NAME

pcap_compile − compile a filter expression

SYNOPSIS

#include <pcap/pcap.h>

int pcap_compile(pcap_t *p, struct bpf_program *fp,
const char *str, int optimize, bpf_u_int32 netmask);

DESCRIPTION

pcap_compile() is used to compile the string *str* into a filter program. See **pcap-filter**(7) for the syntax of that string. *program* is a pointer to a *bpf_program* struct and is filled in by **pcap_compile**(). *optimize* controls whether optimization on the resulting code is performed. *netmask* specifies the IPv4 netmask of the network on which packets are being captured; it is used only when checking for IPv4 broadcast addresses in the filter program. If the netmask of the network on which packets are being captured isn't known to the program, or if packets are being captured on the Linux "any" pseudo-interface that can capture on more than one network, a value of **PCAP_NETMASK_UNKNOWN** can be supplied; tests for IPv4 broadcast addresses will fail to compile, but all other tests in the filter program will be OK.

NOTE: in libpcap 1.8.0 and later, **pcap_compile**() can be used in multiple threads within a single process. However, in earlier versions of libpcap, it is *not* safe to use **pcap_compile**() in multiple threads in a single process without some form of mutual exclusion allowing only one thread to call it at any given time.

RETURN VALUE

pcap_compile() returns **0** on success and **PCAP_ERROR** on failure. If **PCAP_ERROR** is returned, **pcap_geterr**(3PCAP) or **pcap_perror**(3PCAP) may be called with *p* as an argument to fetch or display the error text.

BACKWARD COMPATIBILITY

The **PCAP_NETMASK_UNKNOWN** constant became available in libpcap release 1.1.0.

SEE ALSO

pcap(3PCAP), **pcap_setfilter**(3PCAP), **pcap_freecode**(3PCAP)