Реализация протокола IPv4 в ОС Linux поддерживает беспротокольные сокеты **raw** (SOCK_RAW). Эти сокеты позволяют реализовать в пользовательском пространстве новые протоколы стека Ipv4. Сокеты **raw** принимают и передают необработанные дейтаграммы без добавления в них каких-либо заголовков.

Уровень IPv4 генерирует заголовки IP для передаваемых пакетов, если для сокета не установлен флаг **IP_HDRINCL**, означающий, что пользовательская программа уже включила в данный пакет свой заголовок.

Беспротокольные сокеты могут использоваться только процессами имеющими флаг возможностей **CAP_NET_RAW** или запущенными от имени пользователя с эффективным UID=0 (root).

Сокету передаются все пакеты и сообщения об ошибках, соответствующие номеру протокола, заданному для этого сокета. Списки номеров протоколов можно найти в документе RFC 1700[1] или получить с помощью системного вызова **getprotobyname**.

Протокол **IPPROTO_RAW** предполагает наличие флага **IP_HDRINCL** и может передавать любые протоколы IP, указанные в полученном заголовке. Прием пакетов IP с использованием беспротокольных сокетов через **IPPROTO_RAW** невозможен ни для каких протоколов IP.

При установленном флаге **IP_HDRINCL** некоторые поля заголовков IP изменяются в соответствии с приведенной ниже таблицей.

Таблица 1 Изменение полей заголовка IP для пакетов RAW

Поле	Изменение
IP Checksum	Заполняется в соответствии с реальной контрольной суммой.
Source Address	Устанавливается нулевое значение.
Packet Id	Устанавливается нулевое значение.
Total Length	Заполняется в соответствии с реальным размером пакета.

Если задан флаг **IP_HDRINCL** и заголовок IP содержит ненулевой адрес получателя, тогда для маршрутизации пакета используется адрес получателя, заданный для сокета. При наличии флага **MSG_DONTROUTE** адрес получателя должен быть связан с локальным интерфейсом, иначе интерфейс будет определяться из таблицы маршрутов (при этом маршруты через шлюзы игнорируются).

Если флаг **IP_HDRINCL** не установлен, можно задать опции IP с использованием функции **setsockopt**.

В Linux все поля и опции заголовков IP можно устанавливать с использованием опций сокета

1

IP. Это означает, что сокеты типа **raw** обычно нужны только для новых протоколов или протоколов, не имеющих пользовательского интерфейса (например, ICMP).

При получении пакета всем сокетам **raw**, которые связаны с указанным в пакете протоколом, до передачи какому-либо иному протокольному модулю (например, модулю ядра).

Формат адреса

Сокеты **raw** используют для адресации стандартные структуры **sockaddr**, определенные для протокола IP (стр.). Поле **sin_port** можно использовать для указания номера протокола IP[2]. Во входящих пакетов в поле **sin_port** содержит номер протокола для данного пакета. Корректные номера протоколов IP можно найти в файле **<netinet/in.h>**.

Опции сокета

Опции беспротокольных сокетов можно устанавливать с помощью функции **setsockopt** и читать с помощью **getsockopt**, передавая этим функциям флаг семейства **SOL_RAW**.

Сокеты **raw** поддерживают все опции IP (семейство **SOL_IP**), которые пригодны для сокетов дейтаграмм. Кроме того, для беспротокольных сокетов поддерживается специальная опция **ICMP_FILTER**, которая включает специальный фильтр для raw-сокетов, связанных с протоколом **IPPROTO_ICMP**. Этот флаг устанавливается для всех типов сообщений ICMP, которые должны быть отфильтрованы. По умолчанию фильтрация ICMP отключена.

Замечания по использованию raw-сокетов

Сокеты **raw** фрагментируют пакеты, размер которых превышает значение MTU для интерфейса. Более эффективным решением является использование механизма определения MTU (Path MTU discovery) с помощью опции **IP_PMTU_DISCOVER**.

Сокет **raw** можно связать с локальным адресом, используя функцию **bind**. Если сокет не привязан к адресу, он будет получать все пакеты, заданного для сокета протокола IP. Беспротокольные сокеты можно так же привязать к указанному сетевому интерфейсу с помощью опции **SO_BINDTODEVICE**.

Сокет **IPPROTO_RAW** предназначен только для передачи пакетов. Если вы реально хотите принимать все пакеты IP, используйте сокет **packet** с протоколом **ETH_P_IP**. Отметим, что пакетные сокеты не собирают фрагментированные дейтаграммы IP в отличие от сокетов **raw**.

Если вы хотите принимать все пакеты ICMP для сокета дейтаграмм, зачастую будет удобней воспользоваться опцией **IP_RECVERR** для конкретного сокета.

Сокеты **raw** могут перехватывать пакеты всех протоколов IP в системе Linux, включая протоколы типа ICMP или TCP, имеющие в ядре собственные протокольные модули. В таких случаях пакеты передаются сразу модулю ядра и беспротокольному сокету. Такие возможности не следует использовать в переносимых программах, поскольку другие ОС (в частности, BSD) не могут работать в таком режиме.

Linux никогда не изменяет заголовки, передаваемые от пользователя, за исключением

2

заполнения некоторых полей при использовании опции IP_HDRINCL (см. табл. 1). Большинство реализаций беспротокольных сокетов в других ОС, изменяют пользовательские заголовки.

Сокеты **raw** в большинстве случаев плохо переносимы и их не следует применять в программах, предназначенных для других ОС.

Обработка ошибок

Приходящие из сети сообщения об ошибках передаются пользовательской программе только в тех случаях, когда сокет подключен или установлен флаг **IP_RECVERR**. Для подключенных сокетов в целях обеспечения совместимости передаются только сообщения **EMSGSIZE** и **EPROTO**. При установке флага **IP_RECVERR** все сетевые ошибки помещаются в очередь.

Коды ошибок

Таблица 2 Коды ошибок для сокетов raw

Код	Описание
EMSGSIZE	Размер пакета слишком велик. Такая ошибка может возникать при включенном режиме Path MTU Discovery (флаг IP_PMTU_DISCOVER, а также в тех случаях, когда размер пакета превышает максимальное значение для IPv4 (64 кбайт).
EACCES	Попытка пользователя передать пакет по широковещательному адресу при отсутствии у сокета флага широковещания.
EPROTO	Поступило сообщение ICMP об ошибке, вызванной некорректными параметрами.
EFAULT	Передан некорректный адрес блока памяти.
EOPNOTSUPP	При вызове функции **socket** передан некорректный флаг (например, MSG_OOB).
EINVAL	Некорректный параметр.
EPERM	Пользователь не имеет права открывать raw-сокеты. Такие права имеет пользователь с эффективным идентификатором UID=0 и приложения с установленным флагом возможностей **CAP_NET_RAW**.

Известные проблемы

1. Не описано расширение для прозрачного proxy.
2. При установке опции **IP_HDRINCL** дейтаграммы не будут фрагментироваться,

следовательно их размер ограничен значением MTU для интерфейса. Это ограничение присуще ядрам серии 2.2.

3. Начиная с ядрес серии 2.2 установка номера протокола IP в поле **sin_port** не поддерживается. Всегда используется протокол, с которым сокет связан, или протокол, заданный при первом вызове функции **socket**.

Этот документ изрядно устарел и более не будет обновляться, поэтому лучше обращаться к списку зарегистрированных протоколов на сайте www.iana.org/assignments/protocol-numbers

Это поле игнорируется в ядрах серии 2.2 и для него следует использовать значение 0.