NAME

iftop - display bandwidth usage on an interface by host

SYNOPSIS

iftop -h | [**-nNpbBP**] [**-i** *interface*] [**-f** *filter code*] [**-F** *net*/*mask*]

DESCRIPTION

iftop listens to network traffic on a named *interface*, or on the first interface it can find which looks like an external interface if none is specified, and displays a table of current bandwidth usage by pairs of hosts. **iftop** must be run with sufficient permissions to monitor all network traffic on the *interface*; see **pcap**(3) for more information, but on most systems this means that it must be run as root.

By default, **iftop** will look up the hostnames associated with addresses it finds in packets. This can cause substantial traffic of itself, and may result in a confusing display. You may wish to suppress display of DNS traffic by using filter code such as **not port domain**, or switch it off entirely, by using the **-n** option or by pressing **R** when the program is running.

By default, **iftop** counts all IP packets that pass through the filter, and the direction of the packet is determined according to the direction the packet is moving across the interface. Using the **-N** option it is possible to get **iftop** to show packets entering and leaving a given network. For example, **iftop -N 10.0.0.0/255.0.0.0** will analyse packets flowing in and out of the 10.* network.

Some other filter ideas:

not ether host ff:ff:ff:ff:ff:ff
: Ignore ethernet broadcast packets.

port http and not host *webcache.example.com*
: Count web traffic only, unless it is being directed through a local web cache.

icmp How much bandwith are users wasting trying to figure out why the network is slow?

OPTIONS

-h Print a summary of usage.

-n Don't do hostname lookups.

-N Do not resolve port number to service names

-p Run in promiscuous mode, so that traffic which does not pass directly through the specified interface is also counted.

-P Turn on port display.

-b Don't display bar graphs of traffic.

-B Display bandwidth rates in bytes/sec rather than bits/sec.

-i *interface*
: Listen to packets on *interface*.

-f *filter code*
: Use *filter code* to select the packets to count. Only IP packets are ever counted, so the specified code is evaluated as **(***filter code***) and ip**.

-F *net*/*mask*
: Specifies a network for traffic analysis. If specified, iftop will only include packets flowing in to or out of the given network, and packet direction is determined relative to the network boundary, rather than to the interface. You may specify *mask* as a dotted quad, such as /255.255.255.0, or as a single number specifying the number of bits set in the netmask, such as /24.

-c *config file*
Specifies an alternate config file. If not specified, iftop will use **~/.iftopc** if it exists. See below for a description of config files

DISPLAY

When running, **iftop** uses the whole screen to display network usage. At the top of the display is a logarithmic scale for the bar graph which gives a visual indication of traffic.

The main part of the display lists, for each pair of hosts, the rate at which data has been sent and received over the preceding 2, 10 and 40 second intervals. The direction of data flow is indicated by arrows, <= and =>. For instance,

```
foo.example.com  =>  bar.example.com     1Kb  500b   100b
                 <=                      2Mb    2Mb    2Mb
```

shows, on the first line, traffic from **foo.example.com** to **bar.example.com**; in the preceding 2 seconds, this averaged 1Kbit/s, around half that amount over the preceding 10s, and a fifth of that over the whole of the last 40s. During each of those intervals, the data sent in the other direction was about 2Mbit/s. On the actual display, part of each line is inverted to give a visual indication of the 10s average of traffic. You might expect to see something like this where host **foo** is making repeated HTTP requests to **bar**, which is sending data back which saturates a 2Mbit/s link.

By default, the pairs of hosts responsible for the most traffic (10 second average) are displayed at the top of the list.

At the bottom of the display, various totals are shown, including peak traffic over the last 40s, total traffic transferred (after filtering), and total transfer rates averaged over 2s, 10s and 40s.

SOURCE / DEST AGGREGATION

By pressing **s** or **d** while **iftop** is running, all traffic for each source or destination will be aggregated together. This is most useful when **iftop** is run in promiscuous mode, or is run on a gateway machine.

PORT DISPLAY

S or **D** toggle the display of source and destination ports respectively. **p** will toggle port display on/off.

DISPLAY TYPE

t cycles through the four line display modes; the default 2-line display, with sent and received traffic on separate lines, and 3 1-line displays, with sent, received, or total traffic shown.

DISPLAY ORDER

By default, the display is ordered according to the 10s average (2nd column). By pressing **1**, **2** or **3** it is possible to sort by the 1st, 2nd or 3rd column. By pressing < or > the display will be sorted by source or destination hostname respectively.

DISPLAY FILTERING

l allows you to enter a POSIX extended regular expression that will be used to filter hostnames shown in the display. This is a good way to quickly limit what is shown on the display. Note that this happens at a much later stage than filter code, and does not affect what is actually captured. Display filters DO NOT affect the totals at the bottom of the screen.

PAUSE DISPLAY / FREEZE ORDER

P will pause the current display.

o will freeze the current screen order. This has the side effect that traffic between hosts not shown on the screen at the time will not be shown at all, although it will be included in the totals at the bottom of the screen.

SCROLL DISPLAY

j and **k** will scroll the display of hosts. This feature is most useful when the display order is frozen (see above).

FILTER CODE

f allows you to edit the filter code whilst iftop running. This can lead to some unexpected behaviour.

CONFIG FILE

iftop can read its configuration from a config file. If the **-c** option is not specified, iftop will attempt to read its configuration from **~/.iftoprc**, if it exists. Any command line options specified will override settings in the config file.

The config file consists of one configuration directive per line. Each directive is a name value pair, for example:

interface: eth0

sets the network interface. The following config directives are supported:

interface: *if*
Sets the network interface to *if*.

dns-resolution: *(yes|no)*
Controls reverse lookup of IP addresses.

port-resolution: *(yes|no)*
Controls conversion of port numbers to service names.

filter-code: *bpf*
Sets the filter code to *bpf*.

show-bars: *(yes|no)*
Controls display of bar graphs.

promiscuous: *(yes|no)*
Puts the interface into promiscuous mode.

port-display: *(off|source-only|destination-only|on)*
Controls display of port numbers.

hide-source: *(yes|no)*
Hides source host names.

hide-destination: *(yes|no)*
Hides destination host names.

use-bytes: *(yes|no)*
Use bytes for bandwidth display, rather than bits.

sort: *(2s|10s|40s|source|destination)*
Sets which column is used to sort the display.

line-display: *(two-line|one-line-both|one-line-sent|one-line-received)*
Controls the appearance of each item in the display.

show-totals: *(yes|no)*
Shows cummulative total for each item.

log-scale: *(yes|no)*
Use a logarithmic scale for bar graphs.

max-bandwidth: *bw*
Fixes the maximum for the bar graph scale to *bw*, e.g. "10M"

net-filter: *net/mask*
Defines an IP network boundary for determining packet direction.

screen-filter: *regexp*
Sets a regular expression to filter screen output.

QUIRKS (aka they're features, not bugs)

There are some circumstances in which iftop may not do what you expect. In most cases what it is doing is logical, and we believe it is correct behaviour, although I'm happy to hear reasoned arguments for alternative behaviour.

Totals don't add up

There are several reasons why the totals may not appear to add up. The most obvious is having a screen filter in effect, or screen ordering frozen. In this case some captured information is not being shown to you, but is included in the totals.

A more subtle explanation comes about when running in promiscuous mode without specifying a **-N** option. In this case there is no easy way to assign the direction of traffic between two third parties. For the purposes of the main display this is done in an arbitrary fashion (by ordering of IP addresses), but for the sake of totals all traffic between other hosts is accounted as incoming, because that's what it is from the point of view of your interface. The **-N** option allows you to specify an arbitrary network boundary, and to show traffic flowing across it.

Peak totals don't add up

Again, this is a feature. The peak sent and peak received didn't necessarily happen at the same time. The peak total is the maximum of sent plus received in each captured time division.

Changing the filter code doesn't seem to work

Give it time. Changing the filter code affects what is captured from the time that you entered it, but most of what is on the display is based on some fraction of the last 40s window of capturing. After changing the filter there may be entries on the display that are disallowed by the current filter for up to 40s. DISPLAY FILTERING has immediate effect and does not affect what is captured.

FILES

~/.iftoprc
Configuration file for iftop.

SEE ALSO

tcpdump(8), **pcap**(3), **driftnet**(1).

AUTHOR

Paul Warren <pdw@ex-parrot.com>

VERSION

$Id: iftop.8,v 1.24 2003/10/22 19:28:31 pdw Exp $

COPYING