NAME

pcap_get_selectable_fd − get a file descriptor on which a select() can be done for a live capture

SYNOPSIS

#include <pcap/pcap.h>

int pcap_get_selectable_fd(pcap_t *p);

DESCRIPTION

pcap_get_selectable_fd() returns, on UNIX, a file descriptor number for a file descriptor on which one can do a **select**(2), **poll**(2), **epoll_wait**(2), **kevent**(2), or other such call to wait for it to be possible to read packets without blocking, if such a descriptor exists, or **−1**, if no such descriptor exists.

Some network devices opened with **pcap_create**(3PCAP) and **pcap_activate**(3PCAP), or with **pcap_open_live**(3PCAP), do not support those calls (for example, regular network devices on FreeBSD 4.3 and 4.4, and Endace DAG devices), so **−1** is returned for those devices. In that case, those calls must be given a timeout less than or equal to the timeout returned by **pcap_get_required_select_timeout**(3PCAP) for the device for which **pcap_get_selectable_fd**() returned **−1**, the device must be put in non-blocking mode with a call to **pcap_setnonblock**(3PCAP), and an attempt must always be made to read packets from the device when the call returns. If **pcap_get_required_select_timeout**() returns **NULL**, it is not possible to wait for packets to arrive on the device in an event loop.

Note that a device on which a read can be done without blocking may, on some platforms, not have any packets to read if the packet buffer timeout has expired. A call to**pcap_dispatch**(3PCAP) or **pcap_next_ex**(3PCAP) will return 0 in this case, but will not block.

Note that in:

FreeBSD prior to FreeBSD 4.6;

NetBSD prior to NetBSD 3.0;

OpenBSD prior to OpenBSD 2.4;

Mac OS X prior to Mac OS X 10.7;

select(), **poll**(), and **kevent**() do not work correctly on BPF devices; **pcap_get_selectable_fd**() will return a file descriptor on most of those versions (the exceptions being FreeBSD 4.3 and 4.4), but a simple **select**(), **poll**(), or **kevent**() call will not indicate that the descriptor is readable until a full buffer's worth of packets is received, even if the packet timeout expires before then. To work around this, code that uses those calls to wait for packets to arrive must put the **pcap_t** in non-blocking mode, and must arrange that the call have a timeout less than or equal to the packet buffer timeout, and must try to read packets after that timeout expires, regardless of whether the call indicated that the file descriptor for the **pcap_t** is ready to be read or not. (That workaround will not work in FreeBSD 4.3 and later; however, in FreeBSD 4.6 and later, those calls work correctly on BPF devices, so the workaround isn't necessary, although it does no harm.)

Note also that **poll**() and **kevent**() doesn't work on character special files, including BPF devices, in Mac OS X 10.4 and 10.5, so, while **select**() can be used on the descriptor returned by **pcap_get_selectable_fd**(), **poll**() and **kevent**() cannot be used on it those versions of Mac OS X. **poll**(), b ut not **kevent**(), works on that descriptor in Mac OS X releases prior to 10.4; **poll**() and **kevent**() work on that descriptor in Mac OS X 10.6 and later.

pcap_get_selectable_fd() is not available on Windows.

RETURN VALUE

A selectable file descriptor is returned if one exists; otherwise, **−1** is returned.

SEE ALSO

pcap(3PCAP), **kqueue**(2)