

NAME

gre - encapsulating network device

SYNOPSIS

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

```
device gre
```

Alternatively, to load the driver as a module at boot time, place the following line in loader.conf(5):

```
if_gre_load="YES"
```

DESCRIPTION

The **gre** network interface pseudo device encapsulates datagrams into IP. These encapsulated datagrams are routed to a destination host, where they are decapsulated and further routed to their final destination. The "tunnel" appears to the inner datagrams as one hop.

gre interfaces are dynamically created and destroyed with the ifconfig(8) **create** and **destroy** subcommands.

This driver corresponds to RFC 2784. Encapsulated datagrams are prepended an outer datagram and a GRE header. The GRE header specifies the type of the encapsulated datagram and thus allows for tunneling other protocols than IP. GRE mode is also the default tunnel mode on Cisco routers. **gre** also supports Cisco WCCP protocol, both version 1 and version 2.

The **gre** interfaces support a number of additional parameters to the ifconfig(8):

grekey Set the GRE key used for outgoing packets. A value of 0 disables the key option.

enable_csum Enables checksum calculation for outgoing packets.

enable_seq Enables use of sequence number field in the GRE header for outgoing packets.

udpencap Enables UDP-in-GRE encapsulation (see the *GRE-IN-UDP ENCAPSULATION* Section below for details).

udpport Set the source UDP port for outgoing packets. A value of 0 disables the persistence of source UDP port for outgoing packets. See the *GRE-IN-UDP ENCAPSULATION* Section below for details.

GRE-IN-UDP ENCAPSULATION

The `gre` supports GRE in UDP encapsulation as defined in RFC 8086. A GRE in UDP tunnel offers the possibility of better performance for load-balancing GRE traffic in transit networks. Encapsulating GRE in UDP enables use of the UDP source port to provide entropy to ECMP hashing.

The GRE in UDP tunnel uses single value 4754 as UDP destination port. The UDP source port contains a 14-bit entropy value that is generated by the encapsulator to identify a flow for the encapsulated packet. The `udpport` option can be used to disable this behaviour and use single source UDP port value. The value of `udpport` should be within the ephemeral port range, i.e., 49152 to 65535 by default.

Note that a GRE in UDP tunnel is unidirectional; the tunnel traffic is not expected to be returned back to the UDP source port values used to generate entropy. This may impact NAPT (Network Address Port Translator) middleboxes. If such tunnels are expected to be used on a path with a middlebox, the tunnel can be configured either to disable use of the UDP source port for entropy or to enable middleboxes to pass packets with UDP source port entropy.

EXAMPLES

```

192.168.1.* --- Router A -----tunnel----- Router B --- 192.168.2.*
      \                               /
      \                               /
      +----- the Internet -----+

```

Assuming router A has the (external) IP address A and the internal address 192.168.1.1, while router B has external address B and internal address 192.168.2.1, the following commands will configure the tunnel:

On router A:

```

ifconfig greN create
ifconfig greN inet 192.168.1.1 192.168.2.1
ifconfig greN inet tunnel A B
route add -net 192.168.2 -netmask 255.255.255.0 192.168.2.1

```

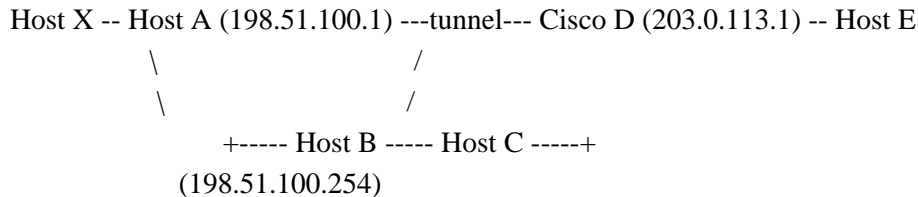
On router B:

```

ifconfig greN create
ifconfig greN inet 192.168.2.1 192.168.1.1
ifconfig greN inet tunnel B A
route add -net 192.168.1 -netmask 255.255.255.0 192.168.1.1

```

In case when internal and external IP addresses are the same, different routing tables (FIB) should be used. The default FIB will be applied to IP packets before GRE encapsulation. After encapsulation GRE interface should set different FIB number to outgoing packet. Then different FIB will be applied to such encapsulated packets. According to this FIB packet should be routed to tunnel endpoint.



On Host A (FreeBSD):

First of multiple FIBs should be configured via loader.conf:

```

net.fibs=2
net.add_addr_allfibs=0

```

Then routes to the gateway and remote tunnel endpoint via this gateway should be added to the second FIB:

```

route add -net 198.51.100.0 -netmask 255.255.255.0 -fib 1 -iface em0
route add -host 203.0.113.1 -fib 1 198.51.100.254

```

And GRE tunnel should be configured to change FIB for encapsulated packets:

```

ifconfig greN create
ifconfig greN inet 198.51.100.1 203.0.113.1
ifconfig greN inet tunnel 198.51.100.1 203.0.113.1 tunnefib 1

```

NOTES

The MTU of **gre** interfaces is set to 1476 by default, to match the value used by Cisco routers. This may not be an optimal value, depending on the link between the two tunnel endpoints. It can be adjusted via `ifconfig(8)`.

For correct operation, the **gre** device needs a route to the decapsulating host that does not run over the tunnel, as this would be a loop.

The kernel must be set to forward datagrams by setting the `net.inet.ip.forwarding` `sysctl(8)` variable to non-zero.

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

SEE ALSO

`gif(4)`, `inet(4)`, `ip(4)`, `me(4)`, `netintro(4)`, `protocols(5)`, `ifconfig(8)`, `sysctl(8)`

STANDARDS

S. Hanks, T. Li, D. Farinacci, and P. Traina, *Generic Routing Encapsulation (GRE)*, RFC 1701, October 1994.

S. Hanks, T. Li, D. Farinacci, and P. Traina, *Generic Routing Encapsulation over IPv4 networks*, RFC 1702, October 1994.

D. Farinacci, T. Li, S. Hanks, D. Meyer, and P. Traina, *Generic Routing Encapsulation (GRE)*, RFC 2784, March 2000.

G. Dommety, *Key and Sequence Number Extensions to GRE*, RFC 2890, September 2000.

AUTHORS

Andrey V. Elsukov <ae@FreeBSD.org>

Heiko W.Rupp <hwr@pilhuhn.de>

BUGS

The current implementation uses the key only for outgoing packets. Incoming packets with a different key or without a key will be treated as if they would belong to this interface.

The sequence number field also used only for outgoing packets.