

NAME

epair - A pair of virtual back-to-back connected Ethernet interfaces

SYNOPSIS

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

```
device epair
```

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

```
if_epair_load="YES"
```

DESCRIPTION

The **epair** is a pair of Ethernet-like software interfaces, which are connected back-to-back with a virtual cross-over cable.

Each **epair** interface pair is created at runtime using interface cloning. This is most easily done with the `ifconfig(8)` **create** command or using the `cloned_interfaces` variable in `rc.conf(5)`. While for cloning you only give either `epair` or `epair<n>` the **epair** pair will be named like `epair<n>[ab]`. This means the names of the first **epair** interfaces will be `epair0a` and `epair0b`.

Like any other Ethernet interface, an **epair** needs to have a network address. Each **epair** will be assigned a locally administered address by default, that is only guaranteed to be unique within one network stack. To change the default addresses one may use the `SIOCSIFADDR` `ioctl(2)` or `ifconfig(8)` utility.

The basic intent is to provide connectivity between two virtual network stack instances. When connected to an `if_bridge(4)`, one end of the interface pair can also be part of another (virtual) LAN. As with any other Ethernet interface, **epair** can have a `vlan(4)` configured on top of it.

SEE ALSO

`ioctl(2)`, `altq(4)`, `bpf(4)`, `if_bridge(4)`, `vlan(4)`, `loader.conf(5)`, `rc.conf(5)`, `ifconfig(8)`

HISTORY

The **epair** interface first appeared in FreeBSD 8.0.

AUTHORS

The **epair** interface was written by Bjoern A. Zeeb, CK Software GmbH, under sponsorship from the FreeBSD Foundation.