

**Сети TCP / IP**  
**Протокол разрешения адресов**  
**Система DNS**

**БГАРФ, кафедра ИБ**

**Зензин Александр**  
**Степанович, к.т.н.**  
**Copyright © 2017**



# Обзор

---

1. Протокол разрешения адресов
2. Протокол Proxy-ARP
3. Система DNS
  - 3.1. Плоские символьные имена
  - 3.2. Иерархические символьные имена
  - 3.3. Схема работы DNS
  - 3.4. Обратная зона
4. Протокол DHCP
  - 4.1. Режим DHCP
  - 4.2. Алгоритм динамического назначения адресов

### Отображение IP-адресов на локальные адреса

Одной из главных задач, которая ставилась при создании протокола IP, являлось обеспечение совместной согласованной работы в сети, состоящей из подсетей, в общем случае использующих разные сетевые технологии. Взаимодействие технологии TCP/IP с локальными технологиями подсетей происходит многократно при перемещении IP-пакета по составной сети. На каждом маршрутизаторе протокол IP определяет, какому следующему маршрутизатору в этой сети надо направить пакет. В результате решения этой задачи протоколу IP становится известен IP-адрес интерфейса следующего маршрутизатора (или конечного узла, если эта сеть является сетью назначения). Чтобы локальная технология сети смогла доставить пакет на следующий маршрутизатор, необходимо:

- ❑ упаковать пакет в кадр соответствующего для данной сети формата (например, Ethernet);
- ❑ снабдить данный кадр локальным адресом следующего маршрутизатора.

Решением этих задач, как уже отмечалось, занимается уровень сетевых интерфейсов стека TCP/IP.

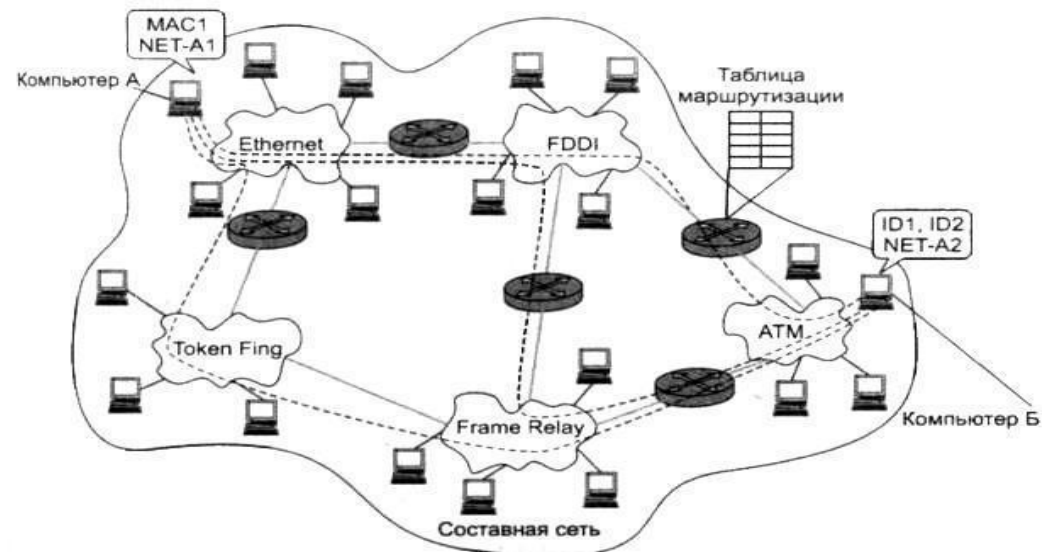


Рис. 2. Пример составной сети

## Протокол разрешения адресов

Как уже было сказано, никакой функциональной зависимости между локальным адресом и его IP-адресом не существует, следовательно, единственный способ установления соответствия — *ведение таблиц*. В результате конфигурирования сети каждый интерфейс «знает» свои IP-адрес и локальный адрес, что можно рассматривать как таблицу, состоящую из одной строки. Проблема состоит в том, как *организовать обмен* имеющейся информацией между узлами сети.

Для определения локального адреса по IP-адресу используется протокол разрешения адресов (Address Resolution Protocol, ARP). Протокол разрешения адресов реализуется различным образом в зависимости от того, работает ли в данной сети протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещания или же какой-либо из протоколов глобальной сети (Frame Relay, ATM), которые, как правило, не поддерживают широковещательный доступ.

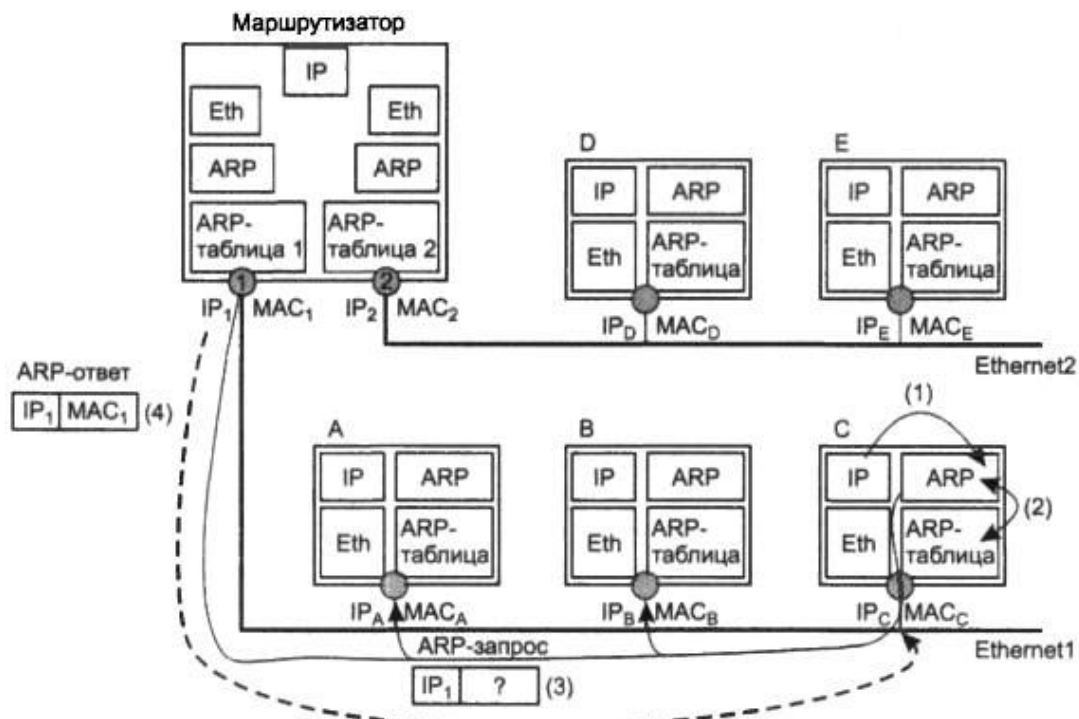
*Рассмотрим работу протокола ARP в локальных сетях с широковещанием.*

На рисунке показан фрагмент IP-сети, включающий две сети — Ethernet 1 (из трех конечных узлов A, B и C) и Ethernet 2 (из двух конечных узлов D и E). Сети подключены соответственно к интерфейсам 1 и 2 маршрутизатора. Каждый сетевой интерфейс имеет IP-адрес и MAC-адрес. Пусть в какой-то момент IP-модуль узла C направляет пакет узлу D. Протокол IP узла C определил IP-адрес интерфейса следующего маршрутизатора — это IP<sub>1</sub>. Теперь, прежде чем упаковать пакет в кадр Ethernet и направить его маршрутизатору, необходимо определить соответствующий MAC-адрес. Для решения этой задачи протокол IP обращается к протоколу ARP. Протокол ARP поддерживает на каждом интерфейсе сетевого адаптера или маршрутизатора отдельную ARP-таблицу, в которой в ходе функционирования сети накапливается информация о соответствии между IP-адресами и MAC-адресами других интерфейсов данной сети.

## Протокол разрешения адресов

Первоначально, при включении компьютера или маршрутизатора в сеть все его ARP-таблицы пусты.

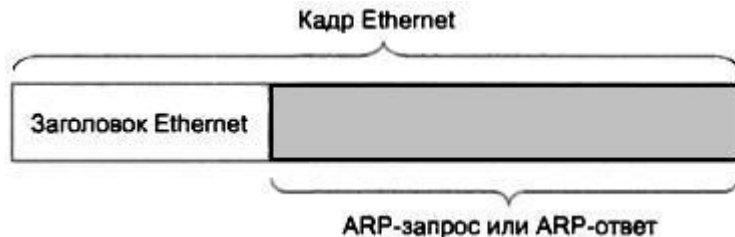
1. На первом шаге происходит передача от протокола IP протоколу ARP примерно такого сообщения: «Какой MAC-адрес имеет интерфейс с адресом  $IP_1$ ».
2. Работа протокола ARP начинается с просмотра собственной ARP-таблицы. Предположим, что среди содержащихся в ней записей отсутствует запрашиваемый IP-адрес.
3. В этом случае исходящий IP-пакет, для которого оказалось невозможным определить локальный адрес из ARP-таблицы, запоминается в буфере, а протокол ARP формирует ARP-запрос, вкладывает его в кадр протокола Ethernet и широковещательно рассылает.
4. Все интерфейсы сети Ethernet1 получают ARP-запрос и направляют его «своему» протоколу ARP. ARP сравнивает указанный в запросе адрес  $IP_1$  с IP-адресом интерфейса, на который поступил этот запрос. Протокол ARP, который констатировал совпадение (ARP маршрутизатора 1), формирует ARP-ответ.



В ARP-ответе маршрутизатор указывает локальный адрес MAC<sub>1</sub> своего интерфейса и отправляет его запрашивающему узлу (в данном примере узлу C), используя его локальный адрес. Широковещательный ответ в этом случае не требуется, так как формат ARP-запроса предусматривает поля локального и сетевого адресов отправителя. Заметим, что зона распространения ARP-запросов ограничивается сетью Ethernet1, так как на пути широковещательных кадров барьером стоит маршрутизатор.

## Протокол разрешения адресов

На рисунке показан кадр Ethernet с вложенным в него ARP-сообщением. ARP-запросы и ARP-ответы имеют один и тот же формат. В табл. 12 (справа) в качестве примера приведены значения полей реального **ARP-запроса**, переданного по сети Ethernet.



Поле	Значение
Тип сети	1(0x1 )
Тип протокола	2048 (0x800)
Длина локального адреса	6(0x6)
Длина сетевого адреса	4(0x4)
Операция	1(0x1)
Локальный адрес отправителя	008048EB7E60
Сетевой адрес отправителя	194.85.135.75
Локальный (искомый ) адрес получателя	000000000000
Сетевой адрес получателя	194.85.135.65

В поле типа сети для сетей Ethernet указывается значение 1. Поле типа протокола позволяет использовать протокол ARP не только с протоколом IP, но и с другими сетевыми протоколами. Для IP значение этого поля равно 0x0800. Длина локального адреса для протокола Ethernet равна 6 байт, а длина IP-адреса — 4 байта. В поле операции для ARP-запросов указывается Значение 1, для ARP-ответов — значение 2.

Из этого запроса видно, что в сети Ethernet узел с IP-адресом 194.85.135.75 пытается определить, какой MAC-адрес имеет другой узел той же сети, сетевой адрес которого 194.85.135.65. Поле искомого локального адреса заполнено нулями.

## Протокол разрешения адресов

Ответ присылает узел, опознавший свой IP-адрес. Если в сети нет машины с искомым IP-адресом, то ARP-ответа не будет. Протокол IP уничтожает IP-пакеты, направляемые по этому адресу. В следующей табл. 2 показаны значения полей **ARP-ответа**, который мог бы поступить на приведенный выше ARP-запрос.

Поле	Значение
Тип сети	1(0x1)
Тип протокола	2048 (0x800)
Длина локального адреса	6(0x6)
Длина сетевого адреса	4(0x4)
Операция	2(0x1)
Локальный адрес отправителя	00E0F77F1920
Сетевой адрес отправителя	194.85.135.65
Локальный (искомый) адрес получателя	008048EB7E60
Сетевой адрес получателя	194.85.135.75

В результате обмена ARP-сообщениями и модуль IP, пославший запрос с интерфейса, имеющего адрес 194.85.135.75, определил, что IP-адресу 194.85.135.65 соответствует MAC-адрес 00E0F77F1920. Этот адрес затем помещается в заголовок кадра Ethernet, ожидавшего отправления IP-пакета.

Чтобы уменьшить число ARP-обращений в сети, найденное соответствие между IP-адресом и MAC-адресом сохраняется в ARP-таблице соответствующего интерфейса, в данном случае — это запись:

194.85.135.65 - 00E0F77F1920

Данная запись в ARP-таблице появляется автоматически, спустя несколько миллисекунд после того, как модуль ARP проанализирует ARP-ответ.

## Протокол разрешения адресов

Теперь, если вдруг вновь возникнет необходимость послать пакет по адресу 194.85.135.65, то протокол IP прежде, чем посылать широковещательный запрос, проверит, нет ли уже такого адреса в ARP-таблице.

ARP-таблица пополняется не только за счет поступающих на данный интерфейс ARP-ответов, но и в результате извлечения полезной информации из широковещательных ARP-запросов. Действительно, в каждом запросе, как это видно из табл. 1 и 2, содержатся IP-адрес и MAC-адрес отправителя. Все интерфейсы, получившие этот запрос, могут поместить информацию о соответствии локального и сетевого адресов отправителя в собственную ARP-таблицу. В частности, все узлы, получившие ARP-запрос (см. табл. 1), могут пополнить свою ARP-таблицу записью:

194.85.135.75 - 008048EB7E60

Таким образом, вид ARP-таблицы, в которую входе работы сети были добавлены две упомянутые нами записи, иллюстрирует табл. 3.

IP-адрес	MAC-адрес	Тип записи
194.85.135.65	00E0F77F1920	Динамический
194.85.135.75	008048EB7E60	Динамический
194.85.60.21	008048EB7567	Статический

В ARP-таблицах существует два типа записей : динамические и статические. Статические записи создаются вручную с помощью утилиты *agr* и не имеют срока устаревания, точнее, они существуют до тех пор, пока компьютер или маршрутизатор остается включенным.

Динамические записи должны периодически обновляться. Если запись не обновлялась в течение определенного времени (порядка нескольких минут), то она исключается из таблицы.



## Протокол разрешения адресов

Таким образом, в ARP-таблице содержатся записи не обо всех узлах сети, а только о тех, которые активно участвуют в сетевых операциях. Поскольку такой способ хранения информации называют кэшированием, ARP-таблицы иногда называют *ARP-кэшем*.

Совсем другой способ разрешения адресов используется в *глобальных сетях*, в которых не поддерживается широковещательная рассылка. Здесь администратору сети чаще всего приходится вручную формировать и помещать на какой-либо сервер ARP-таблицы, в которых он задает, например, соответствие IP-адресов адресам X.25, имеющих для протокола IP смысл локальных адресов. В то же время сегодня наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети.

При таком централизованном подходе вручную нужно задать для всех узлов и маршрутизаторов только IP-адрес и локальный адрес выделенного для этих целей маршрутизатора.

При включении каждый узел и маршрутизатор регистрирует свои адреса в выделенном маршрутизаторе. Всякий раз, когда возникает необходимость определения по IP-адресу и автоматически получают ответ без участия администратора. Работающий таким образом маршрутизатор называют *ARP-сервером*.

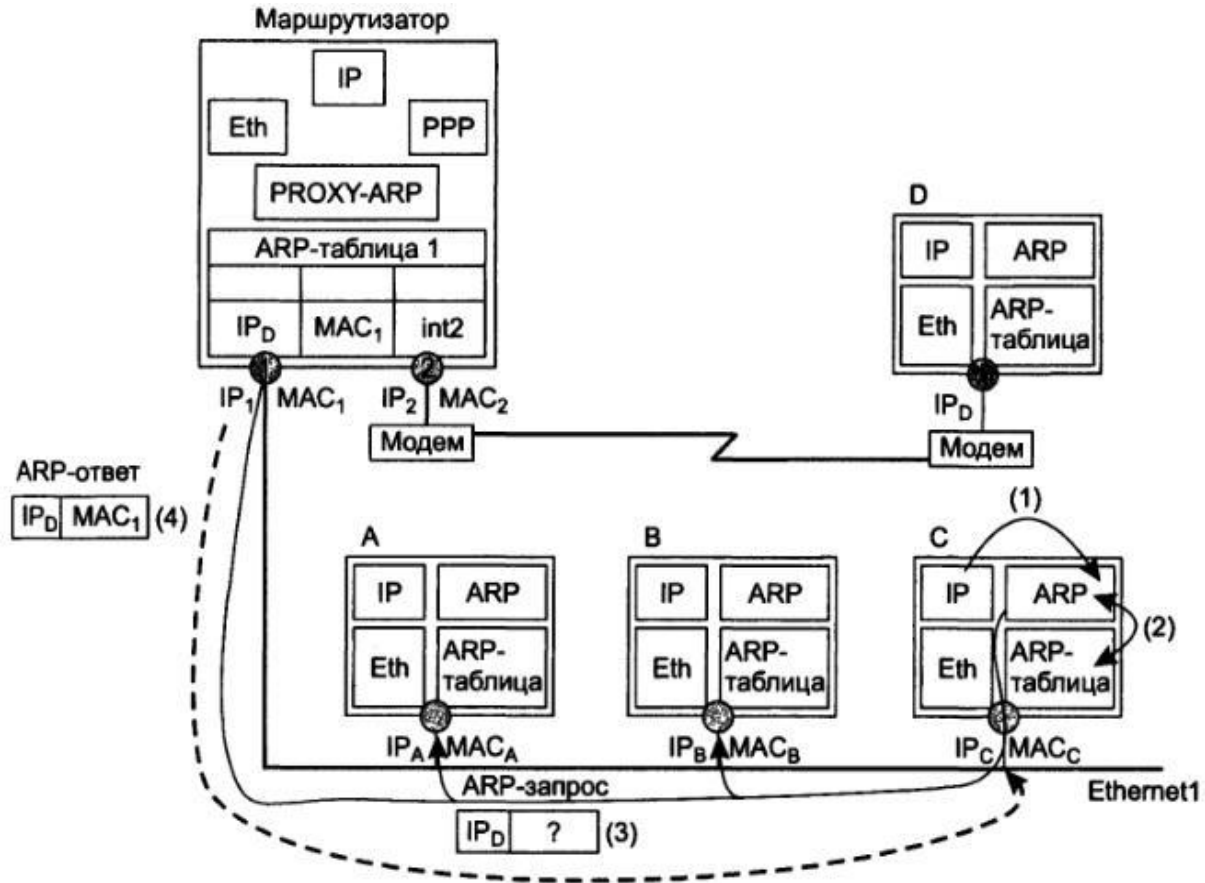
В некоторых случаях возникает обратная задача — нахождение IP-адреса по известному локальному адресу. Тогда в действие вступает реверсивный протокол разрешения адресов (Reverse Address Resolution Protocol, RARP). Этот протокол используется, например, при старте бездисковых станций, не знающих в начальный момент времени своего IP-адреса, но знающих MAC-адрес своего сетевого адаптера.

## Протокол Proxy-ARP

Протокол Proxy-ARP — это одна из разновидностей протокола ARP, позволяющая отображать IP-адреса на аппаратные адреса в сетях, поддерживающих широковещание, даже в тех случаях, когда искомым узел находится за пределами данного домена коллизий.

На рисунке показана сеть, один из конечных узлов которой (компьютер D) работает в режиме удаленного узла.

Конечный узел в таком режиме обладает всеми возможностями компьютеров, работающих в основной части сети Ethernet, в частности он имеет IP-адрес ( $IP_D$ ), относящийся к той же сети. Для всех конечных узлов сети Ethernet особенности подключения удаленного узла (наличие модемов, коммутируемая связь, протокол PPP) абсолютно прозрачны — они взаимодействуют с ним обычным образом. Чтобы такой режим взаимодействия стал возможным, среди прочего, необходим протокол Proxy-ARP. Поскольку удаленный узел подключен к сети по протоколу PPP, то он, очевидно, не имеет MAC-адреса.



Пусть приложение, работающее, например, на компьютере C, решает послать пакет компьютеру D.

## Протокол Proxu-ARP

Ему известен IP-адрес узла назначения ( $IP_D$ ), однако, как мы уже не раз отмечали, для передачи пакета по сети Ethernet его необходимо упаковать в кадр Ethernet и снабдить MAC-адресом. Для определения MAC-адреса IP-протокол узла С обращается к протоколу ARP, который посылает широковещательное сообщение с ARP-запросом. Если бы в этой сети на маршрутизаторе не был установлен протокол Proxu-ARP, на этот запрос не откликнулся бы ни один узел.

Однако протокол Proxu-ARP установлен на маршрутизаторе и работает следующим образом. При подключении к сети удаленного узла D в таблицу ARP-маршрутизатора заносится запись

$$IP_D - MAC_1 - int2,$$

которая означает, что:

- при поступлении ARP-запроса на маршрутизатор относительно адреса  $IP_D$  в ARP-ответ будет помещен аппаратный адрес  $MAC_1$ , соответствующий аппаратному адресу интерфейса 1 маршрутизатора;
- узел, имеющий адрес  $IP_D$ , подключен к интерфейсу 2 маршрутизатора.

В ответ на посланный узлом С широковещательный ARP-запрос откликается маршрутизатор с установленным протоколом Proxu-ARP. Он посылает «ложный» ARP-ответ, в котором на место аппаратного адреса компьютера D помещает собственный адрес  $MAC_1$ . Узел С, не подозревая «подвоха», посылает кадр с IP-пакетом по адресу  $MAC_1$ . Получив кадр, маршрутизатор с установленным протоколом Proxu-ARP «понимает», что он направлен не ему (в пакете указан чужой IP-адрес) и, следовательно, надо искать адресата в ARP-таблице. Из таблицы видно, что кадр надо направить узлу, подключенному ко второму интерфейсу.



## *Система DNS*

### *Плоские символьные имена*

---

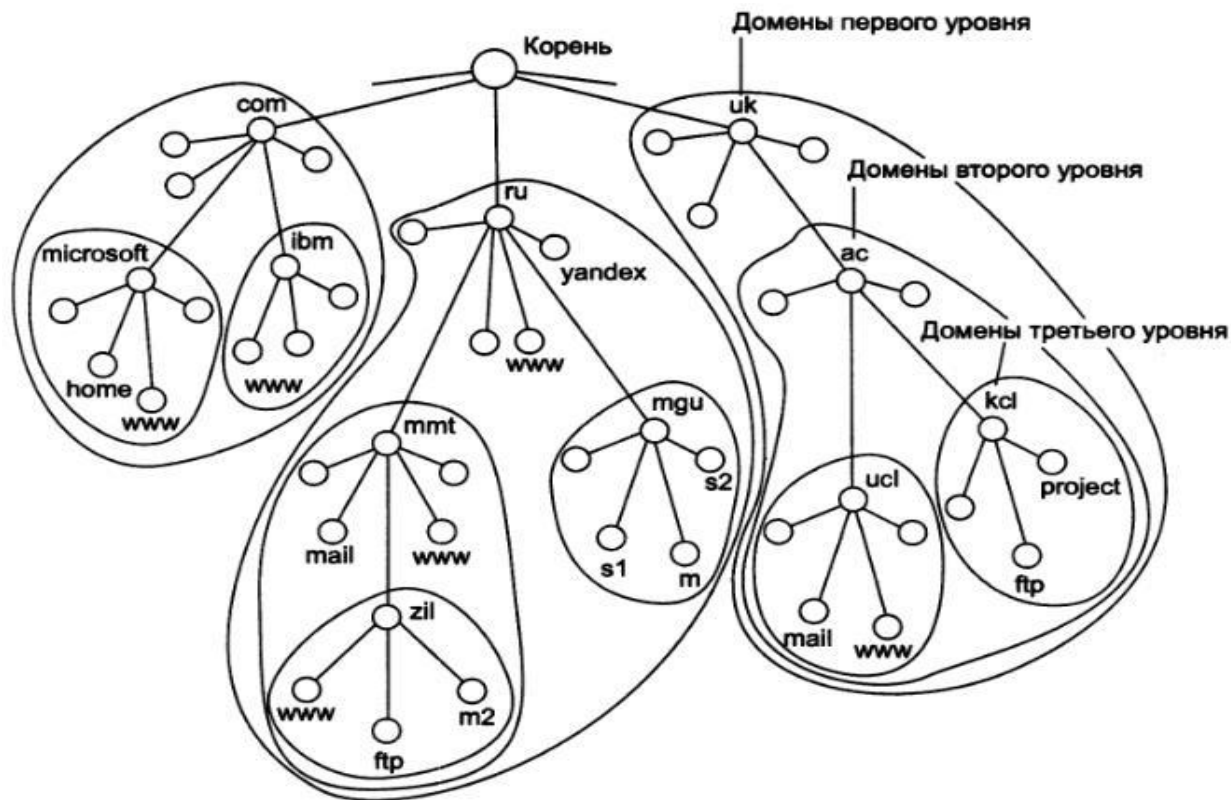
В операционных системах, которые первоначально разрабатывались для локальных сетей, таких как Novell NetWare, Microsoft Windows или IBM OS/2, пользователи всегда работали с символьными именами компьютеров. Так как локальные сети состояли из небольшого числа компьютеров, применялись так называемые плоские имена, состоящие из последовательности символов, не разделенных на части. Примерами таких имен являются: NW1\_1, mail2, MOSCOW\_SALES\_2. Для установления соответствия между символьными именами и MAC-адресами в этих операционных системах применялся механизм широковещательных запросов, подобный механизму запросов протокола ARP. Так, широковещательный способ разрешения имен реализован в протоколе NetBIOS, на котором были построены многие локальные ОС. Так называемые NetBIOS-имена стали на долгие годы одним из основных типов плоских имен в локальных сетях.

Для стека TCP/IP, рассчитанного в общем случае на работу в больших территориально распределенных сетях, подобный подход оказывается неэффективным.

# Система DNS

## Иерархические символьные имена

В стеке TCP/IP применяется доменная система имен, которая имеет иерархическую древовидную структуру, допускающую наличие в имени произвольного количества составных частей (рисунок ниже).



Иерархия доменных имен аналогична иерархии имен файлов, принятой во многих популярных файловых системах. Дерево имен начинается с корня, обозначаемого здесь точкой (.). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть имени и т. д.

# Система DNS

## Иерархические символьные имена

Младшая часть имени соответствует конечному узлу сети. В отличие от имен файлов, при записи которых сначала указывается самая старшая составляющая, затем составляющая более низкого уровня и т. д., запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяются друг от друга точкой. Например, в имени home.microsoft.com составляющая home является именем одного из компьютеров в домене microsoft.com.

Разделение имени на части позволяет разделить административную ответственность за назначение уникальных имен между различными людьми или организациями в пределах своего уровня иерархии. Так, для примера, приведенного выше, один человек может нести ответственность за то, чтобы все имена с окончанием «ru» имели уникальную следующую вниз по иерархии часть. То есть все имена типа [www.ru](http://www.ru), mall.mmt.ru или m2.zil.mmt.ru отличаются второй по старшинству частью.

Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций между организациями, отвечающими за имена одного уровня иерархии. Очевидно, что должна существовать одна организация, отвечающая за назначение имен верхнего уровня иерархии. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен (domain). Например, имена [www.zil.mmt.ru](http://www.zil.mmt.ru), ftp.zil.mmt.ru, yandex.ru и sl.mgu.ru входят в домен ru, так как все они имеют одну общую старшую часть — имя ru. Другим примером является домен mgu.ru. Из представленных на рис. 1 имен в него входят имена s1.mgu.ru, s2.mgu.ru и rn.mgu.ru. Этот домен образуют имена, у которых две старшие части равны mgu.ru. Администратор домена mgu.ru несет ответственность за уникальность имен следующего уровня, входящих в домен, то есть имен s1, s2 и m.

Образованные домены s1.mgu.ru, s2.mgu.ru и rn.mgu.ru являются поддоменами домена mgu.ru, так как имеют общую старшую часть имени. Часто поддомены для краткости называют только младшей частью имени, то есть в нашем случае поддоменами являются s1, s2 и rn.

Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой в доменной системе имен различают краткие, относительные и полные доменные имена. Краткое доменное имя — это имя конечного узла сети: хоста или порта маршрутизатора. Краткое имя — это лист дерева имен. Относительное доменное имя — это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Например, [www.zil](#) — это относительное имя. Полное доменное имя (Fully Qualified Domain Name, FQDN) включает составляющие всех уровней иерархии, начиная от краткого имени и кончая корневой точкой: [www.zil.mmt.ru.](#)

Корневой домен управляется центральными органами Интернета, в частности уже упоминавшейся нами организацией ICANN. Домены верхнего уровня назначаются для каждой страны, а также для различных типов организаций. Имена этих доменов должны следовать международному стандарту ISO 3166.


Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, например ru (Россия), uk (Великобритания), fi (Финляндия), us (Соединенные Штаты), а для различных типов организаций, например, следующие обозначения:

- com — коммерческие организации (например, microsoft.com);
- edu — образовательные организации (например, mit.edu);
- gov — правительственные организации (например, nsf.gov);
- org — некоммерческие организации (например, fidonet.org);
- net — сетевые организации (например, nsf.net).

Каждый домен администрирует отдельная организация, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Чтобы получить доменное имя, необходимо зарегистрироваться в какой-либо организации, которой делегированы полномочия по распределению имен доменов.

Доменная система имен реализована в Интернете, но она может работать и как автономная система имен в любой крупной корпоративной сети, которая хотя и использует стек TCP/IP, никак не связана с Интернетом.





## Система DNS

### Схема работы DNS

---

Широковещательный способ установления соответствия между символьными именами и локальными адресами, подобный протоколу ARP, хорошо работает только в небольшой локальной сети, не разделенной на подсети. В крупных сетях, где возможность всеобщей широковещательной рассылки не поддерживается, нужен другой способ разрешения символьных имен. Хорошей альтернативой широковещательной рассылке является применение централизованной службы, поддерживающей соответствие между различными типами адресов всех компьютеров сети. Например, компания Microsoft для своей корпоративной операционной системы Windows NT разработала централизованную службу WINS, которая поддерживала базу данных NetBIOS-имен и соответствующих им IP-адресов.

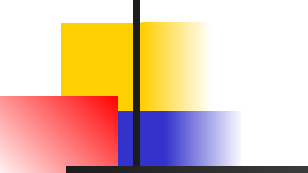
В сетях TCP/IP соответствие между доменными именами и IP-адресами может устанавливаться средствами как локального хоста, так и централизованной службы.

На раннем этапе развития Интернета на каждом хосте вручную создавался текстовый файл с известным именем `hosts.txt`. Этот файл состоял из некоторого количества строк, каждая из которых содержала одну пару «доменное имя — IP-адрес», например:

```
rhino.acme.com — 102.54.94.97
```

По мере роста Интернета файлы `hosts.txt` также увеличивались в объеме, и *создание масштабируемого решения* для разрешения имен стало необходимостью.

Таким решением стала централизованная **служба DNS** (Domain Name System — система доменных имен), основанная на распределенной базе отображений «доменное имя — IP-адрес». Служба DNS использует в своей работе DNS-серверы и DNS-клиенты. DNS-серверы поддерживают распределенную базу отображений, а DNS-клиенты обращаются к серверам с запросами об отображении разрешения доменного имени на IP-адрес.



## Система DNS

### Схема работы DNS

Служба DNS использует текстовые файлы почти такого же формата, как и файл hosts, и эти файлы администратор также подготавливает вручную. Однако служба DNS опирается на иерархию доменов, и каждый DNS-сервер хранит только часть имен сети, а не все имена, как это происходит при использовании файлов hosts. При росте количества узлов в сети проблема масштабирования решается созданием новых доменов и поддоменов имен и добавлением в службу DNS новых серверов.

Для каждого домена имен создается свой DNS-сервер. На серверах применяют два подхода к распределению имен. В первом случае сервер может хранить отображения «доменное имя — IP-адрес» для всего домена, включая все его поддомены. Однако такое решение оказывается плохо масштабируемым, так как при добавлении новых поддоменов нагрузка на этот сервер может превысить его возможности. Чаще используется другой подход, когда сервер домена хранит только имена, которые заканчиваются на следующем ниже уровне иерархии по сравнению с именем домена. (Аналогично каталогу файловой системы, который содержит записи о файлах и подкаталогах, непосредственно в него «входящих».) Именно при такой организации службы DNS нагрузка по разрешению имен распределяется более-менее равномерно между всеми DNS-серверами сети. Например, в первом случае DNS-сервер домена mmt.ru будет хранить отображения для всех имен, заканчивающихся на mmt.ru (www.zil.mmt.ru, ftp.zil.mmt.ru, mail.mmt.ru и т. д.). Во втором случае этот сервер хранит отображения только имен типа mail.mmt.ru, [www.mmt.ru](http://www.mmt.ru), а все остальные отображения должны храниться на DNS-сервере поддомена zil.

Каждый DNS-сервер помимо таблицы отображений имен содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой *IP-адреса соответствующих серверов*. Для обслуживания корневого домена выделено несколько дублирующих друг друга DNS-серверов, IP-адреса которых широко известны (их можно узнать, например, в InterNIC).



## Система DNS

### Схема работы DNS

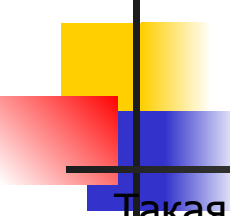
Процедура разрешения DNS-имени во многом аналогична процедуре поиска файловой системой адреса файла по его символьному имени. Действительно, в обоих случаях составное имя отражает иерархическую структуру организации соответствующих справочников — каталогов файлов или DNS-таблиц. Здесь домен и доменный DNS-сервер являются аналогом каталога файловой системы. Для доменных имен, так же как и для символьных имен файлов, характерна независимость именованного от физического местоположения.

Процедура поиска адреса файла по символьному имени заключается в последовательном просмотре каталогов, начиная с корневого. При этом предварительно проверяются кэш и текущий каталог. Для определения IP-адреса по доменному имени также необходимо просмотреть все DNS-серверы, обслуживающие цепочку поддоменов, входящих в имя хоста, начиная с корневого домена.

Существенным отличием файловой системы от службы DNS является то, что первая расположена на одном компьютере, а вторая по своей природе является распределенной.

Существует две основные схемы разрешения DNS-имен. В первом варианте работу по поиску IP-адреса координирует DNS-клиент:

1. DNS-клиент обращается к корневому DNS-серверу с указанием полного доменного имени.
2. DNS-сервер отвечает клиенту, указывая адрес следующего DNS-сервера, обслуживающего домен верхнего уровня, заданный в следующей старшей части запрошенного имени.
3. DNS-клиент делает запрос следующего DNS-сервера, который отсылает его к DNS-серверу нужного поддомена и т. д., пока не будет найден DNS-сервер, в котором хранится соответствие запрошенного имени IP-адресу. Этот сервер дает окончательный ответ клиенту.



## Система DNS

### Схема работы DNS

---

Такая процедура разрешения имени называется *нерекурсивной*, когда клиент сам итеративно выполняет последовательность запросов к разным серверам имен. Эта схема загружает клиента достаточно сложной работой, и она применяется редко.

Во втором варианте реализуется *рекурсивная* процедура:

1. DNS-клиент запрашивает локальный DNS-сервер, то есть тот сервер, обслуживающий поддомен, которому принадлежит имя клиента.
2. Далее возможны два варианта действий:
  - если локальный DNS-сервер знает ответ, то он сразу же возвращает его клиенту (это может произойти, когда запрошенное имя входит в тот же поддомен, что и имя клиента, или когда сервер уже узнавал данное соответствие для другого клиента и сохранил его в своем кэше);
  - если локальный сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу и т. д. точно так же, как это делал клиент в предыдущем варианте, а получив ответ, передает его клиенту, который все это время просто ждет его от своего локального DNS-сервера.

В этой схеме клиент *перепоручает* работу своему серверу, именно поэтому схема называется *рекурсивной*, или *косвенной*. Практически все DNS-клиенты используют рекурсивную процедуру.

Для ускорения поиска IP-адресов DNS-серверы широко применяют кэширование проходящих через них ответов. Чтобы служба DNS могла оперативно отрабатывать изменения, происходящие в сети, ответы кэшируются на относительно короткое время — обычно от нескольких часов до нескольких дней.



## Система DNS

### Обратная зона

Служба DNS предназначена не только для нахождения IP-адреса по имени хоста, но и для решения обратной задачи — нахождению DNS-имени по известному IP-адресу.

Многие программы и утилиты, пользующиеся службой DNS, пытаются найти имя узла по его адресу в том случае, когда пользователем задан только адрес (или этот адрес программа узнала из пришедшего пакета). Обратная запись не всегда существует даже для тех адресов, для которых есть прямые записи. Ее могут просто забыть создать или же ее создание требует дополнительной оплаты. Обратная задача решается в Интернете путем организации так называемых *обратных зон*.

Обратная зона — это система таблиц, которая хранит соответствие между IP-адресами и DNS-имена хостов некоторой сети. Для организации распределенной службы и использования для поиска имен того же программного обеспечения, что и для поиска адресов, применяется оригинальный подход, связанный с представлением IP-адреса в виде DNS-имени.

Первый этап преобразования заключается в том, что составляющие IP-адреса интерпретируются как составляющие DNS-имени. Например, адрес 192.31.106.0 рассматривается как состоящий из старшей части, соответствующей домену 192, затем идет домен 31, в который входит домен 106.

Далее, учитывая, что при записи IP-адреса старшая часть является самой левой частью адреса, а при записи DNS-имени — самой правой, то составляющие в преобразованном адресе указываются в обратном порядке, то есть для данного примера — 106.31.192.

Для хранения соответствия всех адресов, начинающихся, например, с числа 192, заводится зона 192 со своими серверами имен.

Для записей о серверах, поддерживающих старшие в иерархии обратные зоны, создана специальная зона in-addr.arpa, поэтому полная запись для использованного в примере адреса выглядит так:

106.31.192.in-addr.arpa.

Серверы для обратных зон используют файлы баз данных, не зависящие от файлов основных зон, в которых имеются записи о прямом соответствии тех же имен и адресов. Такая организация данных может приводить к несогласованности, так как одно и то же соответствие вводится в файлы дважды.

### **Протокол DHCP**

Для нормальной работы сети каждому сетевому интерфейсу компьютера и маршрутизатора должен быть назначен IP-адрес.

Процедура присвоения адресов происходит в ходе конфигурирования компьютеров и маршрутизаторов. Назначение IP-адресов может происходить вручную в результате выполнения процедуры конфигурирования интерфейса, для компьютера сводящейся, например, к заполнению системы экранных форм. При этом администратор должен помнить, какие адреса из имеющегося множества он уже использовал для других интерфейсов, а какие еще свободны. При конфигурировании помимо IP-адресов сетевых интерфейсов (и соответствующих масок) устройству сообщается ряд других конфигурационных параметров. При конфигурировании администратор должен назначить клиенту не только IP-адрес, но и другие параметры стека TCP/IP, необходимые для его эффективной работы, например маску и IP-адрес маршрутизатора по умолчанию, IP-адрес DNS-сервера, доменное имя компьютера и т. п. Даже при не очень большом размере сети эта работа представляет для администратора утомительную процедуру.

## Протокол DHCP

### Режим DHCP

Протокол **динамического конфигурирования хостов** (Dynamic Host Configuration Protocol, DHCP) автоматизирует процесс конфигурирования сетевых интерфейсов, обеспечивая отсутствие дублирования адресов за счет централизованного управления их распределением. Работа DHCP описана в RFC 2131 и 2132.

Протокол DHCP работает в соответствии с моделью клиент-сервер. Во время старта системы компьютер, являющийся DHCP-клиентом, посылает в сеть широковещательный запрос на получение IP-адреса. DHCP-сервер откликается и посылает сообщение-ответ, содержащее IP-адрес и некоторые другие конфигурационные параметры.

При этом сервер DHCP может работать в разных режимах, включая:

- ручное назначение статических адресов;
- автоматическое назначение статических адресов;
- автоматическое распределение динамических адресов.

Во всех режимах работы администратор при конфигурировании DHCP-сервера сообщает ему один или несколько диапазонов IP-адресов, причем все эти адреса относятся к одной сети, то есть имеют одно и то же значение в поле номера сети.

В **ручном** режиме администратор, помимо пула доступных адресов, снабжает DHCP-сервер информацией о жестком соответствии IP-адресов физическим адресам или другим идентификаторам клиентских узлов. DHCP-сервер, пользуясь этой информацией, всегда выдаст определенному DHCP-клиенту один и тот же назначенный ему администратором IP-адрес (а также набор других конфигурационных параметров).

В режиме **автоматического назначения** статических адресов DHCP-сервер самостоятельно без вмешательства администратора произвольным образом выбирает клиенту IP-адрес из пула наличных IP-адресов.



## Протокол DHCP

### Режим DHCP

---

Адрес дается клиенту из пула в постоянное пользование, то есть между идентифицирующей информацией клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первого назначения DHCP-сервером IP-адреса клиенту. При всех последующих запросах сервер возвращает клиенту тот же самый IP-адрес.

При **динамическом** распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, называемое сроком аренды. Когда компьютер, являющийся DHCP-клиентом, удаляется из подсети, назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс.

Это дает возможность впоследствии повторно использовать этот IP-адрес для назначения другому компьютеру. Таким образом, помимо основного преимущества DHCP — автоматизации рутинной работы администратора по конфигурированию стека TCP/IP на каждом компьютере, режим динамического распределения адресов в принципе позволяет строить IP-сеть, количество узлов в которой превышает количество имеющихся в распоряжении администратора IP-адресов.

### **ПРИМЕР**

Рассмотрим преимущества, которые дает динамическое распределение пула адресов на примере организации, в которой сотрудники значительную часть рабочего времени проводят вне офиса — дома или в командировках. Каждый из них имеет портативный компьютер, который во время пребывания в офисе подключается к корпоративной IP-сети. Возникает вопрос, сколько IP-адресов необходимо этой организации?





## Протокол DHCP

### Режим DHCP

---

Первый ответ — столько, скольким сотрудникам необходим доступ в сеть. Если их 500 человек, то каждому из них должен быть назначен IP-адрес и выделено рабочее место. То есть администрация должна получить у поставщика услуг адреса двух сетей класса С и оборудовать соответствующим образом помещение. Однако вспомним, что сотрудники в этой организации редко появляются в офисе, значит, большая часть ресурсов при таком решении будет простаивать.

Второй ответ — столько, сколько сотрудников обычно присутствует в офисе (с некоторым запасом). Если обычно в офисе работает не более 50 сотрудников, то достаточно получить у поставщика услуг пул из 64 адресов и установить в рабочем помещении сеть с 64-мя коннекторами для подключения компьютеров. Но возникает другая проблема — кто и как будет конфигурировать компьютеры, состав которых постоянно меняется?

Существует два пути. Во-первых, администратор (или сам мобильный пользователь) может конфигурировать компьютер вручную каждый раз, когда возникает необходимость подключения к офисной сети. Такой подход требует от администратора (или пользователей) большого объема рутинной работы, следовательно — это плохое решение. Гораздо привлекательнее выглядят возможности автоматического динамического назначения DHCP-адресов. Действительно, администратору достаточно один раз при настройке DHCP-сервера указать диапазон из 64 адресов, а каждый вновь прибывающий мобильный пользователь будет просто физически подключать в сеть свой компьютер, на котором запускается DHCP-клиент.

Он запросит конфигурационные параметры и автоматически получит их от DHCP-сервера. Таким образом, для работы 500 мобильных сотрудников достаточно иметь в офисной сети 64 IP-адреса и 64 рабочих места.

Администратор управляет процессом "конфигурирования сети, определяя два основных конфигурационных параметра DHCP-сервера: пул адресов, доступных к распределению, и срок аренды. Срок аренды диктует, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от DHCP-сервера. Срок аренды зависит от режима работы пользователей сети. Если это небольшая сеть учебного заведения, куда со своими компьютерами приходят многочисленные студенты для выполнения лабораторных работ, то срок аренды может быть равен длительности лабораторной работы. Если же это корпоративная сеть, в которой сотрудники предприятия работают на регулярной основе, то срок аренды может быть достаточно длительным — несколько дней или даже недель.

DHCP-сервер должен находиться в одной подсети с клиентами, учитывая, что клиенты посылают ему широковещательные запросы (рисунок ниже). Для снижения риска выхода сети из строя из-за отказа DHCP-сервера в сети иногда ставят резервный DHCP-сервер (такой вариант соответствует сети 1).

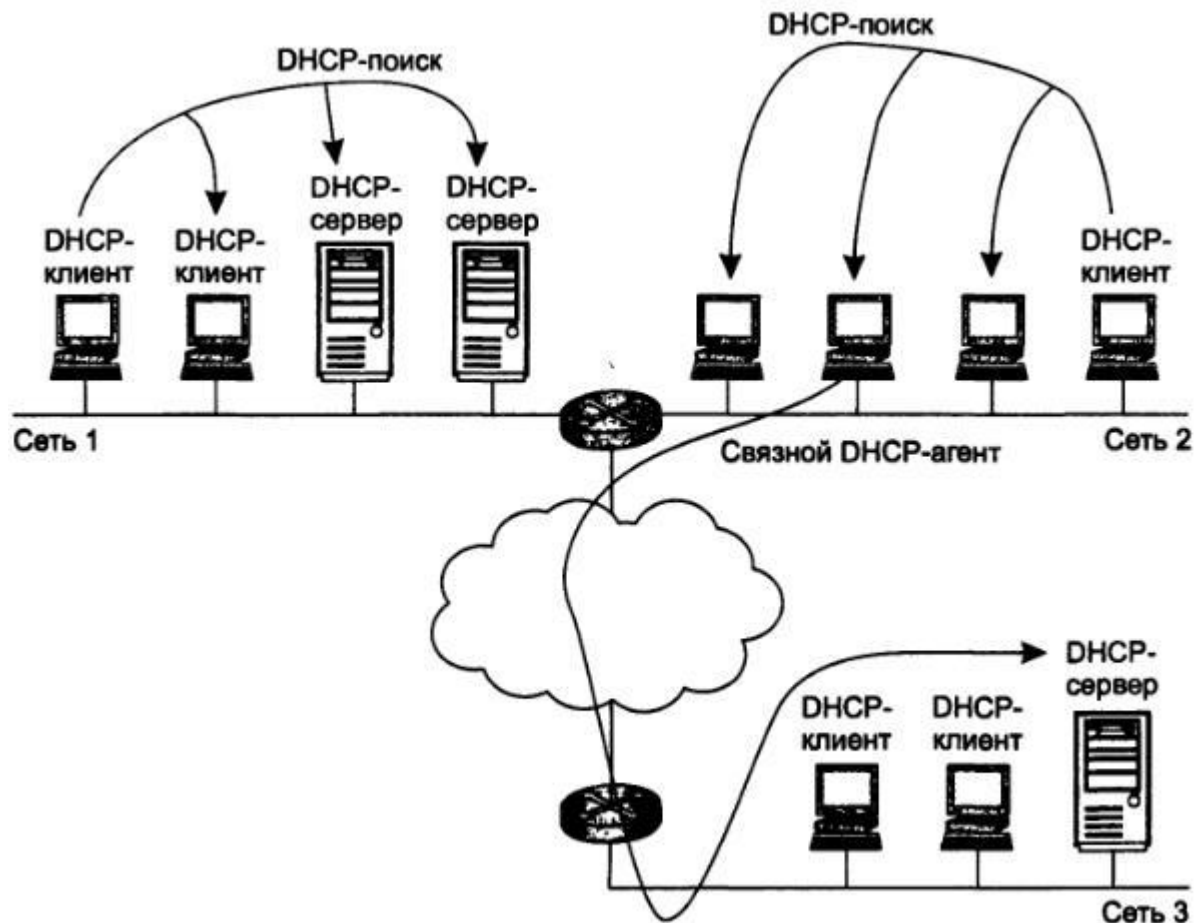
Иногда наблюдается и обратная картина: в сети нет ни одного DHCP-сервера. В этом случае его подменяет связной DHCP-агент — программное обеспечение, играющее роль посредника между DHCP-клиентами и DHCP-серверами (пример такого варианта — сеть 2). Связной агент переправляет запросы клиентов из сети 2 DHCP-серверу сети 3. Таким образом, один DHCP-сервер может обслуживать DHCP-клиентов нескольких разных сетей.

# Протокол DHCP

## Алгоритм динамического назначения адресов

Вот как выглядит упрощенная схема обмена сообщениями между клиентскими и серверными частями DHCP.

1. Когда компьютер включают, установленный на нем DHCP-клиент посылает ограниченное широковещательное сообщение DHCP-поиска (IP-пакет с адресом назначения, состоящим из одних единиц, который должен быть доставлен всем узлам данной IP-сети).



2. Находящиеся в сети DHCP-серверы получают это сообщение. Если в сети DHCP-серверы отсутствуют, то сообщение DHCP-поиска получает связной DHCP-агент. Он пересылает это сообщение в другую, возможно, значительно отстоящую от него сеть DHCP-серверу, IP-адрес которого ему заранее известен.
3. Все DHCP-серверы, получившие сообщение DHCP-поиска, посылают DHCP-клиенту, обратившемуся с запросом, свои DHCP-предложения. Каждое предложение содержит IP-адрес и другую конфигурационную информацию. (DHCP-сервер, находящийся в другой сети, посылает ответ через агента.)
4. DHCP-клиент собирает конфигурационные DHCP-предложения от всех DHCP-серверов. Как правило, он выбирает первое из поступивших предложений и отправляет в сеть широковещательный DHCP-запрос. В этом запросе содержатся идентификационная информация о DHCP-сервере, предложение которого принято, а также значения принятых конфигурационных параметров.
5. Все DHCP-серверы получают DHCP-запрос, и только один выбранный DHCP-сервер посылает положительную DHCP-квитанцию (подтверждение IP-адреса и параметров аренды), а остальные серверы аннулируют свои предложения, в частности возвращают в свои пулы предложенные адреса.
6. DHCP-клиент получает положительную DHCP-квитанцию и переходит в рабочее состояние.

Время от времени компьютер пытается обновить параметры аренды у DHCP-сервера. Первую попытку он делает задолго до истечения срока аренды, обращаясь к тому серверу, от которого он получил текущие параметры. Если ответа нет или ответ отрицательный, он через некоторое время снова посылает запрос. Так повторяется несколько раз, и если все попытки получить параметры у того же сервера оказываются безуспешными, клиент обращается к другому серверу. Если и другой сервер отвечает отказом, то клиент теряет свои конфигурационные параметры и переходит в режим автономной работы.

Также DHCP-клиент может по своей инициативе досрочно отказаться от выделенных ему параметров.

В сети, где адреса назначаются динамически, нельзя быть уверенным в адресе, который в данный момент имеет тот или иной узел. И такое непостоянство IP-адресов влечет за собой некоторые проблемы.

Во-первых, возникают сложности при преобразовании символьного доменного имени в IP-адрес. Действительно, представьте себе функционирование системы DNS, которая должна поддерживать таблицы соответствия символьных имен IP-адресам в условиях, когда последние меняются каждые два часа! Учитывая это обстоятельство, для серверов, к которым пользователи часто обращаются по символьному имени, назначают статические IP-адреса, оставляя динамические только для клиентских компьютеров. Однако в некоторых сетях количество серверов настолько велико, что их ручное конфигурирование становится слишком обременительным. Это привело к разработке усовершенствованной версии DNS (так называемой динамической системы DNS), в основе которой лежит согласование информационной адресной базы в службах DHCP и DNS.

Во-вторых, трудно осуществлять удаленное управление и автоматический мониторинг интерфейса (например, сбор статистики), если в качестве его идентификатора выступает динамически изменяемый IP-адрес.

Наконец, для обеспечения безопасности сети многие сетевые устройства могут блокировать (фильтровать) пакеты, определенные поля которых имеют некоторые заранее заданные значения. Другими словами, при динамическом назначении адресов усложняется фильтрация пакетов по IP-адресам.

Последние две проблемы проще всего решаются отказом от динамического назначения адресов для интерфейсов, фигурирующих в системах мониторинга и безопасности.