

Network Working Group

Request For Comments: 826

David C. Plummer

(DCP@MIT-MC)

November 1982

An Ethernet Address Resolution Protocol

-- or --

Converting Network Protocol Addresses

to 48.bit Ethernet Address

for Transmission on

Ethernet Hardware

Протокол преобразования адресов Ethernet

Тезисы

Реализация протокола Р на передающем хосте S решает (с использованием механизма маршрутизации протокола Р), что следует передать информацию хосту Т, расположенному где-то в кабельном сегменте Ethernet 10 мбит/с. Для передачи реального пакета Ethernet требуется знать 48-битовый адрес Ethernet¹. Адрес хоста в представлении протокола Р не всегда совместим с соответствующим адресом Ethernet (различаться может как значение, так и размер адреса). В настоящем документе представлен протокол, позволяющий динамически находить информацию, требуемую для преобразования адреса А, используемого протоколом Р в 48-битовый адрес Ethernet.

Возможно обобщение протокола для использования в сетях, отличных от Ethernet 10 мбит/с. Примером таких сетей могут служить некоторые сети передачи данных по радиоканалам.

Предлагаемый здесь протокол является результатом обсуждения с несколькими специалистами, среди которых особенно важный вклад внесли J. Noel Chiappa, Yogen Dalal и James E. Kulp. Весьма полезные комментарии дал David Moon.

[Целью настоящего RFC является представление метода преобразования протокольных адресов (например, адресов IP) в адреса ЛВС (например, Ethernet). Этот вопрос в настоящее время представляет значительный интерес для сообщества ARPA Internet. Предложенный здесь метод представлен на ваше рассмотрение. Документ не содержит стандарта Internet².]

Примечания:

- 1) Протокол был изначально разработан для ЛВС DEC/Intel/Xerox Ethernet 10 Мбит/с. Впоследствии было сделано обобщение протокола для других типов сетей. Большая часть обсуждений в настоящем документе относится к сетям Ethernet 10 мбит/с. Обобщения, если они имеются, приводятся после обсуждения, связанного с Ethernet.
- 2) Протокол DOD Internet Protocol в документе обозначается как протокол Internet³.
- 3) Цифровые значения адресов указываются в соответствии со стандартом Ethernet – старший бит пишется слева. Такое написание отличается от принятого для PDP-11 и VAX написания адресов. Следовательно, при рассмотрении поля opcode (ar\$op) нужно обращать внимание на порядок битов. Распределение пространства адресов должно регулироваться специальной организацией.
- 4) До утверждения соответствующей организации запросы на выделение адресного пространства следует адресовать:

David C. Plummer

Symbolics, Inc.

243 Vassar Street

Cambridge, Massachusetts 02139

или отправлять электронной почтой по адресу DCP@MIT-MC⁴.

Описание задачи

Мир представляет собой джунгли и в сетевых играх принимает участие множество животных. Практически на каждом уровне сетевой архитектуры может использоваться множество протоколов. Например, на верхнем уровне могут использоваться протоколы TELNET и SUPDUP для удаленного доступа. На нижележащих уровнях используется потоковый протокол с гарантированной доставкой, в качестве которого может применяться протокол CHAOS, DOD TCP, Xerox BSP или DECnet. Ближе к аппаратному уровню на уровне логического транспорта может использоваться протокол CHAOS, DOD Internet, Xerox PUP или DECnet. Ethernet поддерживает сосуществование всех перечисленных здесь (и многих других) протоколов в одном кабеле,

¹ MAC-адрес в современной терминологии. *Прим. перев.*

² В настоящее время RFC 826 относится к числу стандартов Internet [STD 0037] *Прим. перев.*

³ В соответствии со сложившейся практикой в переводе этот протокол будет указываться как IP. *Прим. перев.*

⁴ В настоящее время распределением адресного пространства Ethernet занимается IEEE Registration Authority. *Прим. перев.*

используя для идентификации протокола поле типа в заголовке пакетов Ethernet. Однако Ethernet требует использования 48-битовой адресации устройств, подключенных к кабелю, тогда как большинство протоколов использует адреса других размеров и эти адреса могут быть не связаны с аппаратными адресами Ethernet. Например, протокол CHAOS использует 16-битовую адресацию, DOD Internet – 32-битовую, а Xerox PUP – 8-битовую. Поэтому требуется протокол, способный динамически распространять информацию о соответствии между протокольными адресами и адресами Ethernet.

Мотивация

Использование сетей Ethernet расширяется по мере роста числа производителей оборудования, соответствующего спецификации DIX. По мере роста числа устройств Ethernet увеличивается и число программ, работающих с этими интерфейсами. В такой ситуации возможны два варианта развития: (1) каждый разработчик использует собственный метод преобразования между протокольными и аппаратными адресами или (2) все разработчики используют единый стандарт, позволяющий распространять таблицы соответствия адресов без какого-либо преобразования. Данный документ является попыткой создания такого стандарта.

Определения

Определим значения, помещаемые в поле TYPE заголовков пакетов Ethernet:

```
ether_type$XEROX_PUP,
ether_type$DOD_INTERNET,
ether_type$CHAOS,
```

и дополнительно определим новый тип:

```
ether_type$ADDRESS_RESOLUTION.
```

Определим также следующие значения (они будут использоваться в дальнейшем обсуждении):

```
ares_op$REQUEST (= 1, если сначала передается старший бит),
ares_op$REPLY (= 2),
ares_hrd$Ethernet (= 1).
```

Формат пакета

Для обмена информацией о преобразовании между протокольными адресами и 48-битовыми адресами Ethernet требуется формат пакета, который будет при необходимости включать протокол преобразования адресов (Address Resolution). Формат такого пакета показан ниже.

Уровень передачи Ethernet (не обязательно доступный пользователю):

```
48-битовый Ethernet-адрес получателя
48-битовый Ethernet-адрес отправителя
16-битовое значение типа протокола = ether_type$ADDRESS_RESOLUTION
```

Поле данных пакета Ethernet :

```
16 битов: (ar$hrd) - пространство аппаратных адресов (например, Ethernet, Packet Radio Net и др.)
16 битов: (ar$pro) - пространство протокольных адресов. Для Ethernet это набор значений поля "тип" ether_type$<protocol>.
8 битов: (ar$hln) - размер каждого аппаратного адреса в байтах
8 битов: (ar$pln) - размер каждого протокольного адреса в байтах
16 битов: (ar$op) - код операции (ares_op$REQUEST | ares_op$REPLY)
n байтов: (ar$sha) – аппаратный адрес отправителя пакета (n берется из поля ar$hln).
m байтов: (ar$spa) – протокольный адрес отправителя пакета (m берется из поля ar$pln).
n байтов: (ar$tha) – аппаратный адрес получателя, если он известен.
m байтов: (ar$tpa) – протокольный адрес получателя.
```

Генерация пакетов

Когда пакет передается на нижние уровни, система маршрутизации определяет протокольный адрес следующего хопа для этого пакета и аппаратный интерфейс, через который осуществляется связь с адресатом, заданным протокольным адресом. В случае Ethernet преобразование, требуемое для одного из нижних уровней, осуществляется с использованием модуля преобразования адресов (возможно, реализованного в модуле поддержки Ethernet), который должен определить по паре <тип протокола, протокольный адрес получателя> 48-битовый Ethernet-адрес получателя. Модуль преобразования адресов пытается найти нужную пару в своей таблице. Если соответствующая пара найдена, адрес Ethernet возвращается вызвавшему модулю (драйверу устройства), который передает пакет. Если нужная пара отсутствует в таблице, генерируется пакет Ethernet с типом ether_type\$ADDRESS_RESOLUTION, а вызвавшему модулю может передаваться информация об отказе в передаче пакета (в предположении, что передача будет повторена на вышележащем уровне). Модуль преобразования адресов устанавливает ar\$hrd = ares_hrd\$Ethernet, в поле ar\$pro помещается тип протокола, для которого требуется преобразование, для поля ar\$hln устанавливается значение 6 (число байтов в аппаратном адресе Ethernet), в поле ar\$pln указывается размер адреса для данного протокола, устанавливается ar\$op = ares_op\$REQUEST, в поле ar\$sha указывается 48-битовый адрес Ethernet отправителя, в поле ar\$spa – протокольный адрес отправителя, а в поле ar\$tpa – протокольный адрес искомого адресата. Значение поля ar\$tha не задается, поскольку именно это значения пытаются определить. Можно указать в поле ar\$tha широкополосный аппаратный адрес (для Ethernet этот адрес содержит значение 1 во всех битах), если это удобно по каким-то соображениям, связанным с

данной реализацией. В таких случаях выполняется широковещательная передача пакета всем станциям сегмента Ethernet, определенного на этапе маршрутизации.

Прием пакета

При получении пакета преобразования адресов⁵ принявший пакет модуль Ethernet передает этот пакет модулю преобразования адресов, который реализует описанный ниже алгоритм. Отрицательный результат в приведенной иллюстрации алгоритма означает завершение обработки и отбрасывание пакета.

? У меня имеется аппаратный адрес, заданный полем ar\$hrd?

Да: (почти определенный результат)

[дополнительная проверка размера адреса - ar\$hln]

? Указанный в поле ar\$pro протокол поддерживается?

Да:

[дополнительная проверка размера адреса - ar\$pln]

Merge_flag := false

Если пара <тип протокола, протокольный адрес отправителя> уже присутствует в таблице преобразования, поле аппаратного адреса отправителя обновляется в соответствии с полученным пакетом и устанавливается Merge_flag = true.

? У меня имеется протокольный адрес получателя?

Да:

Если Merge_flag = false, в таблицу преобразования добавляется триплет <тип протокола, протокольный адрес отправителя, аппаратный адрес отправителя>.

? Код операции ares_op\$REQUEST? (смотрим код запрошенной операции!!)

Да:

Меняем местами протокольное и аппаратное поля, помещаем локальный протокольный и аппаратный адрес в соответствующие поля отправителя.

Устанавливаем ar\$op = ares_op\$REPLY

Передаем пакет с использованием аппаратного адреса отправителя в качестве адресата.

Отметим, что триплет <тип протокола, протокольный адрес отправителя, аппаратный адрес отправителя> включается в таблицу до просмотра кода операции. Это делается, исходя из предположения о наличии двухсторонней связи (если А имеет что-то сказать В, то В возможно тоже имеет что сказать А). Отметим также, что при наличии в таблице записи для пары <тип протокола, протокольный адрес отправителя> в ней просто изменяется соответствующим образом поле аппаратного адреса⁶.

Обобщение: Поля ar\$hrd и ar\$hln позволяют использовать описанный здесь протокол и формат пакетов для сетей, отличных от Ethernet. Для сетей Ethernet поля <ar\$hrd, ar\$hln> имеют значения <1, 6>. Для сетей других типов поле ar\$pro может не соответствовать полю Ethernet type, но это значение должно быть связано с протоколом, для которого выполняется преобразование адресов.

Зачем так делать??

Периодическая рассылка широковещательных пакетов нежелательна. Представим сегмент Ethernet, содержащий 100 станций, каждая из которых отправляет широковещательный пакет преобразования адресов каждые 10 минут. В результате передача широковещательных пакетов будет происходить в среднем каждые 6 секунд. Это значение невелико – почему же не пойти таким путем? Рабочая станция обычно не обменивается данными со всеми другими станциями и, следовательно, не имеет в таблице все 100 записей, которые могут потребоваться. Станция обычно обменивается данными с серверами, мейнфреймами и значительно реже – с другими станциями. Описанный здесь протокол распространяет информацию только по мере необходимости и (возможно однократно) при загрузке станции.

Описанный формат не позволяет получить преобразование для нескольких адресов с использованием одного пакета. Это сделано в целях упрощения. Если использовать мультиплексирование, формат пакетов станет существенно сложнее и много информации будет передаваться бессмысленно. Можно представить себе мост, который поддерживает 4 протокола и сообщает станциям все 4 протокольных адреса, три из которых может быть никогда не потребуются.

Предлагаемый формат позволяет повторно использовать буфер пакетов для генерации откликов – отклик имеет такой же размер, как запрос, и многие поля запроса и отклика совпадают.

Значение поля ar\$hrd описываемый протокол берет из списка. В настоящее время определено только значение для сетей Ethernet 10 Мбит/с (ares_hrd\$Ethernet = 1). Обсуждается также вопрос об использовании протокола для сетей Packet Radio Networks и для этого типа сетей также потребуется определить соответствующее значение поля. В будущем, когда протокол станет использоваться для других типов оборудования, потребуется определить соответствующие значения и для них.

Для сетей Ethernet 10 мбит/с значение поля протокола (ar\$pro) берется из набора ether_type\$. Это позволяет воспользоваться уже определенными идентификаторами типа протокола. Комбинация этого поля с типом операции (ar\$op) будет позволять эффективно увеличить число протоколов, для которых может обеспечиваться преобразование адресов, но усложняет мониторинг и отладку⁷. Будем надеяться, что число протоколов никогда не превысит 32768, но закон Мэрфи (Murphy) оставляет место для некоторых сомнений.

⁵ Запрос ARP в современной терминологии. *Прим. перев.*

⁶ См. комментарии в параграфе *Связанные вопросы*.

⁷ См. параграф *Мониторинг и отладка в сети*.

В теории поля размера (`ar$hlen` и `ar$plen`) являются избыточными, поскольку размер протокольного адреса должен определяться типами оборудования (`ar$hrd`) и протокола (`ar$pro`). Эти поля включены для дополнительной проверки согласованности, а также мониторинга и отладки (см. ниже).

Код операции (`opcode`) определяет тип пакета – запрос (который может вызвать отклик) или отклик на полученный ранее запрос. 16 битов для этого слишком много, но это поле (флаг) является обязательным.

Аппаратный и протокольный адреса отправителя являются безусловно необходимыми – значения этих полей помещаются в таблицу трансляции адресов.

Протокольный адрес получателя обязателен в запросах, поскольку этот адрес служит для проверки наличия записи в таблице трансляции и генерации отклика. В откликах это поле является необязательным, если отклик генерируется в ответ на запрос. Это поле используется для полноты, мониторинга и упрощения алгоритма обработки, описанного выше (алгоритм не просматривает код операции до включения в таблицу сведений об отправителе).

Аппаратный адрес получателя включается для полноты и мониторинга. Это поле не имеет смысла для запросов, поскольку запрашивается именно значение аппаратного адреса получателя. В откликах это поле включает аппаратный адрес машины, сделавшей запрос на трансляцию адреса. В некоторых реализациях (например, в тех, которые не просматривают 14-байтовый заголовок Ethernet) возможна некоторая экономия ресурсов (регистров процессора и пространства стека) за счет передачи этого поля драйверу устройства в качестве аппаратного адреса получателя пакета.

Между адресами не помещается никаких битов заполнения. Данные пакета должны рассматриваться как поток байтов, в котором только три пары байтов определены как слова (`ar$hrd`, `ar$pro` и `ar$op`), содержащие сначала старший бит (стиль передачи Ethernet/PDP-10).

Мониторинг и отладка в сети

Описанный выше протокол преобразования адресов (Address Resolution protocol) позволяет компьютеру получить информацию об активности протоколов вышележащих уровней (например, CHAOS, Internet, PUP, DECnet) машин, подключенных к сегменту Ethernet. Можно определить тип используемого протокола Ethernet (значение идентификатора) и протокольный адрес для каждого типа протокола. Фактически для мониторинга активности протоколов вышележащих уровней не требуется осуществлять прослушивание каждой станции. Вместо этого можно сделать что-то похожее на описанные ниже процедуры:

Когда монитор получает пакет Address Resolution, он помещает триплет <тип протокола, протокольный адрес отправителя, аппаратный адрес отправителя> в таблицу. Размеры аппаратного и протокольного адреса могут быть определены из полей `ar$hlen` и `ar$plen` принятого пакета. Если код операции указывает на **отклик** (REPLY), монитор может отбросить принятый пакет. Если код операции указывает на **запрос** (REQUEST) и заданный протокольный адрес соответствует протокольному адресу монитора, последний передает отклик в обычном режиме. Монитор будет выполнять только одно отображение, поскольку **отклик** на **запрос** передается непосредственно запрашивающему хосту. Монитор может попытаться передавать свои запросы, но это может привести к возникновению петли в передаче запросов, поэтому следует принимать меры предосторожности.

Поскольку протокол и код операции не объединяются в одно поле, монитор не обязан знать, какой код операции для запроса связан с кодом операции для отклика того же протокола вышележащего уровня. Поле размера должно также давать достаточно информации для возможности “разбора” протокольного адреса, несмотря на отсутствие информации об устройстве протокольного адреса.

Рабочая реализация протокола преобразования адресов (Address Resolution) может также использоваться для отладки тестовых реализаций протокола. Можно предположить, что драйвер устройства будет успешно передавать широковещательные пакеты со значением поля типа Ethernet равным `ether_type$ADDRESS_RESOLUTION`. Формат пакета может быть не совсем корректным, поскольку тестовая реализация может содержать ошибки, а управление таблицей может быть достаточно сложным. Поскольку запросы передаются в широковещательном режиме, монитор будет получать из и сможет отображать для отладки.

Пример

Предположим, что имеется два компьютера X и Y, подключенных к одному сегменту Ethernet. Эти машины имеют Ethernet-адреса EA(X) и EA(Y) и IP-адреса IPA(X), IPA(Y). Пусть тип Ethernet для протокола Internet имеет значение ET(IP). После включения машины X она рано или поздно будет передавать пакеты IP, адресованные машине Y, которая находится в том же сегменте. X знает, что нужно передать пакет по адресу IPA(Y) и сообщает драйверу устройства (в нашем примере, драйверу Ethernet) адрес IPA(Y). Драйвер запрашивает модуль преобразования адресов (Address Resolution) о возможности преобразования значений <ET(IP), IPA(Y)> в 48-битовый адрес Ethernet, но поскольку компьютер X был включен недавно, он не имеет такой информации. В результате пакет будет отброшен и взамен будет сгенерирован пакет ARP (ADDRESS RESOLUTION) с полями

```
(ar$hrd) = ares_hrd$Ethernet
(ar$pro) = ET(IP)
(ar$hlen) = размер(EA(X))
(ar$plen) = размер(IPA(X))
(ar$op) = ares_op$REQUEST
(ar$sha) = EA(X)
(ar$spa) = IPA(X)
(ar$tha) = не имеет значения
(ar$tpa) = IPA(Y)
```

и передавать его в широковещательном режиме всем станциям сегмента Ethernet.

Машина Y получает такой пакет и видит, что она понимает этот тип оборудования (Ethernet), указанный протокол (Internet) и имеет адрес, для которого предназначен пакет (`ar$tpa=IPA(Y)`). В результате этот компьютер вносит принятую с пакетом информацию (возможно заменяя имевшуюся запись) в таблицу преобразования адресов, связывая пару <ET(IP), IPA(X)> с

аппаратным адресом EA(X). После этого машина определяет, что пакет является запросом, меняет местами поля, помещая EA(Y) в поле адреса отправителя Ethernet (ar\$sha), останавливает для кода операции значение отклика и передает пакет напрямую (без использования широковещания) по адресу EA(X). С этого момента Y знает, как передавать пакеты X, но X пока не знает, как передавать пакеты Y.

Машина X, получив отклик от Y, формирует отображение <ET(IP), IPA(Y)> на адрес EA(Y), узнает, что пакет является откликом и отбрасывает его. В следующий раз, когда IP-модуль компьютера X будет передавать пакеты машине Y, трансляция адресов завершится успешно и пакет может быть доставлен адресату. Если IP-модуль машины Y решит передать пакет компьютеру X, трансляция адресов также будет успешной, поскольку Y хранит информацию, полученную с запросом от X.

Связанные вопросы

Может возникнуть потребность в организации контроля возраста записей в таблицах трансляции и/или таймаутов. Реализация этих функций выходит за пределы данной спецификации. Более детальное рассмотрение этих вопросов приведено ниже (спасибо MOON@SCRC@MIT-MC).

При перемещении хоста любые иницируемые соединения будут работать в предположении, что таблица преобразования адресов этого хоста была очищена (сброшена) при перемещении. Однако соединения, иницированные другими хостами, не имеют веских причин для корректировки своих таблиц трансляции адресов. Предполагается, что 48-битовые адреса Ethernet являются фиксированными, т. е., не могут изменяться. Однако хост может “перенести” свое имя (и некоторые протокольные адреса) на другое оборудование. Из опыта также известно, что большую опасность представляет некорректная маршрутная информация, переданная в результате аппаратной или программной ошибки – такие случаи просто недопустимы. Возможно при отказе в организации соединения с хостом передавать модулю преобразования адресов сигнал о необходимости удаления из таблицы информации о недоступном хосте – этот хост может быть выключен или старое отображение адресов перестало быть корректным. Можно также при получении пакета от хоста сбрасывать таймер для соответствующей записи в таблице трансляции адресов, используемый для удаления записи по возрасту (запись удаляется, если в течение заданного времени от хоста не было получено ни одного пакета). Однако просмотр таблицы при получении каждого пакета может вызвать чрезмерную загрузку процессора. Для предотвращения перегрузки могут использоваться механизмы хэширования или индексации.

Предложенный механизм приема пакетов трансляции адресов пытается в течение некоторого времени услышать перенесенный хост. Напомним, что при наличии в таблице преобразования пары <тип протокола, протокольный адрес отправителя> в этой записи будет заменяться поле аппаратного адреса отправителя запроса. Следовательно, в сети Ethernet, где широковещательные запросы слышат все станции, каждая станция будет получать новый аппаратный адрес.

Другим вариантом является использование демона, отслеживающего тайм-ауты. По истечении определенного времени такой демон рассматривает вопрос удаления записи из таблицы трансляции. Демон сначала передает (при необходимости с небольшим числом повторов) пакет трансляции адреса с кодом операции для запроса (REQUEST) по адресу Ethernet, имеющемуся в таблице. Если в течение заданного времени отклика не приходит, запись удаляется из таблицы трансляции. Запрос передается напрямую, чтобы не загружать работой каждую станцию Ethernet. Простое удаление записей из таблицы будет приводить к бессмысленной потере информации, поэтому его следует избегать.

Поскольку хост передает информацию только о себе, перезагрузка хоста не ведет к необходимости замены связанной с этим хостом записи в таблице трансляции. Некорректная информация не должна никаким способом перемещаться от одной машины к другой – единственный случай допустимости существования некорректной информации допустим лишь тогда, когда хост еще не знает об изменении 48-битового адреса Ethernet другого хоста. Возможно, что в таких случаях достаточно будет сбросить (очистить) вручную таблицу трансляции.

Этот вопрос требует дополнительного рассмотрения, если важность его достаточно велика. Вопрос может быть связан с любыми протоколами, подобными трансляции адресов.

Перевод на русский язык

Николай Малых

Nikolai MalykhBiLiM Systems

nmalykh@bilim.com