

**NAME**

**ipfw** - IP packet filter and traffic accounting

**SYNOPSIS**

To compile the driver into the kernel, place the following option in the kernel configuration file:

```
options IPFIREWALL
```

Other related kernel options which may also be useful are:

```
options IPFIREWALL_DEFAULT_TO_ACCEPT
```

```
options IPDIVERT
```

```
options IPFIREWALL_NAT
```

```
options IPFIREWALL_NAT64
```

```
options IPFIREWALL_NPTV6
```

```
options IPFIREWALL_PMOD
```

```
options IPFIREWALL_VERBOSE
```

```
options IPFIREWALL_VERBOSE_LIMIT=100
```

```
options LIBALIAS
```

To load the driver as a module at boot time, add the following line into the loader.conf(5) file:

```
ipfw_load="YES"
```

**DESCRIPTION**

The **ipfw** system facility allows filtering, redirecting, and other operations on IP packets travelling through network interfaces.

The default behavior of **ipfw** is to block all incoming and outgoing traffic. This behavior can be modified, to allow all traffic through the **ipfw** firewall by default, by enabling the `IPFIREWALL_DEFAULT_TO_ACCEPT` kernel option. This option may be useful when configuring **ipfw** for the first time. If the default **ipfw** behavior is to allow everything, it is easier to cope with firewall-tuning mistakes which may accidentally block all traffic.

When using `natd(8)` in conjunction with **ipfw** as NAT facility, the kernel option `IPDIVERT` enables diverting packets to `natd(8)` for translation.

When using the in-kernel NAT facility of **ipfw**, the kernel option `IPFIREWALL_NAT` enables basic `libalias(3)` functionality in the kernel.

When using any of the IPv4 to IPv6 transition mechanisms in **ipfw**, the kernel option `IPFIREWALL_NAT64` enables all of these NAT64 methods in the kernel.

When using the IPv6 network prefix translation facility of **ipfw**, the kernel option `IPFIREWALL_NPTV6` enables this functionality in the kernel.

When using the packet modification facility of **ipfw**, the kernel option `IPFIREWALL_PMOD` enables this functionality in the kernel.

To enable logging of packets passing through **ipfw**, enable the `IPFIREWALL_VERBOSE` kernel option. The `IPFIREWALL_VERBOSE_LIMIT` option will prevent `syslogd(8)` from flooding system logs or causing local Denial of Service. This option may be set to the number of packets which will be logged on a per-entry basis before the entry is rate-limited.

When using the in-kernel NAT facility of **ipfw**, the kernel option `LIBALIAS` enables full `libalias(3)` functionality in the kernel. Full functionality refers to included support for ftp, bbt, skinny, irc, pptp and smedia packets, which are missing in the basic `libalias(3)` functionality accomplished with the `IPFIREWALL_NAT` kernel option.

The user interface for **ipfw** is implemented by the `ipfw(8)` utility, so please refer to the `ipfw(8)` man page for a complete description of the **ipfw** capabilities and how to use it.

#### **SEE ALSO**

`setsockopt(2)`, `divert(4)`, `ip(4)`, `ip6(4)`, `ipfw(8)`, `libalias(3)`, `natd(8)`, `sysctl(8)`, `syslogd(8)`, `pf(9)`