosts** and **netmasks** tables are not designed for storage of netmasks; adding a netmask entry by using this option may confuse other programs, so we advise against using it. The system administrator may configure the source and lookup order in the tables via the name service switch. See **nsswitch.conf**(4) for more information.

LOGICAL INTERFACES

Solaris TCP/IP allows multiple logical interfaces to be associated with a physical network interface. This allows a single machine to be assigned multiple IP addresses, even though it may have only one network interface. Physical network interfaces have names of the form *driver-name physical-unit-number*, while logical interfaces have names of the form *driver-name physical-unit-number logical-unit-number.* A physical interface is configured into the system using the plumb sub-command, for example:

```
ifconfig le0 plumb
```

Once a physical interface has been "plumbed", additional local interfaces can be configured by simply naming them in subsequent ifconfig commands. Logical interfaces do not need to be "plumbed". Simply mentioning their name in an ifconfig command is sufficient. For example, the command:

ifconfig le0:1

will allocate a logical interface associated with the physical interface le0. A logical interface can be configured with parameters (address, netmask, etc.) different from the physical interface that it is associated with. And logical interfaces that are associated with the same physical interface can be given different parameters as well. Each logical interface must be associated with a physical interface. So, for example, the logical interface le0:1 can only be configured after the physical interface le0 has been plumbed.

EXAMPLES

If your workstation is not attached to an Ethernet, the **le0** interface should be marked "down" as follows:

example% ifconfig le0 down

To print out the addressing information for each interface, use the following command:

example% ifconfig −a

To reset each interface's broadcast address after the netmasks have been correctly set, use the next command:

example% ifconfig −a broadcast +

To change the Ethernet address for interface **le0**, use the following command:

example% ifconfig le0 ether aa:1:2:3:4:5

FILES

/etc/netmasks netmask data

SEE ALSO

in.routed(1M), **netstat**(1M), **ethers**(3N), **hosts**(4), **netmasks**(4), **networks**(4), **nsswitch.conf**(4), **arp**(7P)

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.

NOTES

It is recommended that the names **broadcast**, **down**, **private**, **trailers**, **up**, and the other possible option names not be selected when choosing host names. Choosing anyone of these names as host names will cause bizarre problems that can be extremely difficult to diagnose.