

NAME

icmp - Internet Control Message Protocol

SYNOPSIS

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
```

int

```
socket(AF_INET, SOCK_RAW, proto);
```

DESCRIPTION

ICMP is the error and control message protocol used by IP and the Internet protocol family. It may be accessed through a "raw socket" for network monitoring and diagnostic functions. The *proto* parameter to the socket call to create an ICMP socket is obtained from `getprotobyname(3)`. ICMP sockets are connectionless, and are normally used with the `sendto(2)` and `recvfrom(2)` calls, though the `connect(2)` call may also be used to fix the destination for future packets (in which case the `read(2)` or `recv(2)` and `write(2)` or `send(2)` system calls may be used).

Outgoing packets automatically have an IP header prepended to them (based on the destination address). Incoming packets are received with the IP header and options intact.

Types

ICMP messages are classified according to the type and code fields present in the ICMP header. The abbreviations for the types and codes may be used in rules in `pf.conf(5)`. The following types are defined:

Num	Abbrev.	Description
0	echorep	Echo reply
3	unreach	Destination unreachable
4	squench	Packet loss, slow down
5	redir	Shorter route exists
6	alhost	Alternate host address
8	echoreq	Echo request
9	routeradv	Router advertisement
10	routersol	Router solicitation
11	timex	Time exceeded
12	paramprob	Invalid IP header
13	timereq	Timestamp request

14	timerep	Timestamp reply
15	inforeq	Information request
16	inforep	Information reply
17	maskreq	Address mask request
18	maskrep	Address mask reply
30	trace	Traceroute
31	dataconv	Data conversion problem
32	mobredir	Mobile host redirection
33	ipv6-where	IPv6 where-are-you
34	ipv6-here	IPv6 i-am-here
35	mobregreq	Mobile registration request
36	mobregrep	Mobile registration reply
39	skip	SKIP
40	photuris	Photuris

The following codes are defined:

Num	Abbrev.	Type	Description
0	net-unr	unreach	Network unreachable
1	host-unr	unreach	Host unreachable
2	proto-unr	unreach	Protocol unreachable
3	port-unr	unreach	Port unreachable
4	needfrag	unreach	Fragmentation needed but DF bit set
5	srcfail	unreach	Source routing failed
6	net-unk	unreach	Network unknown
7	host-unk	unreach	Host unknown
8	isolate	unreach	Host isolated
9	net-prohib	unreach	Network administratively prohibited
10	host-prohib	unreach	Host administratively prohibited
11	net-tos	unreach	Invalid TOS for network
12	host-tos	unreach	Invalid TOS for host
13	filter-prohib	unreach	Prohibited access
14	host-preced	unreach	Precedence violation
15	cutoff-preced	unreach	Precedence cutoff
0	redir-net	redir	Shorter route for network
1	redir-host	redir	Shorter route for host
2	redir-tos-net	redir	Shorter route for TOS and network
3	redir-tos-host	redir	Shorter route for TOS and host
0	normal-adv	routervadv	Normal advertisement

16	common-adv	routeradv	Selective advertisement
0	transit	timex	Time exceeded in transit
1	reassemb	timex	Time exceeded in reassembly
0	badhead	paramprob	Invalid option pointer
1	optmiss	paramprob	Missing option
2	badlen	paramprob	Invalid length
1	unknown-ind	photuris	Unknown security index
2	auth-fail	photuris	Authentication failed
3	decrypt-fail	photuris	Decryption failed

MIB (sysctl) Variables

The ICMP protocol implements a number of variables in the *net.inet.icmp* branch of the sysctl(3) MIB, which can also be read or modified with sysctl(8).

bmcastecho (boolean) Enable/disable ICMP replies received via broadcast or multicast. Defaults to false.

drop_redirect (boolean) Enable/disable dropping of ICMP Redirect packets. Defaults to false.

icmplim (integer) Bandwidth limit for ICMP replies in packets/second. If set to zero, no limiting will occur. Defaults to 200.

icmplim_output (boolean) Enable/disable logging of ICMP replies bandwidth limiting. Defaults to true.

log_redirect (boolean) Enable/disable logging of ICMP Redirect packets. Defaults to false.

maskfake (unsigned integer) When *maskrepl* is set and this value is non-zero, it will be used instead of the real address mask when the system replies to an ICMP Address Mask Request packet. Defaults to 0.

maskrepl (boolean) Enable/disable replies to ICMP Address Mask Request packets. Defaults to false.

quotelen (integer) Number of bytes from original packet to quote in ICMP reply. This number is internally enforced to be at least 8 bytes (per RFC792) and at most the maximal space left in the ICMP reply mbuf.

redirtimeout (integer) Delay in seconds before expiring route created by ICMP redirect.

reply_from_interface

(*boolean*) Use the IP address of the interface the packet came in through for responses to packets which are not directly addressed to us. If enabled, this rule is processed before all others. By default, continue with normal source selection. Enabling this option is particularly useful on routers because it makes external traceroutes show the actual path a packet has taken instead of the possibly different return path.

reply_src (*str*) An interface name used for the ICMP reply source in response to packets which are not directly addressed to us. By default continue with normal source selection.

tstamprepl (*boolean*) Enable/disable replies to ICMP Timestamp packets. Defaults to true.

ERRORS

A socket operation may fail with one of the following errors returned:

[EISCONN] when trying to establish a connection on a socket which already has one, or when trying to send a datagram with the destination address specified and the socket is already connected;

[ENOTCONN] when trying to send a datagram, but no destination address is specified, and the socket has not been connected;

[ENOBUFS] when the system runs out of memory for an internal data structure;

[EADDRNOTAVAIL] when an attempt is made to create a socket with a network address for which no network interface exists.

SEE ALSO

recv(2), send(2), sysctl(3), inet(4), intro(4), ip(4), pf.conf(5), sysctl(8)

HISTORY

The **icmp** protocol implementation appeared in 4.2BSD.