NAME

ng_ipfw - interface between netgraph and IP firewall

SYNOPSIS

#include <netinet/ip_var.h>
#include <netgraph/ng_ipfw.h>

DESCRIPTION

The **ipfw** node implements interface between ipfw(4) and netgraph(4) subsystems.

HOOKS

The **ipfw** node supports an arbitrary number of hooks, which must be named using only numeric characters.

OPERATION

Once the **ng_ipfw** module is loaded into the kernel, a single node named *ipfw* is automatically created. No more **ipfw** nodes can be created. Once destroyed, the only way to recreate the node is to reload the **ng_ipfw** module.

Packets can be injected into netgraph(4) using either the **netgraph** or **ngtee** commands of the ipfw(8) utility. These commands require a numeric cookie to be supplied as an argument. Packets are sent out of the hook whose name equals the cookie value. If no hook matches, packets are discarded. Packets injected via the **netgraph** command are tagged with *struct ipfw_rule_ref*. This tag contains information that helps the packet to re-enter ipfw(4) processing, should the packet come back from netgraph(4) to ipfw(4).

Packets received by a node from netgraph(4) subsystem must be tagged with *struct ipfw_rule_ref* tag. Packets re-enter IP firewall processing at the next rule. If no tag is supplied, packets are discarded.

CONTROL MESSAGES

This node type supports only the generic control messages.

SHUTDOWN

This node shuts down upon receipt of a NGM_SHUTDOWN control message. Do not do this, since the new **ipfw** node can only be created by reloading the **ng_ipfw** module.

SEE ALSO

ipfw(4), netgraph(4), ipfw(8), mbuf_tags(9)

HISTORY

The **ipfw** node type was implemented in FreeBSD 6.0.

AUTHORS

The **ipfw** node was written by Gleb Smirnoff <*glebius@FreeBSD.org*>.