NAME

ddp – Linux AppleTalk protocol implementation

SYNOPSIS

#include <sys/socket.h>
#include <netatalk/at.h>

ddp_socket **= socket(AF_APPLETALK, SOCK_DGRAM, 0);**
raw_socket **= socket(AF_APPLETALK, SOCK_RAW,** *protocol***);**

DESCRIPTION

Linux implements the AppleTalk protocols described in *Inside AppleTalk*. Only the DDP layer and AARP are present in the kernel. They are designed to be used via the **netatalk** protocol libraries. This page documents the interface for those who wish or need to use the DDP layer directly.

The communication between AppleTalk and the user program works using a BSD-compatible socket interface. For more information on sockets, see **socket**(7).

An AppleTalk socket is created by calling the **socket**(2) function with a **AF_APPLETALK** socket family argument. Valid socket types are **SOCK_DGRAM** to open a **ddp** socket or **SOCK_RAW** to open a **raw** socket. *protocol* is the AppleTalk protocol to be received or sent. For **SOCK_RAW** you must specify **ATPROTO_DDP**.

Raw sockets may be opened only by a process with effective user ID 0 or when the process has the **CAP_NET_RAW** capability.

Address format

An AppleTalk socket address is defined as a combination of a network number, a node number, and a port number.

```
struct at_addr {
    unsigned short s_net;
    unsigned char  s_node;
};

struct sockaddr_atalk {
    sa_family_t    sat_family;     /* address family */
    unsigned char  sat_port;       /* port */
    struct at_addr sat_addr;       /* net/node */
};
```

sat_family is always set to **AF_APPLETALK**. *sat_port* contains the port. The port numbers below 129 are known as *reserved ports*. Only processes with the effective user ID 0 or the **CAP_NET_BIND_SERVICE** capability may **bind**(2) to these sockets. *sat_addr* is the host address. The *net* member of *struct at_addr* contains the host network in network byte order. The value of **AT_ANYNET** is a wildcard and also implies "this network." The *node* member of *struct at_addr* contains the host node number. The value of **AT_ANYNODE** is a wildcard and also implies "this node." The value of **ATADDR_BCAST** is a link local broadcast address.

Socket options

No protocol-specific socket options are supported.

/proc interfaces

IP supports a set of */proc* interfaces to configure some global AppleTalk parameters. The parameters can be accessed by reading or writing files in the directory */proc/sys/net/atalk/*.

aarp-expiry-time
The time interval (in seconds) before an AARP cache entry expires.

aarp-resolve-time
The time interval (in seconds) before an AARP cache entry is resolved.

aarp-retransmit-limit
The number of retransmissions of an AARP query before the node is declared dead.

aarp-tick-time
The timer rate (in seconds) for the timer driving AARP.

The default values match the specification and should never need to be changed.

Ioctls
All ioctls described in **socket**(7) apply to DDP.

ERRORS

EACCES
The user tried to execute an operation without the necessary permissions. These include sending to a broadcast address without having the broadcast flag set, and trying to bind to a reserved port without effective user ID 0 or **CAP_NET_BIND_SERVICE**.

EADDRINUSE
Tried to bind to an address already in use.

EADDRNOTAVAIL
A nonexistent interface was requested or the requested source address was not local.

EAGAIN
Operation on a nonblocking socket would block.

EALREADY
A connection operation on a nonblocking socket is already in progress.

ECONNABORTED
A connection was closed during an **accept**(2).

EHOSTUNREACH
No routing table entry matches the destination address.

EINVAL
Invalid argument passed.

EISCONN
connect(2) was called on an already connected socket.

EMSGSIZE
Datagram is bigger than the DDP MTU.

ENODEV
Network device not available or not capable of sending IP.

ENOENT
SIOCGSTAMP was called on a socket where no packet arrived.

ENOMEM and **ENOBUFS**
Not enough memory available.

ENOPKG
A kernel subsystem was not configured.

ENOPROTOOPT and **EOPNOTSUPP**
Invalid socket option passed.

ENOTCONN
The operation is defined only on a connected socket, but the socket wasn't connected.

EPERM
User doesn't have permission to set high priority, make a configuration change, or send signals to the requested process or group.

EPIPE The connection was unexpectedly closed or shut down by the other end.

ESOCKTNOSUPPORT
The socket was unconfigured, or an unknown socket type was requested.

VERSIONS
AppleTalk is supported by Linux 2.0 or higher. The */proc* interfaces exist since Linux 2.2.

NOTES
Be very careful with the **SO_BROADCAST** option; it is not privileged in Linux. It is easy to overload the network with careless sending to broadcast addresses.

Compatibility
The basic AppleTalk socket interface is compatible with **netatalk** on BSD-derived systems. Many BSD systems fail to check **SO_BROADCAST** when sending broadcast frames; this can lead to compatibility problems.

The raw socket mode is unique to Linux and exists to support the alternative CAP package and AppleTalk monitoring tools more easily.

BUGS
There are too many inconsistent error values.

The ioctls used to configure routing tables, devices, AARP tables, and other devices are not yet described.

SEE ALSO
recvmsg(2), **sendmsg**(2), **capabilities**(7), **socket**(7)

COLOPHON
This page is part of release 5.02 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at https://www.kernel.org/doc/man−pages/.