Network Working Group
Request for Comments: 917

Jeffrey Mogul
Computer Science Department
Stanford University

October 1984

Подсети Internet

INTERNET SUBNETS

Статус документа

В этом RFC описан протокол, предложенный сообществу ARPA-Internet, и содержится приглашение к дискуссии и совершенствованию протокола. Документ может распространяться без ограничений.

Обзор

Мы обсуждаем парадигму «подсетей» (subnet) в сетях Internet[1], представляющих собой видимые подмножества единой сети IP. По административным и техническим причинам многие организации делят свою сеть IP на несколько подсетей вместо приобретения дополнительных блоков адресов IP.

Мы предлагаем процедуры для использования подсетей и обсуждаем варианты решения возникающих проблем (в частности, маршрутизации).

Благодарности

Эти предложения являются результатом обсуждения с некоторыми людьми. В частности, J. Noel Chiappa, Chris Kent, и Tim Mann внесли важные предложения.

1. Введение

Исходное представление сети Internet включало два уровня иерархии — верхний уровень представлял сеть catenet в целом, и нижний — множество сетей IP, каждая из которых имеет свой номер (мы не предполагаем в Internet иерархической топологии, но интерпретация адресов является иерархической).

Хотя такое представление обеспечивает простоту и мощь, многие организации считают его неадекватным и добавляют третий уровень в иерархическую схему интерпретации адресов IP. В этом случае сеть IP может быть поделена на некоторое количество подсетей.

Исходное, двухуровневое представление основано на строгом допущении о том, что для хоста в сети IP эта сеть может представляться, как единое целое; иными словами, сеть можно трактовать, как «черный ящик», к которому подключены хосты. Это верно для ARPANET, поскольку IMP маскируют использование конкретных каналов в сети. Это верно также для большинства технологий ЛВС типа Ethernet или сетей с кольцевой топологией.

Однако это допущение неверно для многих практических случаев, поскольку организациям средних размеров (например, университетам или компаниям, расположенным в нескольких зданиях) зачастую приходится использовать более одной ЛВС[2] для покрытия «локальной области». Например, на момент подготовки этого документа в сети Стэнфордского университета использовалось 18 ЛВС и планировалось увеличение их числа.

Существует несколько причин, по которым организации могут использовать несколько ЛВС для покрытия территории:

- ***Различные топологии*** – зачастую (особенно в исследовательских организациях) в сети могут присутствовать ЛВС различных типов (например, Ethernet и сети с кольцевой топологией).
- ***Технологические ограничения*** – большинству топологий ЛВС присущи те или иные ограничения (электрические параметры, число хостов в ЛВС, общая протяженность кабеля). Использование подсетей позволяет достаточно просто преодолеть эти ограничения (особенно ограничение протяженности кабеля).
- ***Насыщение сети*** – в локальных сетях могут возникать ситуации, когда небольшая группа хостов фактически монополизирует полосу пропускания сетевой среды. Общим решением таких проблем является сегментирование сети (разбиение на несколько фрагментов) с организацией раздельных кабельных систем.
- ***Использование каналов «точка-точка»*** - иногда «локальная сеть» (например, кампусная сеть университета) разделена на несколько фрагментов (например, ЛВС отдельных зданий), соединенных между собой скоростными каналами «точка-точка».

Организации, имеющие более одной ЛВС, могут выбрать один из трех вариантов распределения адресов IP:

1. Получить отдельный блок адресов для каждой ЛВС.
2. Использовать для всей организации общий номер сети и распределять адреса независимо от принадлежности хостов к ЛВС. Такой вариант получил название transparent subnets (прозрачные подсети).

[1] Сеть IP в современной терминологии, используемой в переводе. *Прим. перев.*
[2] В оригинале используется термин «кабель», поскольку ЛВС во время создания документа чаще всего представляла собой сеть с «общей» коаксиальной шиной, соединяющей последовательно все компьютеры сети. *Прим. перев.*

3. Используя один номер сети, разделить адресное пространство на несколько подсетей в соответствии с имеющимися ЛВС. Такой вариант называется explicit subnets (явные подсети).

Каждый из вариантов имеет свои преимущества и недостатки. Первый вариант не требует добавления или изменения протоколов, но ведет к разрастанию таблиц маршрутизации Internet. Информация о внутренней структуре сети распространяется повсюду, хотя она мало кому нужна и в большинстве случаев не используется за пределами организации. В некоторых реализациях шлюзов может возникать проблема нехватки пространства в таблице маршрутов, поэтому такой вариант лучше не использовать[1].

Второй вариант требует использования неких соглашений или протоколов, позволяющих объединить множество ЛВС в одну сеть IP. Например, этот вариант можно реализовать в ЛВС, где каждый адрес IP транслируется в аппаратный адрес с использованием протокола ARP[2], имея мосты между ЛВС, которые будут перехватывать запросы ARP для нелокальных адресатов. Однако такое решение возможно не для всех технологий ЛВС, особенно в тех случаях, когда технология не использует протокол ARP или ЛВС не поддерживает широковещания. Более фундаментальная проблема заключается в том, что мосты должны узнавать к какой ЛВС принадлежит хост, возможно используя для этого широковещательные рассылки. По мере увеличения числа ЛВС расходы на поддержку такого широковещания будут быстро возрастать; расти будет и размер кэша трансляции в мостах, требуемого для преобразования адресов.

Третий вариант рещает ключевую проблему - существующие стандарты предполагают, что все хосты в сети IP подключены к одной ЛВС. Решение заключается в явной поддержке подсетей. Этот вариант тоже имеет недостатки – в частности, требуется модификация протокола IP, которая влечет за собой необходимость изменения существующих реализаций IP (если планируется использовать их с подсетями). Однако требуемые изменения сравнительно невелики и после их внесения проблема будет эффективно решена[3]. Кроме того, этот вариант не требует каких-либо изменений, которые будут несовместимы с существующими хостами в сетях без разбиения на подсети.

После выбора одного из вариантов может оказаться, что хосты, которые предполагалось использовать в среде без подсетей, попадут в одну из подсетей, как описано ниже. Такое решение может быть полезно в тех случаях, когда нет возможности явно поддерживать подсети или требуется постепенный переход. С учетом этого для использования второго варианта, описанного выше, нет достаточно веских причин.

В остальной части этого документа описывается модель разбиения сетей IP на подсети.

1.1. Терминология

Во избежание разнотолков и многословия определим термины, которые будут использоваться далее.

Catenet

Набор соединенных между собой сетей IP.

Network - сеть

Отдельная сеть IP (поделенная или не поделенная на подсети).

Subnet - подсеть

Подсеть сети IP.

Network Number - номер сети

В соответствии с [8].

Local Address - локальный адрес

Биты адреса IP, не используемые в номере сети. Используется также термин «rest field» (остальные биты).

Subnet Number - номер подсети

Номер, идентифицирующий подсеть внутри сети.

Subnet Field - поле подсети

Битовое поле в адресе IP, используемое для задания номера подсети.

Host Field - поле хоста

Битовое поле в адресе IP, используемое для адресации конкретного хоста.

Gateway - шлюз[4]

Узел, подключенный к двум или более сетям с различным административным управлением и/или подсетям, которому хосты направляют дейтаграммы для пересылки.

Bridge - мост

Узел, подключенный к двум или более подсетям, разделенным физически, но находящимся под единым административным управлением, который автоматически пересылает между этими сетями дейтаграммы (при необходимости), оставаясь «незаметным» для остальных хостов. Для обозначения мостов используют также термин software repeater[5] (программный повторитель).

2. Стандарты для адресации подсетей

Следуя делению, представленному в [2], мы обнаружили, что подсети являются фундаментальной проблемой адресации. В этом разделе мы впервые описываем предложенную интерпретацию адресации Internet Addressing для поддержки подсетей. Далее рассматривается взаимодействие этого формата адресации с широковещанием и в заключение представлен протокол детектирования используемого в конкретной сети способа интерпретации адресов.

---

[1] Современные маршрутизаторы и протоколы маршрутизации не порождают таких проблем, но этот вариант все равно остается нежелательным по причине общей нехватки адресов IP и снижения уровня защиты сетей. *Прим. перев.*

[2] Address Resolution Protocol - протокол преобразования адресов. *Прим. перев.*

[3] В настоящее время эта проблема успешно разрешена. *Прим. перев.*

[4] Сейчас такие узлы обычно называют «маршрутизаторами». *Прим. перев.*

[5] В настоящее время этот термин устарел и практически не используется. *Прим. перев.*

2.1. Интерпретация адресов IP

Предположим, что организации был выделен номер сети Internet[1], сеть была поделена на множество подсетей и нужно распределять адреса для хостов. Как следует это делать? Поскольку существуют минимальные ограничения на выделение «локальной части» адреса Internet, предлагается несколько вариантов представления номеров подсетей:

1. **Поле переменной длины** - для нумерации подсетей используется произвольное число битов локальной части адреса. Размер поля номера подсети сохраняется в масштабах каждой подсети, но может различаться в разных подсетях. Если поле имеет нулевой размер, это говорит об отсутствии подсетей.
2. **Поле фиксированной длины** - для нумерации подсетей используется фиксированное число битов (например, 8).
3. **Поле переменной длины с автокодированием**. Классы сетей (размер поля номера сети) определяются значениями старших битов адреса IP – аналогично этому старшие биты локальной части адреса могут задавать размер поля номера подсети.
4. **Поле фиксированной длины с автокодированием**. Для нумерации подсетей используется заданное число битов. Единица в старшем бите этого поля говорит об использовании подсетей, при нулевом значении старшего бита вся локальная часть адреса служит для нумерации хостов[2].

Поскольку отсутствуют видимые признаки преимуществ иного варианта, во всех этих схемах для нумерации подсетей используются старшие биты локальной части адреса. Поскольку для сетей класса C локальная часть достаточно мала, для этого класса адресов использование подсетей менее осмысленно, нежели для сетей класса A или B.

По каким критериям следует выбирать одну из четырех предложенных схем? Во-первых, мы хотим использовать автокодирование, позволяющее по адресу Internet определить, используется ли в сети разбиение на подсети без привлечения другой информации.

Одним из преимуществ автокодирования является возможность определить, поделена ли нелокальная сеть на подсети. Практическое использование этого преимущества не очевидно. Однако основное преимущество заключается в том, что реализации не требуется дополнительных данных для определения принадлежности двух адресов к одной подсети. Однако это может казаться недостатком, поскольку может вызывать проблемы для сетей без разбиения на подсети, в которых номера с 1 в старшем бите локальной части адреса уже используются[3] (см. примечание 2 в конце страницы). Иными словами, полезно иметь возможность контроля разбиения сети на подсети независимо от распределения номеров хостов. Другим недостатком любой схемы с автокодированием является то, что такие схемы уменьшают доступный размер локального пространства адресов по крайней мере вдвое.

Если не использовать схему с автокодированием, ясно, что предпочтительным будет вариант с переменным размером поля номера подсети. Поскольку в любом случае требуется некий «флаг», показывающий использование подсетей, дополнительные издержки в результате использования целого числа (размер поля номера подсети) вместо логической переменной (флаг) пренебрежимо малы. Преимущество использования поля номера подсети с переменным размером заключается в том, что каждая организация получает возможность выбора способа использования достаточно скудного числа битов локальной части адреса для нумерации подсетей и хостов.

Наше предложение, следовательно, заключается в том, что адреса Internet интепретируются, как три поля:

```
<network-number><subnet-number><host-number>
```

где поле <network-number> определено в [8], поле <host-number> имеет размер не менее 1 бита, а поле <subnet-number> постоянно для данной подсети. Дополнительной структуры для полей <subnet-number> и <host-number> не требуется. Если размер поля <subnet-number> равен 0, сеть не делится на подсети (т. е., используется интерпретация [8]).

Например, в сети класса A с 8-битовым полем номера подсети поля адреса имеют вид, показанный на рисунке справа.

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|       Сеть    |     Подсеть   |          Номер хоста          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Предполагается, что для упрощения и повышения эффективности реализаций протокола большинство организаций будет выбирать размер поля номера подсети, кратный восьми битам. Однако реализации должны быть готовы к обработке полей номера подсети иного размера.

Мы отказываемся от рекурсивного использования подсетей с дополнительным делением поля номера хоста по ряду причин:

– четырехуровневая иерархия обычно не требуется;

– число битов адреса IP недостаточно велико для такого деления;

– сложность механизмов дополнительного деления.

2.2. Измнение программ хостов для поддержки подсетей

В большинстве реализация протокола IP имеется код для обслуживания исходящих потоков типа приведенного ниже:

```
IF ip_net_number(packet.ip_dest) = ip_net_number(my_ip_addr)
    THEN
```

[1] В современной терминологии «блок адресов». *Прим. перев.*

[2] В RFC 950 этот тезис уже исключен. Причина, по-видимому, заключается в некоторой противоречивости самого тезиса и неэффективном использовании адресов при отказе от подсетей (старший бит локальной части в этом случае не может иметь значение 1, что приводит к невозможности использования половины адресов. *Прим. перев.*

[3] Например, для некоторых хостов используются адреса, представляющие собой комбинацию номера сети класса A и 24 младших битов 48-битового аппаратного адреса Ethernet.

```
            send_packet_locally(packet, packet.ip_dest)
        ELSE
            send_packet_locally(packet, gateway_to(ip_net_number(packet.ip_dest)))
```

Код с поддержкой множества подключенных сетей несколько сложнее, но в данном случае это не имеет значения.

Для поддержки подсетей нужно сохранить одно дополнительное 32-битовое значение - маску IP (my_ip_mask). Эта битовая маска представляет собой строку битов, в которой установлены (1) значения битов, соответствующий номеру сети IP и номеру подсети. Например, для сети класса A с 8-битовым полем номера подсети маска будет иметь значение 255.255.0.0.

Упомянутый выше код тогда принимает вид:

```
IF bitwise_and(packet.ip_dest, my_ip_mask) = bitwise_and(my_ip_addr, my_ip_mask)
    THEN
        send_packet_locally(packet, packet.ip_dest)
    ELSE
        send_packet_locally(packet, gateway_to(bitwise_and(packet.ip_dest, my_ip_mask)))
```

Очевидно, что часть выражения в условном операторе может быть вычислена заранее.

Может потребоваться изменение функции gateway_to с включением такого же условия сравнения.

Для поддержки хостов со множеством подключения код может быть изменен так, чтобы значения my_ip_addr и my_ip_mask сохранялись для каждого интерфейса, а выражения в условном операторе должны вычисляться для каждого интерфейса.

2.3. Подсети и широковещание

В отсутствии подсетей для протокола IP возможны только два варианта широковещания[1] - всем хостам указанной сети или всем хостам данной сети. Последний вариант полезен в тех случаях, когда хост не знает номера своей сети.

При наличии подсетей ситуация слегка усложняется. Во-первых появляется возможность широковещательной передачи в масштабе подсети. Во-вторых, для широковещания всем хостам подсети требуется дополнительный механизм (в работе [6] предложено использовать механизм[2] [3]). И, наконец, интерпретация широковещания в данную сеть изменяется и пакеты не пересылаются за пределы исходной подсети.

Следовательно, реализации должны распознавать три типа широковещательных адресов в дополнение к своему адресу хоста:

This physical network - данная физическая сеть

Адрес получателя, состоящий из одних единиц (255.255.255.255), приводит к передаче дейтаграммы как широковещательной в масштабе физической сети; для шлюзов пересылка таких дейтаграмм недопустима.

Specific network - указанная сеть

Адрес получателя содержит корректный номер сети, а локальная часть адреса - только единицы (например, 36.255.255.255).

Specific subnet - указанная подсеть

Адрес получателя содержит корректные номера сети и подсети, а номер хоста — только единицы (например, 36.40.255.255).

Дополнительную информацию о широковещании Internet можно найти в работе [6].

Одним из факторов при решении вопроса об использовании подсетей является возможность широковещания всем хостам сети с подсетями в один прием со стороны передающего хоста. Если бы хосты находились в разных сетях, такая операция стала бы невозможной.

2.4. Определение размера поля номера подсети

Как хост (или шлюз) может определить использование поля номера подсети в подключенной к нему сети? Эта проблема аналогична проблемам, возникающим при загрузке хостов Internet, - как хост может узнать свой адрес и адрес шлюза в своей сети? Для всех ситуаций существует два варианта решения этих проблем - «аппаратная» информация и протоколы на основе шероковещания.

«Аппаратная» информация удобна для изолированных (не подключенных к сети) хостов. Она может быть «встроена» или храниться в дисковом файле (предпочтительно). Однако для широко распространенного случая загрузки бездисковых станций через ЛВС, «аппаратная» информация не подходит. Если технология ЛВС поддерживает широковещание, лучшим методом будет использование «свежезагруженным» хостом широковещательной передачи запроса на получение требуемой информации. Например, для определения адреса IP хост может использовать протокол обратного преобразования адресов RARP[3] [4].

Мы предлагаем расширить протокол ICMP [9], добавив пару типов сообщений - Address Format Request и Address Format Reply[4], аналогичных сообщениям Information Request и Information Reply (см Приложение I)[5].

Новые сообщения ICMP используются следующим образом – хост при загрузке передает широковещательное сообщение Address Format Request[6], а шлюз (или хост, действующий вместо шлюза), получив такое сообщение, будет

[1]Широковещание Internet рассматривается на основе работы [6].
[2]Reverse Path Forwarding - пересылка по обратному пути.
[3]Reverse Address Resolution Protocol.
[4]Запрос адресной информации и отклик на такой запрос, соответственно.
[5]В конечном итоге в протокол ICMP были добавлены два типа сообщений Address Format Request и Address Format Reply. *Прим. перев.*
[6]Если широковещание не поддерживается, предполагается, что хост «знает» соседнего шлюза и сообщение ICMP следует передавать этому шлюзу.

передавать отклик Address Format Reply. Если в запросе отправитель не был указан (поле IP Source Address имеет значение 0), отклик также передается в широковещательном сообщении. Запросивший информацию хост получит это сообщение и сможет определить размер поля номера подсети.

Поскольку для каждой конкретной ЛВС в сообщении Address Format Reply может содержаться только одно значение, нет необходимости проверять соответствие отклика запросу. Даже при получении отклика от нескольких шлюзов, информация во всех сообщениях будет совпадать. Предполагается, что хосты перезагружаются достаточно редко, поэтому количество широковещательных сообщений для определения маски будет достаточно мало.

Если хост подключен к множеству ЛВС, он должен использовать этот протокол для каждой из сетей, пока не будет обнаружено (по отклику одной из ЛВС), что некоторые ЛВС относятся к одной сети и, следовательно, имеют одинаковый размер поля номера подсети.

Одной из возможных проблем является отсутствие сообщений Address Format Request после разумного числа попыток запроса. Возможны три причины возникновения таких ситуаций:

1. Локальная сеть постоянно изолирована от всех других сетей.
2. Подсети не используются и ни один из хостов не поддерживает таких запросов ICMP.
3. Все шлюзы локальной сети (временно) находятся в нерабочем состоянии (down).

В первых двух случаях подразумевается, что поле номера подсети имеет нулевой размер. В третьем случае не существует способа определить размер поля номера подсети и самым безопасным вариантом будет считать этот размер нулевым. Хотя позднее может обнаружиться некорректность такого выбора, это не порождает проблем при передаче. После того, как восстановится нормальная работа шлюза, он будет передать широковещательное сообщение Address Format Reply; когда хост получит это сообщение, он сможет заменить свои допущения полученным от шлюза значением. Хостам и шлюзам не следует передавать сообщений Address Format Reply с «предполагаемой» маской.

В заключение отметим, что хост не обязан использовать протокол ICMP для определения размера поля номера подсети – параметры хоста могут сохраняться в энергонезависимой памяти.

3. Методы маршрутизации подсетей

Обной из проблем, с которыми сталкиваются все хосты Internet, является определение маршрута к другому хосту. Использование подсетей лишь слегка изменяет эту проблему.

При использовании подсетей возникает два уровня процесса маршрутизации вместо одного. Если получатель находится в одной сети с отправителем, в маршрутизации участвуют только маршрутизаторы подсетей между хостами. Если получатель находится в другой сети, процесс маршрутизации требует выбора шлюза для выхода из сети отправителя и маршрута через эту сеть к выбранному шлюзу.

К счастью, большинство хостов могут игнорировать это различие (фактически, игнорировать любой выбор маршрута), используя принятый по умолчанию шлюз в качестве начала маршрута ко всем получателям и опираясь на сообщения ICMP Host Redirect для определения более подходящих маршрутов. Однако такой метод неэффективен для шлюзов и многодомных хостов, поскольку перенаправление может оказаться бесполезным при некорректном начальном выборе маршрута. Таким хостам следует использовать протокол обмена маршрутной информацией, но этот вопрос выходит за пределы настоящего документа. В любом случае, описанная выше проблема не зависит от использования подсетей.

Проблема хостов с одним подключением заключается в поиске хотя бы одного соседнего шлюза. Здесь также имеются два варианта - аппаратная информация и широковещание. Мы полагаем, что проблема поиска соседнего шлюза не зависит от использования подсетей и, следовательно, решение этой проблемы при наличии подсетей будет таким же.

Однако еще одна проблема сохраняется - отправитель должен понять, следует передавать дейтаграмму для адресата шлюзу или ее можно отправить получателю напрямую. Иными словами, находится ли адресат в одной физической сети с отправителем? Эта фаза процесса маршрутизации является единственной, которая требует от реализации явной поддержки подсетей. Фактически, если широковещание не применяется, это единственная ситуация, когда реализацию IP требуется изменить для поддержки подсетей.

В силу сказанного, возможно использовать существующие реализации без внесения изменений даже при наличии подсетей[1]. Для того, чтобы это тсало возможным, реализация должна:

– использоваться только на хостах с одним подключением, но не на шлюзах;
– использоваться в широковещательной ЛВС;
– использовать протокол преобразования адресов (типа ARP [7]);
– не требовать поддержки соединения в случаях отказов шлюза.

В такой ситуации можно ограничиться модификацией сервера ARP в подсети так, чтобы он при получении запроса ARP проверял адрес получателя на предмет определения своей принадлежности к лучшему пути в направлении получателя. При положительном результате проверки сервер передает запрашивающему хосту отклик ARP со своим аппаратным адресом. Запрашивающий хост в результате предполагает, что ему известен аппаратный адрес получателя и передает пакеты по этому адресу. Фактически пакеты получает шлюз, который пересылает их адресату обычным путем.

Этот метод требует некоторого «размывания» уровней на шлюзах, поскольку сервер ARP и таблица маршрутизации IP обычно не связаны. Однако реализовать такую связь можно достаточно просто и без существенного снижения производительности. Одна из проблем состоит в том, что при аварии исходного шлюза для отправителя не будет способа узнать другие маршруты к получателю, если такие маршруты имеются. В результате соединение будет прервано.

[1] В более ранних работах это назувалось «сосуществованием прозрачных и явных подсетей в одной сети».

Не следует путать этот метод организации «подсетей на базе ARP» со слегка похожим на него использованием мостов на основе ARP. Подсети на базе ARP используют возможность шлюза проверять адрес IP и устанавливать маршрут к получателю на основе явной топологии подсети. Иными словами, малая часть процесса выбора маршрута переносится от хоста-отправителя к шлюзу. Мост на основе ARP, напротив, должен узнать расположение каждого хоста без помощи отображения между адресами хостов и топологией. Системы, построенные на основе таких мостов, не следует считать поделенными на подсети.

Важно отметить, что использование подсетей на базе ARP осложняется широковещанием. Серверу ARP [7] никогда не следует отвечать на запросы, где получатель имеет широковещательный адрес. Такие запросы могут исходить только от хостов, которые не распознают широковещательных адресов в таком качестве и ответ на подобный запрос почти всегда будет приводить к возникновению маршрутной петли. Если имеется N таких хостов, которые не распознают адрес в качестве широковещательного, пакет, переданный с TTL[1] может приводить к возникновению T^N ненужных широковещательных пакетов.

4. Примеры использования подсетей

В этом разделе кратко описаны примеры использования подсетей в нескольких организациях.

4.1. Сиэнфордский университет

В Стенфордском университете подсети появились по историческим причинам. В сети университета использовались протоколы Pup [1] для нескольких экспериментальных сетей Ethernet [5] с 1979 года, на несколько лет раньше начала использования протоколов Internet. В работе было множество шлюзов Pup, а все хосты и шлюзы обменивались таблицами маршрутизации с использованием простого широковещательного протокола.

После появления протокола IP было принято решение об использовании восьмибитового поля номера подсети и номера подсетей Internet были выбраны в соответствии с номерами сетей Pup для данной ЛВС Ethernet и номерами хостов Pup (тоже восемь битов) в качестве номеров хостов в адресах Internet.

Шлюзы, поддерживающие только Pup, были модифицированы для пересылки дейтаграмм Internet в соответствии с таблицами маршрутизации Pup, поскольку без такой модификации они просто не понимали пакетов Internet и фактически не меняли значение поля TTL в заголовках IP. Такое решение представлялось приемлемым, поскольку ошибок, вызывающих появление маршрутных петель, не наблюдалось. Хосты Internet, которые были многодомными и могли, таким образом, функционировать в качестве маршрутизаторов, меняли значение TTL. Поскольку все такие хосты одновременно являлись шлюзами Pup дополнительного обмена маршрутными данными не требовалось.

Реализации хостов Internet были модернизированы для поддержки подсетей (различными путями, но с одинаковым результатом). Поскольку все эти хосты уже имели реализации Pup, таблицы маршрутизации Internet поддерживались теми же процессами, что и таблицы Pup - номера сетей Pup просто транслировались в номера подсетей Internet.

При добавлении сетей Ethernet 10 Мбит/с шлюзы были модифицированы для использования описанной выше схемы на базе ARP. Это позволило использовать немодифицированные хосты в сетях Ethernet 10 Мбит/с.

Подсети IP начали использоваться с 1982 г; в настоящее время[2] насчитывается около 330 хостов, 18 подсетей и близкое число шлюзов между подсетями. Поскольку шлюзы, поддерживающие только Pup, были преобразованы в шлюзы Internet, был добавлен протокол обмена маршрутной информацией Internet, заменивший протокол Pup.

4.2. MIT

MIT[3] был первым сайтом IP с большим набором локальных сетевых соединений. Поскольку это было до деления сетевых номеров на классы, выделение каждому каналу MIT своего номера сети IP привело бы к расходу значительной части адресного пространства. В MIT было принято решение использовать один номер сети IP и самостоятельно управлять оставшимися 24 битами адреса, разделив их на три 8-битовых поля - номер подсети, резервное поле (0) и номер хоста. Поскольку уже использовавшийся в MIT протокол CHAOS работает с восьмибитовым полем номера подсети, можно было выделить каждому каналу одинаковые номера для обоих протоколов. Для номера хоста IP было выбрано 8-битовое поле потому, что в тот момент большинство сетевого оборудования использовало 8-битовые адреса, как в протоколе CHAOS. Это позволило также зарезервировать часть битов адреса IP на будущее.

Первоначальный план предполагал использование протоколов динамической маршрутизации между шлюзами подсетей IP - несколько протоколов этого типа тогда обсуждалось, но ни один не был реализован, поэтому продолжают использоваться статические таблицы. Ясно, что переход к динамической маршрутизации со временем произойдет.

Для решения проблемы с необходимостью модификации импортируемых программ IP для работы в средаxx с подсетями, MIT начал искать модель, позволяющую обойтись минимальными изменениями IP на хостах. В результате была выбрана модель, в которой шлюзы IP передают сообщения ICMP Host Redirect, а не Network Redirect. Сейчас все внутренние IP-шлюзы MIT поступают таким образом. С хостами, которые могут поддерживать таблицы маршрутизации IP для нелокального обмена пакетами на уровне хостов, это позволяет скрыть основную часть структуры подсетей. Минимальные правки программ на хостах для корректной работы в средах с подсетями и без них включали поддержку алгоритма битовых масок, упомянутого выше.

У MIT нет плана незамедлительного перехода на использование одного «одобренного» протокола — это обусловлено локальной автономностью и количеством установленных программ, а также отсутствием единого индустриального стандарта. Вместо этого была принята модель обеспечения одного набора физических каналов и пакетных коммутаторов и деление на несколько «виртуальных» сетей в одном наборе физических каналов. У MIT был некоторый опыт попыток обмена маршрутной информацией между протоколами и «заворачивания» одного протокола в другой. Общей моделью было сохранение изоляции протоколов при использовании общего базового оборудования. Использование ARP для сокрытия структуры подсетей не так важно, поскольку ведет к усложнению операций преобразования адресов. В усложненной системе (с петлями и разноскоростными каналами) потребуется более

[1] Time-To-Live - время жизни пакета в сети.
[2] 1984 год. *Прим. перев.*
[3] Massachusetts Institute of Technology - Массачусетский технологический институт. *Прим. перев.*

изощренный обмен информацией между шлюзами, что делает этот явный (но изолированный от хоста) механизм лучшим решением.

4.3. Университет Карнеги-Мэллона

CMU[1] использует сеть класса B, поделенную на 11 физических подсетей (2 сети 3Mbit Experimental Ethernet, 7 сетей Ethernet 10 Мбит/с и два кольца ProNet). Адреса хостов распределены так, что все адреса с данным третьим октетом относятся к одной подсети (обратное не обязательно верно), это важно с точки зрения удобства администрирования. Программы не знают специфики этого механизма распределения и не зависят от маршрутов между ЛВС.

Вместо этого используется схема с мостами на базе ARP. Когда хост передает широковещательный запрос ARP, все мосты, получившие этот запрос, кэшируют исходное отображение протокольного адреса и пересылают запрос (после предварительных операций), как широковещательный запрос ARP в каждую из подключенных к мосту сетей. Когда мост получает (не широковещательный) отклик ARP, в котором целевой протокольный адрес не относится к этому мосту, он просматривает кэш ARP для определения сети, в которую следует переслать отклик. Мосты, таким образом, пытаются расширить протокол ARP на гетелогенную сеть из множества ЛВС. Следовательно, мостам требуется преобразовывать широковещательные запросы ARP из одной сети в широковещательные запросы ARP во все другие сети, подключенные к мосту, даже когда мост «знает лучшую» сеть. Этот алгоритм работает только в отстуствии циклов в графе сетевой связности (в данном случае это выполняется). Ведутся работы по замене этого простого алгоритма протоколом, реализованным в мостах, для поддержки избыточных путей и снижения объема широковещания. Задача состоит в сохранении базы ARP и прозрачности хостов, если это возможно.

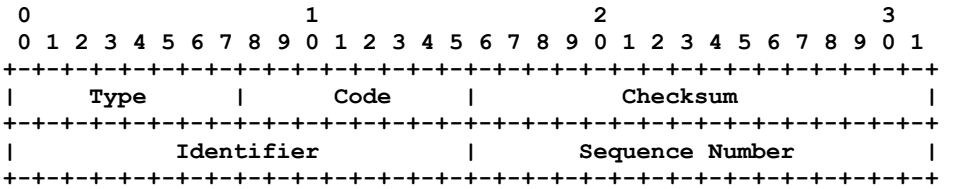
Реализации, поддерживающие Ethernet 3 Мбит/с и кольца proNET 10 Мбит/с в CMU, используют RFC 826 ARP (взамен того или иного аппаратного отображения типа простого использования 8-битовых аппаратных адресов в качестве четвертого октета адреса IP).

Поскольку избыточные пути между ЛВС отсутствуют, возникает вопрос поддержки соединений при аварии моста. При количестве поддерживающих IP хостов порядка 150 на сеть, размер кэша в мостах сохраняется в разумных пределах и на пересылку широковещательных запросов ARP расходуется достаточно малая полоса.

Сеть CMU растет от небольшой конфгурации с одним подключением в подразделении CS/RI до кампусной сети с множеством департаментов, 5000-10000 хостов и избыточными соединениями между ЛВС. Возможно, что схема с мостами на базе ARP не обеспечит требуемого масштабирования и может потребоваться система с явными подсетями. Среднесрочной задачей, однако, является создание среды, в которую можно импортировать необновленные реализации IP (в частности, Ethernet 10 Мбит/с), чтобы сохранить прозрачный (т. е., основанный на ARP) механизм маршрутизации как можно дольше. CMU считает, что даже при включении подсетей в стандарты IP, они не будут использоваться достаточно широко — в этом состоит основная помеха использованию подсетей в CMU.

I. Формат пакетов ICMP

Пакеты Address Format Request и Address Format Reply включают показанные на рисунке поля[2].



Поля IP

Адреса

Адрес отправителя в сообщении Address Format Request будет адресом получателя Address Format Reply. При создании отклика адрес отправителя из запроса становится адресом получателя, а в качестве отправителя отклика указывается адрес отвечающего хоста, код типа меняется на A2, значение размера поля номера подсети помещается в поле Code и запово рассчитывается контрольная сумма. Однако если в качестве адреса отправителя запроса указан 0, в качестве адреса получателя отклика указывается широковещательный адрес.

Поля ICMP

Type - тип

A1 для запросов;
A2 для откликов.

Code - код

0 для сообщений Address Format Request.
Размер поля номера подсети в сообщениях Address Format Reply.

Checksum - контрольная сумма

Контрольная сумма представляет собой 16-битовое поразрядное дополнение до 1 суммы дополнений до 1, рассчитанной для сообщения ICMP, начиная с поля ICMP Type. При расчете контрольной суммы значение поля принимается нулевым. Механизм расчета контрольной суммы в будущем может измениться.

Identifier - идентификатор

Идентификатор служит для сопоставления запросов и откликов. Может иметь нулевое значение.

Sequence Number - порядковый номер

Номер служит для сопоставления запросов и откликов. Может иметь нулевое значение.

Описание

Шлюзу, получившему сообщение Address Format Request следует возвратить отклик на него, указав в поле Code число битов поля номера подсети в адресах IP для сети, в которую дейтаграмма была адресована. Если запрос

[1]Carnegie-Mellon University.

[2]В RFC 950, принятом в качетсве стандарта, даны слегка отличающиеся имена сообщений и их полей. Перевод этого стандарта имеется на сайте www.protocols.ru. *Прим. перев.*

был широковещательным, получателем будет «данная сеть». Размер поля Subnet может принимать значения от 0 до (31 - N), где N задает число битов в поле номера сети IP (т. е., 8, 16 или 24).

Если запрашивающий хост не знает своего адреса IP, он может указать значение 0 в поле адреса отправителя; отклик следует передавать по широковещательному адресу. Поскольку для сети существует один возможный формат адреса, нет необходимости устанавливать соответствие между запросами и откликами. Однако такого подхода следует, по возможности, избегать, поскольку он ведет к росту широковещательной нагрузки на сеть.

Тип A1 может приходить от шлюза или хоста.

Тип A2 может приходить от шлюза или хоста, действующего в качестве шлюза.

II. Примеры

В приведенных здесь примерах предполагается, что запрашивающий хост имеет адрес 36.40.0.123, адрес шлюза - 36.40.0.62, сеть - 36.0.0.0, а поле номера подсети занимает 8 битов.

Сначала предположим, что широковещание разрешено и хост 36.40.0.123 знает свой адрес. Он будет отправилять дейтаграмму:

```
Source address:          36.40.0.123
Destination address:     36.255.255.255
Protocol:                ICMP = 1
Type:                    Address Format Request = A1
Code:                    0
```

36.40.0.62 будет слышать эту дейтаграмму в ответ на которую ему следует передать отклик:

```
Source address:          36.40.0.62
Destination address:     36.40.0.123
Protocol:                ICMP = 1
Type:                    Address Format Reply = A2
Code:                    8
```

В качестве следующего примера предположим, что 255.255.255.255 означает «широковещательный адрес данной физической сети», как описано в [6].

Предыдущий пример неэффективен, поскольку запрос может в широковещательном режиме пересылаться во множество подсетей. Более эффективный метод, который мы рекомендуем, заключается в том, что хост сначала определяет свой адрес (возможно с помощью протокола RARP, описанного в [4]), а после этого передает запрос ICMP по адресу 255.255.255.255:

```
Source address:          36.40.0.123
Destination address:     255.255.255.255
Protocol:                ICMP = 1
Type:                    Address Format Request = A1
Code:                    0
```

Шлюз в этом случае может напрямую ответить запрашивающему хосту.

Предположим, что 36.40.0.123 является бездисковой станцией, которая не знает даже свой номер хоста. Она может передать дейтаграмму:

```
Source address:          0.0.0.0
Destination address:     255.255.255.255
Protocol:                ICMP = 1
Type:                    Address Format Request = A1
Code:                    0
```

36.40.0.62 услышит эту дейтаграмму, на которую ему следует ответить дейтаграммой:

```
Source address:          36.40.0.62
Destination address:     36.40.255.255
Protocol:                ICMP = 1
Type:                    Address Format Reply = A2
Code:                    8
```

Отметим, что шлюз использует максимально «узкий» широковещательный адрес для отклика (передача отклика по адресу 36.255.255.255 приведет к рассылке отклика во множество подсетей в дополнение к той, которой он предназначен). Но даже в этом случае излишнее использование широковещания приведет к неоправданному росту широковещательной нагрузки на все хосты подсети, поэтому рекомендуется использовать «анонимный» адрес отправителя запросов как можно реже.

Если широковещание не разрешено, мы предполагаем, что хост имеет «встроенную» информацию о соседних шлюзах. Таким образом, 36.40.0.123 может передать дейтаграмму:

```
Source address:          36.40.0.123
Destination address:     36.40.0.62
Protocol:                ICMP = 1
Type:                    Address Format Request = A1
Code:                    0
```

36.40.0.62 следует отвечать на эту дейтаграмму, как в предыдущем случае.

Литература

1. D.R. Boggs, J.F. Shoch, E.A. Taft, and R.M. Metcalfe. "Pup: An Internetwork Architecture." IEEE Transactions on Communications COM-28, 4, pp612-624, April 1980.

2. David D. Clark. Names, Addresses, Ports, and Routes. RFC-814, MIT-LCS, July 1982.

3. Yogan K. Dalal and Robert M. Metcalfe. "Reverse Path Forwarding of Broadcast Packets." Comm. ACM 21, 12, pp1040-1048, December 1978.
4. Ross Finlayson, Timothy Mann, Jeffrey Mogul, Marvin Theimer. A Reverse Address Resolution Protocol. RFC-903, Stanford University, June 1984.
5. R.M. Metcalfe and D.R. Boggs. "Ethernet: Distributed Packet Switching for Local Computer Networks." Comm. ACM 19, 7, pp395-404, July 1976. Also CSL-75-7, Xerox Palo Alto Research Center, reprinted in CSL-80-2.
6. Jeffrey Mogul. Broadcasting Internet Datagrams. RFC-919, Stanford University, October 1984.
7. David Plummer. An Ethernet Address Resolution Protocol. RFC-826, Symbolics, September 1982.
8. Jon Postel. Internet Protocol. RFC-791, USC-ISI, September 1981.
9. Jon Postel. Internet Control Message Protocol. RFC-792, USC-ISI, September 1981.

Перевод на русский язык

Николай Малых
nmalykh@gmail.com