NAME
   gif - generic tunnel interface

SYNOPSIS
   device gif

DESCRIPTION
   The **gif** interface is a generic tunnelling device for IPv4 and IPv6. It can tunnel IPv[46] traffic over
   IPv[46]. Therefore, there can be four possible configurations. The behavior of **gif** is mainly based on
   RFC2893 IPv6-over-IPv4 configured tunnel. On NetBSD, **gif** can also tunnel ISO traffic over IPv[46]
   using EON encapsulation. Note that **gif** does not perform GRE encapsulation; use gre(4) for GRE
   encapsulation.

   Each **gif** interface is created at runtime using interface cloning. This is most easily done with the
   "**ifconfig create**" command or using the *ifconfig_<interface>* variable in rc.conf(5).

   To use **gif**, the administrator needs to configure the protocol and addresses used for the outer header.
   This can be done by using ifconfig(8) **tunnel**, or SIOCSIFPHYADDR ioctl. The administrator also
   needs to configure the protocol and addresses for the inner header, with ifconfig(8). Note that IPv6 link-
   local addresses (those that start with fe80::) will be automatically configured whenever possible. You
   may need to remove IPv6 link-local addresses manually using ifconfig(8), if you want to disable the use
   of IPv6 as the inner header (for example, if you need a pure IPv4-over-IPv6 tunnel). Finally, you must
   modify the routing table to route the packets through the **gif** interface.

   The **gif** device can be configured to be ECN friendly. This can be configured by IFF_LINK1.

  ECN friendly behavior
   The **gif** device can be configured to be ECN friendly, as described in draft-ietf-ipsec-ecn-02.txt. This is
   turned off by default, and can be turned on by the IFF_LINK1 interface flag.

   Without IFF_LINK1, **gif** will show normal behavior, as described in RFC2893. This can be
   summarized as follows:

        Ingress Set outer TOS bit to 0.

        Egress   Drop outer TOS bit.

   With IFF_LINK1, **gif** will copy ECN bits (0x02 and 0x01 on IPv4 TOS byte or IPv6 traffic class byte)
   on egress and ingress, as follows:

Ingress Copy TOS bits except for ECN CE (masked with 0xfe) from inner to outer. Set ECN CE bit to 0.

Egress   Use inner TOS bits with some change. If outer ECN CE bit is 1, enable ECN CE bit on the inner.

Note that the ECN friendly behavior violates RFC2893. This should be used in mutual agreement with the peer.

Security

A malicious party may try to circumvent security filters by using tunnelled packets. For better protection, **gif** performs both martian and ingress filtering against the outer source address on egress. Note that martian/ingress filters are in no way complete. You may want to secure your node by using packet filters. Ingress filtering can break tunnel operation in an asymmetrically routed network. It can be turned off by IFF_LINK2 bit.

Miscellaneous

By default, **gif** tunnels may not be nested. This behavior may be modified at runtime by setting the sysctl(8) variable *net.link.gif.max_nesting* to the desired level of nesting.

SEE ALSO

gre(4), inet(4), inet6(4), ifconfig(8)

R. Gilligan and E. Nordmark, "Transition Mechanisms for IPv6 Hosts and Routers", *RFC2893*, http://tools.ietf.org/html/rfc2893, August 2000.

Sally Floyd, David L. Black, and K. K. Ramakrishnan, *IPsec Interactions with ECN*, December 1999, draft-ietf-ipsec-ecn-02.txt.

HISTORY

The **gif** device first appeared in the WIDE hydrangea IPv6 kit.

BUGS

There are many tunnelling protocol specifications, all defined differently from each other. The **gif** device may not interoperate with peers which are based on different specifications, and are picky about outer header fields. For example, you cannot usually use **gif** to talk with IPsec devices that use IPsec tunnel mode.

If the outer protocol is IPv4, **gif** does not try to perform path MTU discovery for the encapsulated packet (DF bit is set to 0).

If the outer protocol is IPv6, path MTU discovery for encapsulated packets may affect communication over the interface. The first bigger-than-pmtu packet may be lost. To avoid the problem, you may want to set the interface MTU for **gif** to 1240 or smaller, when the outer header is IPv6 and the inner header is IPv4.

The **gif** device does not translate ICMP messages for the outer header into the inner header.

In the past, **gif** had a multi-destination behavior, configurable via IFF_LINK0 flag. The behavior is obsolete and is no longer supported.