IP Filter

Current version: 5.1.0

Next release status

Patches for last release

What's new ? Click here!

Mailing list ?

Send mail to
majordomo@coombs.anu.edu.au
with "subscribe ipfilter" in the body of the mail.

What is it ?

IPFilter is a software package that can be used to provide network address translation (NAT) or firewall services. To use, it can either be used as a loadable kernel module or incorporated into your UNIX kernel; use as a loadable kernel module where possible is highly recommended. Scripts are provided to install and patch system files, as required.

To see an overview of how IP Filter fits into the overall picture of TCP/IP with your kernel and the order in which the various phases of packet processing is done, click here.

The IPFilter **FAQ** by Phil Dibowitz!

It comes as a part of the following operating systems:

- FreeBSD-current (post 2.2)
- NetBSD-current (post 1.2)
- xMach
- Solaris 10
- Open Solaris

It has been tested and run on:

- Solaris/Solaris-x86 2.3 - 9
- SunOS 4.1.4 - 4.1.4
- NetBSD 1.0 - 1.4
- FreeBSD 2.0.0 - 2.2.8
- BSD/OS-1.1 - 4
- IRIX 6.2, 6.5
- OpenBSD 2.0 - 3.5
- Linux(*) 2.4 - 2.6
- HP-UX 11.00
- Tru64 5.1a
- AIX 5.3 ML05
- QNX 6 Port

* - It has been tested and shown to work on RedHat 9.0, SuSE 9.1 and will, in general work with 2.4 and 2.6 kernels. It should be noted that not all Linux distros are the same so using others may not be smooth.

Donations

Thanks to those who have been able to support IP Filter through donations of hardware.

Download

To ftp this package, see:

the **list of mirrors**

ftp to ftp://coombs.anu.edu.au/pub/net/ip-filter/ip_fil5.1.0.tar.gz

or via http from http://coombs.anu.edu.au/~avalon/ip_fil5.1.0.tar.gz

Beta-versions can usually be found, if available, in ftp://coombs.anu.edu.au/pub/net/ip-filter/

HOW-TO

A How-To document is being written for IP Filter and is avialable at http://www.obfuscation.org/ipf/

Mirrors

- http://www.unixcircle.com/ipf/ [San Jose, CA, USA]
- http://www.pir.net/pir/ipf/ [Boston, MA, USA]
- http://www.openlysecure.org/content/html/www.obfuscation.org/ipf [Surrey, UK]
- http://www.grunta.com/ipf/ [Melbourne, Victoria, AU]
- http://www.darkart.com/mirrors/www.obfuscation.org/ipf/ [Oakland, CA, USA]

In German, you can download this document:

IP Filter Based Firewalls HOWTO-German.pdf

The firewall can:

- explicitly **deny/permit** any packet from passing through
- distinguish between various **interfaces**

and can match on the follow IP header fields:

- **source/destination IP address** (including inverted matches)
- **IP protocol**
- **TOS (Type of Service)**
- any of the 19 **IP options** or 8 registered **IP security classes**
- **fragments** (if it is or isn't)

In addition, IPFilter can

- send back an **ICMP error/TCP reset** for denied packets
- keep **packet state** information for TCP, UDP and ICMP packet flows.
- keep **fragment state** information for any IP packet, applying the same rule to all fragments.
- act as a **Network Address Translator (NAT)**
- use **redirection** to setup true **transparent proxy connections**.
- **provide packet header details to a user program for authentication**
- in addition, supports temporary storage of **pre-authenticated rules** for passing packets through

Special provision is made for the three most common Internet protocols, TCP, UDP and ICMP. IP Filter rules allow for packets to be matched based on:

- TCP/UDP packets by **port number** or a **port number range**
- ICMP packets by **type/code**
- **"established"** TCP packets
- on any arbitrary combination of **TCP flags**
- **"short" (fragmented)** IP packets with incomplete headers

To keep track of the performance of IP Filter, a logging device is used which supports logging of:

- the TCP/UDP/ICMP and IP packet headers
- the first 128 bytes of the packet (including headers)

when:

- a packet is successfully **passed** through
- a packet is **blocked** from passing through
- it matches a rule setup to look for suspicious packets

To examine a set of example rule files and an example of what can be done, click here.

IPFilter keeps its own set of statistics on:

- packets blocked
- packets (and bytes!) used for accounting
- packets passed
- packets logged
- attempts to log which failed (buffer full)

and much more, for packets going both in and out.

The current implementation provides a small set of tools, which can easily be used and integrated with regular unix shells and tools. Amongst these tools is a new addition, ipftest, which is provided so that you can test a rule set before committing it to use in your kernel. A brief description of the tools provided:

- **ipf** - reads in a set of rules, from either stdin or a file, and adds them to the kernels current list (appending them). It can also be used to flush the current firewall rule set or delete

individual firewall rules.

- **ipfstat** - interrogates the kernel for statistics on packet processing, so far, and retrieves the list of firewall rules in operation for **inbound** and **outbound** packets.

- **ipftest** - reads in a **ipf** rule file and then applies sample IP packets to the rule file. This allows for testing of firewall rule list and examination of how a packet is passed along through it.

- **ipmon** - reads buffered data from the logging device (default is /dev/ipl) for output to either:
  * screen (standard output)
  * file
  * syslog
- **ipsend** - generates arbitary IP packets for ethernet connected machines.

- **ipresend** - reads in a data file of saved IP packets (ie snoop/tcpdump/etherfind output) and sends it back across the network.

- **iptest** - contains a set of test "programs" which send out a series of IP packets, aimed at testing the strength of the TCP/IP stack at which it is aimed at. **WARNING:** may crash machine(s) targeted!

- **ipnat** - reads in a set of rules, from either stdin or a file and adds them to the kernels current list of active NAT rules. NAT rules can also be deleted using **ipnat**.

Documentation on **ioctl's** and the format of data saved to the logging character device is provided so that you may develop your own applications to work with or in place of any of the above.

To retrieve this package via anonymous ftp, use: ftp://coombs.anu.edu.au/pub/net/ip-filter/ip_fil5.1.0.tar.gz

---

Mailing List Archive

The mailing list for IP Filter is now archived at: http://marc.theaimsgroup.com/?l=ipfilter This site also supports searching of the IP Filter list archive.

Mirrors!

Australia: planetmirror.com - mirrors coombs.anu.edu.au:/pub/net/ip-filter/
Canada: ftp.localhost.ca - mirrors coombs.anu.edu.au:/pub/net/ip-filter/
Finland: nic.funet.fi - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter
United Kingdom: ftp.tardis.ed.ac.uk - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter
Germany: http://ipfilter.wormulon.net/ - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter.
Greece: ftp.ntua.gr - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter
Hungary: ftp.kfki.hu - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter
Italy: ftp.unipi.it - mirror coombs.anu.edu.au:/pub/net/ip-filter
Japan: ftp.ayamura.org - mirrors coombs.anu.edu.au/pub/net/firewall/ip-filter
Norway: ftp.netrunner.nu - mirror coombs.anu.edu.au:/pub/net/ip-filter
Poland:
ftp.task.gda.pl - mirrors coombs.anu.edu.au:/pub/net/ip-filter
http://ftyczka.org/ipf/
Spain: cache.unicies.cesga.es - mirrors /pub/net/firewall/ip-filter
Sweden: ftp.sekure.net - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter/
Taiwan: http://pds.nchu.edu.tw/pub/firewall/ip-filter - mirrors this web site.
Turkey: www.enderunix.org - mirrors http://coombs.anu.edu.au/~avalon/ipfilter/
USA:
ftp://ftp.gw.com/pub/unix/ip-filter/
ftp.umbc.edu - mirrors coombs.anu.edu.au:/pub/net/kernel
ftp.twtelecom.net - mirrors coombs.anu.edu.au:/pub/firewall/ip-filter
ftp.tmcs.net - mirrors coombs.anu.edu.au:/pub/net/firewall/ip-filter

UCD SNMP

The UCD SNMP package now supports IP Filter. For more information, retrieve version 3.6.2 (or later) of the UCD SNMP package from http://net-snmp.sourceforge.net

Other IP Filter links:

IP Filter HOW-TO
http://www.charvolant.org/~doug/network/index.html Connecting a Private Network to an ISP on Solaris
HOW-TO Guide for using NAT/PPP on Solaris 2
Transparent WWW Proxying with Squid
IP Accounting package

- Tranparent IP Proxy
- NAT for OpenBSD
- IP Filter 3.2.10 for SCO
- A Guide to IPFilter written in Turkish.

Darren Reed

darrenr@pobox.com