NAME

pcap_inject, pcap_sendpacket - transmit a packet

SYNOPSIS

#include <pcap/pcap.h>

int pcap_inject(pcap_t *p, const void *buf, size_t size);
int pcap_sendpacket(pcap_t *p, const u_char *buf, int size);

DESCRIPTION

pcap_inject() sends a raw packet through the network interface; *buf* points to the data of the packet, including the link-layer header, and *size* is the number of bytes in the packet.

Note that, even if you successfully open the network interface, you might not have permission to send packets on it, or it might not support sending packets; as **pcap_open_live**(3) doesn't have a flag to indicate whether to open for capturing, sending, or capturing and sending, you cannot request an open that supports sending and be notified at open time whether sending will be possible. Note also that some devices might not support sending packets.

Note that, on some platforms, the link-layer header of the packet that's sent might not be the same as the link-layer header of the packet supplied to **pcap_inject**(), as the source link-layer address, if the header contains such an address, might be changed to be the address assigned to the interface on which the packet it sent, if the platform doesn't support sending completely raw and unchanged packets. Even worse, some drivers on some platforms might change the link-layer type field to whatever value libpcap used when attaching to the device, even on platforms that *do* nominally support sending completely raw and unchanged packets.

pcap_sendpacket() is like **pcap_inject**(), but it returns **0** on success, rather than returning the number of bytes written. (**pcap_inject**() comes from OpenBSD; **pcap_sendpacket**() comes from WinPcap/Npcap. Both are provided for compatibility.)

RETURN VALUE

pcap_inject() returns the number of bytes written on success, **PCAP_ERROR_NOT_ACTIVATED** if called on a capture handle that has been created but not activated, and **PCAP_ERROR** on other errors.

pcap_sendpacket() returns **0** on success, **PCAP_ERROR_NOT_ACTIVATED** if called on a capture handle that has been created but not activated, and **PCAP_ERROR** on other errors.

If **PCAP_ERROR** is returned, **pcap_geterr**(3) or **pcap_perror**(3) may be called with *p* as an argument to fetch or display the error text.

SEE ALSO

pcap(3)