

ip(7)

НАЗВАНИЕ

ip, **IP** - Internet Protocol

SYNOPSIS

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

s = socket(AF_INET, SOCK_RAW, proto);
t = t_open ("/dev/rawip", O_RDWR);
```

ОПИСАНИЕ

IP - это межсетевой протокол доставки дейтаграмм (internetwork datagram delivery protocol), основной в семействе протоколов Internet. Программы могут использовать протокол IP через протоколы более высокого уровня, такие как протокол управления передачей (Transmission Control Protocol - TCP) или протокол дейтаграмм пользователя (User Datagram Protocol - UDP), или взаимодействовать по протоколу IP непосредственно. См. страницы справочного руководства **tcp(7P)** и **udp(7P)**. Непосредственный доступ возможен через интерфейс сокетов, с использованием "непосредственных" ("raw") сокетов, или через интерфейс транспортного уровня (Transport Level Interface - "TLI"). Опции протокола, определенные в спецификации IP, можно устанавливать для исходящих дейтаграмм.

ФУНКЦИОНАЛЬНЫЙ ИНТЕРФЕЙС

Драйвер STREAMS **/dev/rawip** - это провайдер транспорта TLI, обеспечивающий непосредственный доступ к IP.

Непосредственные IP-сокеты не поддерживают постоянное подключение (connectionless) и обычно используются с помощью вызовов **sendto()** и **recvfrom()** (см. **send(3SOCKET)** и **recv(3SOCKET)**), хотя вызов **connect(3SOCKET)** тоже можно использовать для фиксации места назначения будущей дейтаграммы. В этом случае можно использовать вызовы **read(2)** или **recv(3SOCKET)** и **write(2)** или **send(3SOCKET)**. Если указан протокол **IPPROTO_RAW** или **IPPROTO_IGMP**, предполагается, что приложение включает при отправке полный IP-заголовок. Иначе в исходящих дейтаграммах будет установлен номер протокола, который будет использоваться для фильтрации входящих дейтаграмм, а IP-заголовок будет генерироваться и добавляться перед каждой исходящей дейтаграммой. В любом случае, IP-заголовок и опции входящих дейтаграмм не изменяются.

На уровне IP поддерживаются следующие опции сокетов:

IP_OPTIONS

Опции протокола IP для исходящих дейтаграмм. Эта опция сокета может использоваться для установки опций IP, которые надо включать в каждую исходящую дейтаграмму. Опции IP, которые необходимо отправлять, устанавливаются с помощью функции **setsockopt()** (см. **getsockopt(3SOCKET)**). Вызов **getsockopt(3SOCKET)** возвращает опции IP, установленные с помощью

последнего вызова **setsockopt()**. Опции IP полученных дейтаграмм видны пользовательским программам только при использовании непосредственных сокетов IP. Формат опций IP при задании в функции **setsockopt()** совпадает с указанным в спецификации протокола IP за исключением того, что в списке адресов для опций встроенной маршрутизации (source routing options) адрес первого ближайшего шлюза (first-hop gateway) должен указываться первым в списке шлюзов. Адрес первого ближайшего шлюза будет выбираться из списка опций с соответствующим изменением размера списка перед его использованием. Опции IP можно задавать для любого типа сокета в семействе Internet.

IP_SEC_OPT

Включить или получить установки защиты IPsec для данного сокета. Подробнее о службах защиты IPsec см. на странице справочного руководства **ipsec(7P)**.

IP_ADD_MEMBERSHIP

Войти в группу многоадресной передачи.

IP_DROP_MEMBERSHIP

Выйти из группы многоадресной передачи.

Эти опции принимают в качестве параметра структуру **struct ip_mreq**. Эта структура содержит адрес для многоадресной передачи (multicast address), который необходимо задавать как адрес класса D для многоадресной передачи IP, и адрес сетевого интерфейса. Обычно в качестве адреса сетевого интерфейса задается значение **INADDR_ANY**, что позволяет ядру выбрать интерфейс для вхождения в группу.

IP_MULTICAST_IF

Исходящий интерфейс для многоадресных пакетов. Эта опция требует передачи в качестве аргумента структуры **in_addr** и выбирает задаваемый ею интерфейс для передачи исходящих многоадресных пакетов IP. Если указан адрес **INADDR_ANY**, для выбора исходящего интерфейса (как и принято по умолчанию) будет использоваться обычная таблица маршрутизации.

IP_MULTICAST_TTL

Время жизни для многоадресных дейтаграмм. Эта опция принимает в качестве аргумента **unsigned char**. Его значение используется как время жизни (TTL), которое протокол IP будет использовать для исходящих многоадресных дейтаграмм. Стандартное значение - 1.

IP_MULTICAST_LOOP

Интерфейс закольцовывания для многоадресных дейтаграмм. Обычно многоадресные дейтаграммы доставляются членам группы на посылающем хосте. При установке аргументу типа **unsigned char** значения 0 они перестают доставляться.

Опции многоадресного сокета могут задаваться для дейтаграмм любого типа сокетов в семействе Internet.

На уровне сокета можно задавать опцию сокета **SO_DONTROUTE**. Эта опция приводит к тому, что посылаемые дейтаграммы не маршрутизируются и не перенаправляются, поскольку устанавливает полю время жизни IP (Time To Live) значение 1, - такой пакет не будет перенаправляться маршрутизаторами.

Непосредственные дейтаграммы IP можно также посылать и принимать с помощью базовых функций TLI, не поддерживающих соединение (connectionless primitives).

Дейтаграммы проходят через уровень IP в двух направлениях: вверх из сети до пользовательских процессов и вниз от пользовательских процессов в сеть. С учетом этой ориентации, протокол IP располагается над драйверами сетевых интерфейсов и под транспортными протоколами UDP и TCP. Протокол управляющих сообщений Internet (Internet Control Message Protocol - ICMP) логически является частью протокола IP. См. **icmp(7P)**.

Протокол IP включает контрольную сумму для заголовка, но не данных в дейтаграмме. Значение контрольной суммы вычисляется и устанавливается в процессе отправки дейтаграмм, а проверяется - при их получении.

Опции IP в полученных дейтаграммах обрабатываются на уровне IP в соответствии со спецификацией протокола. В настоящее время распознаются следующие опции IP: защита (security), ненадежный источник и внутренняя маршрутизация (loose source and record route - LSRR), надежный источник и внутренняя маршрутизация (strict source and record route - SSRR), внутренняя маршрутизация (record route) и временная отметка (internet timestamp).

Уровень IP обычно действует как маршрутизатор (помимо прочего, перенаправляющий дейтаграммы, не адресованные ему), если машина имеет два или более работающих интерфейса. Это поведение можно изменить с помощью утилиты **ndd(1M)**, устанавливая значение переменной **/dev/ip, ip_forwarding**. 0 означает не перенаправлять; 1 - перенаправлять. Сценарии инициализации (см. **/etc/init.d/inetinit**) устанавливают это значение при загрузке в зависимости от количества работающих ("up") интерфейсов, но они не будут включать перенаправление IP вообще, если существует файл **/etc/notrouter**. При загрузке модуля IP переменная **ip_forwarding** имеет значение 0 и оно не изменяется, если:

- работает только один интерфейс, управляемый не по DHCP (это - самый типичный случай)
- существует файл **/etc/notrouter** и протокол DHCP не требует включить перенаправление IP
- файл **/etc/defaultrouter** существует и протокол DHCP не требует включить перенаправление IP. В противном случае, переменная **ip_forwarding** получает значение 1.

Кроме того, на уровне IP можно управлять маршрутизацией более детально. Каждый интерфейс будет создавать переменную **<имя_интерфейса>:ip_forwarding /dev/ip**, которую можно изменять с помощью **ndd(1M)**. Если специфическая переменная **:ip_forwarding** интерфейса получила значение 0, пакеты с этого интерфейса на другие и с других интерфейсов на него перенаправляться не будут.

При установке значения переменной **ip_forwarding** все специфические переменные **:ip_forwarding** интерфейсов получают значение, соответствующее **ip_forwarding**.

Уровень IP будет посылать сообщение ICMP обратно на исходный хост во многих случаях, если получает дейтаграмму, которую нельзя обработать. Сообщение ICMP "time exceeded" (устарела) будет посылаться, если поле "time to live" (время жизни) в заголовке уменьшилось до нуля в процессе перенаправления дейтаграммы. Сообщение "destination unreachable" (пункт назначения недостижим) будет посылаться, если дейтаграмма не может быть перенаправлена, поскольку нет маршрута к конечному пункту назначения, или ее нельзя фрагментировать. Если дейтаграмма адресована локальному хосту, но предназначена неподдерживаемому протоколу или неиспользуемому порту, сообщение о недостижимости пункта назначения тоже посылается. Уровень IP может посылать сообщение ICMP "source quench", если он слишком быстро получает дейтаграммы. Сообщения ICMP посылаются только для первого фрагмента фрагментированной дейтаграммы и никогда не возвращаются в ответ на ошибки в других сообщениях ICMP.

Уровень IP поддерживает фрагментацию и обратную сборку фрагментов. Дейтаграммы фрагментируются при выдаче, если их размер превышает размер *максимальной единицы передачи* (maximum transmission unit - MTU) сетевого интерфейса. Фрагменты полученных дейтаграмм удаляются из очередей на сборку, если полную дейтаграмму не удастся восстановить за короткий промежуток времени.

Ошибки при посылке, выявленные на уровне драйвера сетевого интерфейса, возвращаются уровнем IP пользовательскому процессу.

ССЫЛКИ

[ndd\(1M\)](#), [read\(2\)](#), [write\(2\)](#), [bind\(3SOCKET\)](#), [connect\(3SOCKET\)](#), [getsockopt\(3SOCKET\)](#), [recv\(3SOCKET\)](#), [send\(3SOCKET\)](#), [defaultrouter\(4\)](#), [icmp\(7P\)](#), [if_tcp\(7P\)](#), [inet\(7P\)](#), [ip6\(7P\)](#), [ipsec\(7P\)](#), [routing\(7P\)](#), [tcp\(7P\)](#), [udp\(7P\)](#)

Braden, R., RFC 1122, Requirements for Internet Hosts - Communication Layers, Information Sciences Institute, University of Southern California, October 1989.

Postel, J., RFC 791, Internet Protocol - DARPA Internet Program Protocol Specification, Information Sciences Institute, University of Southern California, September 1981.

ДИАГНОСТИКА

Операция с сокетом может закончиться неудачно с возвратом следующих сообщений об ошибках:

EACCES

Операцию **bind()** попытались выполнить с "зарезервированным" номером порта, а эффективный идентификатор пользователя процесса не соответствовал привилегированному пользователю.

EADDRINUSE

Операцию **bind()** попытались выполнить с сокетом, соответствующим паре сетевой адрес/порт, которая уже была привязана к другому сокету.

EADDRNOTAVAIL

Операцию **bind()** попытались выполнить с адресом, не сконфигурированным на этой машине.

EINVAL

Операцию **sendmsg()** попытались выполнить с непустым значением **msg_accrights**.

EINVAL

Была выполнена операция **getsockopt()** или **setsockopt()** с неизвестным именем опции сокета.

EINVAL

Операцию **getsockopt()** или **setsockopt()** попытались выполнить с неправильно сформированным полем опций IP; поле опций оказалось короче минимального значения или длинее, чем предоставленный буфер опции.

EISCONN

Операцию **connect()** попытались выполнить с сокетом, с которым операция **connect()** уже была выполнена, и перед выполнением нового подключения сокет не удалось успешно отключить.

EISCONN

С сокетом, с которым уже была выполнена операция **connect()**, попытались выполнить операцию **sendto()** или **sendmsg()**, задающую адрес, на который необходимо послать сообщение.

EMSGSIZE

Попытались выполнить операцию **send()**, **sendto()** или **sendmsg()** для посылки дейтаграммы, размер которой слишком велик для интерфейса, а фрагментация не разрешена (например, при широковещательной посылке).

ENETUNREACH

Попытались установить соединение с помощью **connect()** или послать дейтаграмму с помощью **sendto()** или **sendmsg()**, а соответствующей записи в таблице маршрутизации нет, или получено сообщение ICMP о недостижимости пункта назначения ("**destination unreachable**").

ENOTCONN

Попытались выполнить операцию **send()**, **write()**, **sendto()** или **sendmsg()** с сокетом, с которым еще не выполнена операция **connect()**, не задав адрес, на который необходимо послать сообщение.

ENOBUFS

Системе не хватило памяти под буферы фрагментации или для других внутренних структур данных.

ПРИМЕЧАНИЯ

Непосредственные (raw) сокеты, в принципе, должны поддерживать получение ошибок ICMP, относящихся к протоколу; в настоящее время такие пакеты просто отбрасываются.

Пользователи протоколов более высокого уровня, таких как TCP и UDP, должны иметь возможность просматривать полученные опции IP.

Последнее изменение: 10 ноября 1999 года

Copyright (no c) - Fuck copyright!, 2003 В. Кравчук, [OpenXS Initiative](#), перевод на русский язык