4.1 Основы функционирования протокола TCP/IP (IP-адрес, маска подсети, основной шлюз; деление на подсети с помощью маски подсети; введение в IP-маршрутизацию).

Адресация узлов в IP-сетях

В сетях TCP/IP принято различать адреса сетевых узлов трех уровней

- физический (или локальный) адрес узла (MAC-адрес сетевого адаптера или порта маршрутизатора); эти адреса назначаются производителями сетевого оборудования;
- IP-адрес узла (например, `192.168.0.1`), данные адреса назначаются сетевыми администраторами или Интернет-провайдерами;
- символьное имя (например, www.microsoft.com); эти имена также назначаются сетевыми администраторами компаний или Интернет-провайдерами.

Рассмотрим подробнее IP-адресацию.

Компьютеры или другие сложные сетевые устройства, подсоединенные к нескольким физическим сетям, имеют несколько IP-адресов — по одному на каждый сетевой интерфейс. Схема адресации позволяет проводить единичную, широковещательную и групповую адресацию. Таким образом, выделяют 3 типа IP-адресов.

1. *Unicast*-адрес (единичная адресация конкретному узлу) — используется в коммуникациях "один-к-одному".
2. Broadcast-адрес (широковещательный адрес, относящийся ко всем адресам подсети) — используется в коммуникациях "один-ко-всем". В этих адресах поле *идентификатора устройства* заполнено единицами. IP-адресация допускает широковещательную передачу, но не гарантирует ее — эта возможность зависит от конкретной физической сети. Например, в сетях Ethernet широковещательная передача выполняется с той же эффективностью, что и обычная передача данных, но есть сети, которые вообще не поддерживают такой тип передачи или поддерживают весьма ограничено.
3. Multicast-адрес (групповой адрес для многоадресной отправки пакетов) — используется в коммуникациях "один-ко-многим". Поддержка групповой адресации используется во многих приложениях, например, приложениях интерактивных конференций. Для групповой передачи рабочие станции и маршрутизаторы используют протокол IGMP, который предоставляет информацию о принадлежности устройств определенным группам.

Unicast-адреса.

Каждый сетевой интерфейс на каждом узле сети должен иметь уникальный *unicast*-адрес. IP-адрес имеет длину 4 байта (или 32 бита). Для удобства чтения адресов 32-битные числа разбивают на октеты по 8 бит, каждый октет переводят в десятичную систему счисления и при записи разделяют точками. Например, IP-адрес `11000000101010000000000000000001` записывается как `192.168.0.1`.

IP-адрес состоит из двух частей — идентификатор сети (префикс сети, Network ID) и идентификатор узла (номер устройства, Host ID). Такая схема приводит к двухуровневой адресной иерархии. Структура IP-адреса изображена на рис. 4.1.

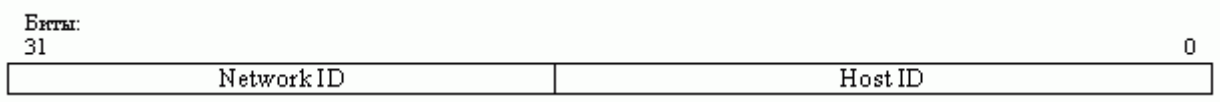


Рис. 4.1.
Идентификатор сети идентифицирует все узлы, расположенные на одном физическом или логическом сегменте сети, ограниченном IP-маршрутизаторами. Все узлы, находящиеся в одном сегменте должны иметь одинаковый идентификатор сети.

Идентификатор узла идентифицирует конкретный сетевой узел (сетевой адаптер рабочей станции или сервера, порт маршрутизатора). Идентификатор узла должен быть уникален для каждого узла внутри IP-сети, имеющей один идентификатор сети.

Таким образом, в целом IP-адрес будет уникален для каждого сетевого интерфейса всей сети TCP/IP.

Соотношение между идентификатором сети и идентификатором узла в IP-адресе определяется с помощью маски подсети (*Network mask*), которая имеет длину также 4 байта и также записывается в десятичной форме по 4 октета, разделенных точками. Старшие биты маски подсети, состоящие из 1, определяют, какие разряды IP-адреса относятся к идентификатору сети. Младшие биты маски, состоящие из 0, определяют, какие разряды IP-адреса относятся к идентификатору узла.

IP-адрес и маска подсети — минимальный набор параметров для конфигурирования протокола TCP/IP на сетевом узле.

Для обеспечения гибкости в присваивании адресов компьютерным сетям разработчики протокола определили, что адресное пространство IP должно быть разделено на три различных класса — A, B и C.

В дополнение к этим трем классам выделяют еще два класса. D — этот класс используется для групповой передачи данных. E — класс, зарезервированный для проведения экспериментов.

IP-адреса класса A.

Старший бит любого IP-адреса в сети класса A всегда равен 0. Идентификатор сети состоит из 8 бит, идентификатор узла — 24 бита. Маска подсети для узлов сетей класса A — 255.0.0.0. Структура IP-адресов класса A приведена на рис. 4.2.

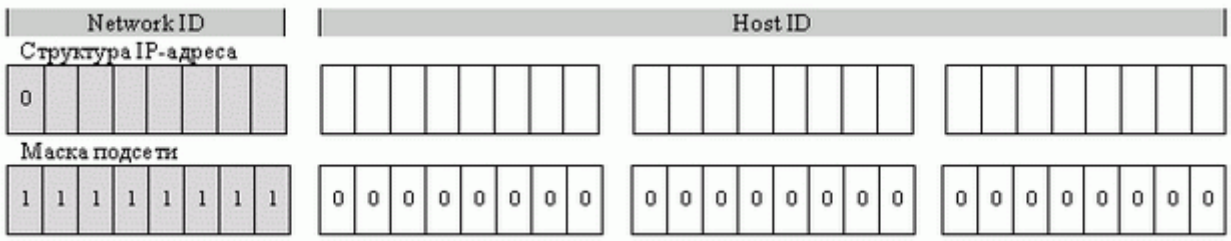


Рис. 4.2.
IP-адреса класса B.

Два старших бита любого IP-адреса в сети класса B всегда равны 10. Идентификатор сети состоит из 16 бит, идентификатор узла — 16 бит. Маска подсети для узлов сетей класса B —255.255.0.0. Структура IP-адресов класса B приведена на рис. 4.3.

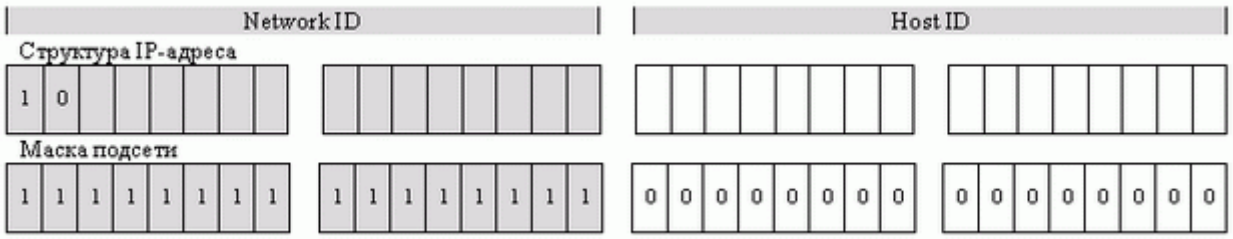


Рис. 4.3.
IP-адреса класса C.

Три старших разряда любого IP-адреса в сети класса C всегда равны `110`. Идентификатор сети состоит из 24 разрядов, идентификатор узла — из 8 разрядов. Маска подсети для узлов сетей класса C — `255.255.255.0`. Структура IP-адресов класса C приведена на рис. 4.4.

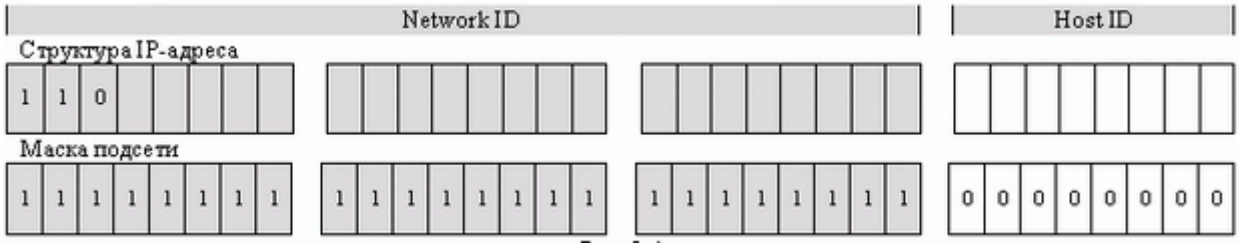


Рис. 4.4.
Класс D

IP-адреса класса D используются для групповых адресов (multicast-адреса). Четыре старших разряда любого IP-адреса в сети класса D всегда равны `1110`. Оставшиеся 28 бит используются для назначения группового адреса.

Класс E

Пять старших разрядов любого IP-адреса в сети класса E равны `11110`. Адреса данного класса зарезервированы для будущего использования (и не поддерживаются системой Windows Server).

Правила назначения идентификаторов сети (Network ID)

- первый октет идентификатора сети не может быть равен `127` (адреса вида `127.x.y.z` предназначены для отправки узлом пакетов самому себе и используются как правило для отладки сетевых приложений, такие адреса называются loopback-адресами, или адресами обратной связи);
- все разряды идентификатора сети не могут состоять из одних `1` (IP-адреса, все биты идентификаторов сети которых установлены в 1, используются при широковещательной передаче информации);
- все разряды идентификатора сети не могут состоять из одних `0` (в IP-адресах все биты, установленные в ноль, соответствуют либо данному устройству, либо данной сети);
- идентификатор каждой конкретной сети должен быть уникальным среди подсетей, объединенных в одну сеть с помощью маршрутизаторов.

Диапазоны возможных идентификаторов сети приведены в табл. 4.1.

Таблица 4.1.

Класс сети	Наименьший идентификатор сети	Наибольший идентификатор сети	Количество сетей

Класс A	1.0.0.0	126.0.0.0	126
Класс B	128.0.0.0	191.255.0.0	16384
Класс C	192.0.0.0	223.255.255.0	2097152

Правила назначения идентификаторов узла (Host ID)

- все разряды идентификатора узла не могут состоять из одних 1 (идентификатор узла, состоящий из одних 1, используется для широковещательных адресов, или broadcast-адресов);
- все разряды идентификатора узла не могут состоять из одних 0 (если разряды идентификатора узла равны 0, то такой адрес обозначает всю подсеть, например, адрес 192.168.1.0 с маской подсети 255.255.255.0 обозначает всю подсеть с идентификатором сети 192.168.1 ;
- идентификатор узла должен быть уникальным среди узлов одной подсети.

Диапазоны возможных идентификаторов узла приведены в табл. 4.2.

Таблица 4.2.

Класс сети	Наименьший идентификатор узла	Наибольший идентификатор узла	Количество узлов
Класс A	w.0.0.1	w.255.255.254	16777214
Класс B	w.x.0.1	w.x.255.254	65534
Класс C	w.x.y.1	w.x.y.254	254

Другим способом обозначения сети, более удобным и более кратким, является обозначение сети с сетевым префиксом. Такое обозначение имеет вид "/число бит маски подсети". Например, подсеть 192.168.1.0 с маской подсети 255.255.255.0 можно более кратко записать в виде 192.168.1.0/24, где число 24 длина маски подсети в битах.

Публичные и приватные (частные) IP-адреса

Все пространство IP-адресов разделено на 2 части: публичные адреса, которые распределяются между Интернет-провайдерами и компаниями международной организацией Internet Assigned Numbers Authority (сокращенно IANA), и приватные адреса, которые не контролируются IANA и могут назначаться внутрикорпоративным узлам по усмотрению сетевых администраторов. Если какая-либо компания приобрела IP-адреса в публичной сети, то ее сетевые узлы могут напрямую маршрутизировать сетевой трафик в сеть Интернет и могут быть прозрачно доступны из Интернета. Если внутрикорпоративные узлы имеют адреса из приватной сети, то они могут получать доступ в Интернет с помощью протокола трансляции сетевых адресов (NAT, Network Address Translation) или с помощью прокси-сервера. В простейшем случае с помощью NAT возможно организовать работу всей компании с использованием единственного зарегистрированного IP-адреса.

Механизм трансляции адресов NAT преобразует IP-адреса из частного адресного пространства IP (эти адреса еще называют "внутренние", или "серые IP") в зарегистрированное открытое адресное пространство IP. Обычно эти функции (NAT) выполняет либо маршрутизатор, либо межсетевой экран (firewall) — эти устройства подменяют адреса в заголовках проходящих через них IP-пакетов.

На практике обычно компании получают через Интернет-провайдеров небольшие сети в пространстве публичных адресов для размещения своих внешних ресурсов — web-сайтов или почтовых серверов. А для внутрикорпоративных узлов используют приватные IP-сети.

Пространство приватных IP-адресов состоит из трех блоков:

- 10.0.0.0/8 (одна сеть класса A);
- 172.16.0.0/12 (диапазон адресов, состоящий из 16 сетей класса B — от 172.16.0.0/16 до 172.31.0.0/16);
- 192.168.0.0/16 (диапазон адресов, состоящий из 256 сетей класса C — от 192.168.0.0/24 до 192.168.255.0/16).

Кроме данных трех блоков имеется еще блок адресов, используемых для автоматической IP-адресации (APIPA, Automatic *Private IP Addressing*). Автоматическая IP-адресация применяется в том случае, когда сетевой интерфейс настраивается для автоматической настройки IP-конфигурации, но при этом в сети отсутствует сервер DHCP. Диапазон адресов для APIPA — сеть класса B169.254.0.0/16.

Отображение IP-адресов на физические адреса

Каждый сетевой адаптер имеет свой уникальный физический адрес (или MAC-адрес). За отображение IP-адресов адаптеров на их физические адреса отвечает протокол ARP (Address Resolution Protocol). Необходимость протокола ARP продиктована тем обстоятельством, что IP-адреса устройств в сети назначаются независимо от их физических адресов. Поэтому для доставки сообщений по сети необходимо определить соответствие между физическим адресом устройства и его IP-адресом — это называется разрешением адресов. В большинстве случаев прикладные программы используют именно IP-адреса. А так как схемы физической адресации устройств весьма разнообразны, то необходим специальный, универсальный протокол. *Протокол разрешения адресов*ARP был разработан таким образом, чтобы его можно было использовать для разрешения адресов в различных сетях. Фактически ARP можно использовать с произвольными физическими адресами и сетевыми протоколами. Протокол ARP предполагает, что каждое устройство знает как свой IP-адрес, так и свой физический адрес. ARP динамически связывает их и заносит в специальную таблицу, где хранятся пары "IP-адрес — физический адрес" (обычно каждая запись в ARP-таблице имеет время жизни 10 мин.). Эта таблица хранится в памяти компьютера и называется кэш протокола ARP (ARP-cache).

Работа протокола ARP заключается в отправке сообщений между сетевыми узлами:

- ARP Request (запрос ARP) — широковещательный запрос, отправляемый на физическом уровне модели TCP/IP, для определения MAC-адреса узла, имеющего конкретный IP-адрес;
- ARP Reply (ответ ARP) — узел, IP-адрес которого содержится в ARP-запросе, отправляет узлу, пославшему ARP-запрос, информацию о своем MAC-адресе;
- RARP Request, или Reverse ARP Request (обратный ARP-запрос) — запрос на определение IP-адреса по известному MAC-адресу;
- RARP Reply, или Reverse ARP Reply (обратный ARP-ответ) — ответ узла на обратный ARP-запрос.

Разбиение сетей на подсети с помощью маски подсети

Для более эффективного использования пространства адресов IP-сети с помощью маски подсети могут быть разбиты на более мелкие подсети (subnetting) или объединены в более крупные сети (supernetting).

Рассмотрим на примере разбиение сети 192.168.1.0/24 (сеть класса C) на более мелкие подсети. В исходной сети в IP-адресе 24 бита относятся к идентификатору сети и 8 бит — к идентификатору узла. Используем маску подсети из 27 бит, или, в десятичном обозначении, — 255.255.255.224, в двоичном обозначении — 11111111 11111111 11111111 11100000. Получим следующее разбиение на подсети:

Таблица 4.3.

Подсеть	Диапазон IP-адресов	Широковещательный адрес в подсети
192.168.1.0/27	192.168.1.1–192.168.1.30	192.168.1.31
192.168.1.32/27	192.168.1.33–192.168.1.62	192.168.1.63
192.168.1.64/27	192.168.1.65–192.168.1.94	192.168.1.95
192.168.1.96/27	192.168.1.97–192.168.1.126	192.168.1.127
192.168.1.128/27	192.168.1.129–192.168.1.158	192.168.1.159
192.168.1.160/27	192.168.1.161–192.168.1.190	192.168.1.191
192.168.1.192/27	192.168.1.193–192.168.1.222	192.168.1.223
192.168.1.224/27	192.168.1.225–192.168.1.254	192.168.1.255

Таким образом, мы получили 8 подсетей, в каждой из которых может быть до 30 узлов. Напомним, что идентификатор узла, состоящий из нулей, обозначает всю подсеть, а идентификатор узла, состоящий из одних единиц, означает широковещательный адрес (пакет, отправленный на такой адрес, будет доставлен всем узлам подсети).

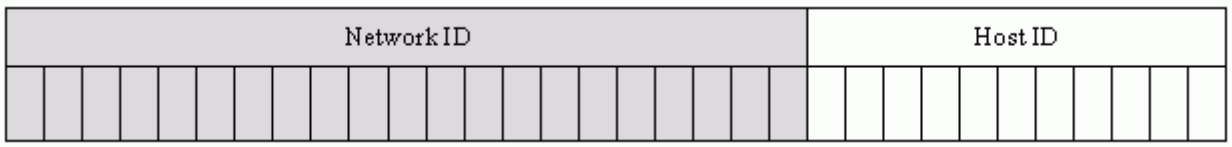
IP-адреса в данных подсетях будут иметь структуру:

Отметим очень важный момент. С использованием такой маски узлы с такими, например, IP-адресами, как 192.168.1.48 и 192.168.1.72, находятся в различных подсетях, и для взаимодействия данных узлов необходимы маршрутизаторы, пересылающие пакеты между подсетями 192.168.1.32/27 и 192.168.1.64/27.

Примечание. Согласно стандартам протокола TCP/IP для данного примера не должно существовать подсетей 192.168.1.0/27 и 192.168.1.224/27 (т.е. первая и последняя подсети). На практике большинство операционных систем (в т.ч. системы семейства Microsoft Windows) и маршрутизаторов поддерживают работу с такими сетями.

Аналогично, можно с помощью маски подсети объединить мелкие сети в более крупные.

Например, IP-адреса сети 192.168.0.0/21 будут иметь следующую структуру:



Диапазон IP-адресов данной сети: 192.168.0.1–192.168.7.254 (всего — 2046 узлов), широковещательный адрес подсети — 192.168.7.255.

Преимущества подсетей внутри частной сети:

- разбиение больших IP-сетей на подсети (subnetting) позволяет снизить объем широковещательного трафика (маршрутизаторы не пропускают широковещательные пакеты);
- объединение небольших сетей в более крупные сети (supernetting) позволяет увеличить адресное пространство с помощью сетей более низкого класса;
- изменение топологии частной сети не влияет на таблицы маршрутизации в сети Интернет (хранят только маршрут с общим номером сети);
- размер глобальных таблиц маршрутизации в сети Интернет не растет;

- администратор может создавать новые подсети без необходимости получения новых номеров сетей.

Старшие биты IP-адреса используются рабочими станциями и маршрутизаторами для определения класса адреса. После того как класс определен, устройство может однозначно вычислить границу между битами, использующимися для идентификации номера сети, и битами номера устройства в этой сети. Однако при разбиении сетей на подсети или при объединении сетей для определения границ битов, идентифицирующих номер подсети, такая схема не подходит. Для этого как раз и используется 32-битная маска подсети, которая помогает однозначно определить требуемую границу. Напомним, что для стандартных *классов сетей* маски имеют следующие значения:

- `255.0.0.0` – маска для сети класса A;
- `255.255.0.0` - маска для сети класса B;
- `255.255.255.0` - маска для сети класса C.

Для администратора сети чрезвычайно важно знать четкие ответы на следующие вопросы:

- Сколько подсетей требуется организации сегодня?
- Сколько подсетей может потребоваться организации в будущем?
- Сколько устройств в наибольшей подсети организации сегодня?
- Сколько устройств будет в самой большой подсети организации в будущем?

Отказ от использования только стандартных классов IP-сетей (A, B, и C) называется бесклассовой междоменной маршрутизацией (Classless Inter-Domain Routing, *CIDR*).

Введение в IP-маршрутизацию

Для начала уточним некоторые понятия:

- сетевой узел (node) — любое сетевое устройство с протоколом TCP/IP;
- хост (host) — сетевой узел, не обладающий возможностями маршрутизации пакетов;
- маршрутизатор (router) — сетевой узел, обладающий возможностями маршрутизации пакетов

IP-маршрутизация — это процесс пересылки *unicast*-трафика от узла-отправителя к узлу –получателю в IP-сети с произвольной топологией.

Когда один узел IP-сети отправляет пакет другому узлу, в заголовке IP-пакета указываются IP-адрес узла отправителя и IP-адрес узла-получателя. Отправка пакета происходит следующим образом:

1. Узел-отправитель определяет, находится ли узел-получатель в той же самой IP-сети, что и отправитель (в локальной сети), или в другой IP-сети (в удаленной сети). Для этого узел-отправитель производит поразрядное логическое умножение своего IP-адреса на маску подсети, затем поразрядное логическое умножение IP-адреса узла получателя также на свою маску подсети. Если результаты совпадают, значит, оба узла находятся в одной подсети. Если результаты различны, то узлы находятся в разных подсетях.
2. Если оба сетевых узла расположены в одной IP-сети, то узел-отправитель сначала проверяет ARP-кэш на наличие в ARP-таблице MAC-адреса узла-получателя. Если нужная запись в таблице имеется, то дальше отправка пакетов производится напрямую узлу-получателю на канальном уровне. Если же в ARP-таблице нужной записи нет, то узел-отправитель посылает ARP-запрос для IP-адреса узла-получателя, ответ помещает в ARP-таблицу и после этого передача

пакета также производится на канальном уровне (между сетевыми адаптерами компьютеров).

3. Если узел-отправитель и узел-получатель расположены в разных IP-сетях, то узел-отправитель посылает данный пакет сетевому узлу, который в конфигурации отправителя указан как "Основной шлюз" (*default gateway*). Основной шлюз всегда находится в той же IP-сети, что и узел-отправитель, поэтому взаимодействие происходит на канальном уровне (после выполнения ARP-запроса). Основной шлюз — это маршрутизатор, который отвечает за отправку пакетов в другие подсети (либо напрямую, либо через другие маршрутизаторы).

Рассмотрим пример, изображенный на рис. 4.5.

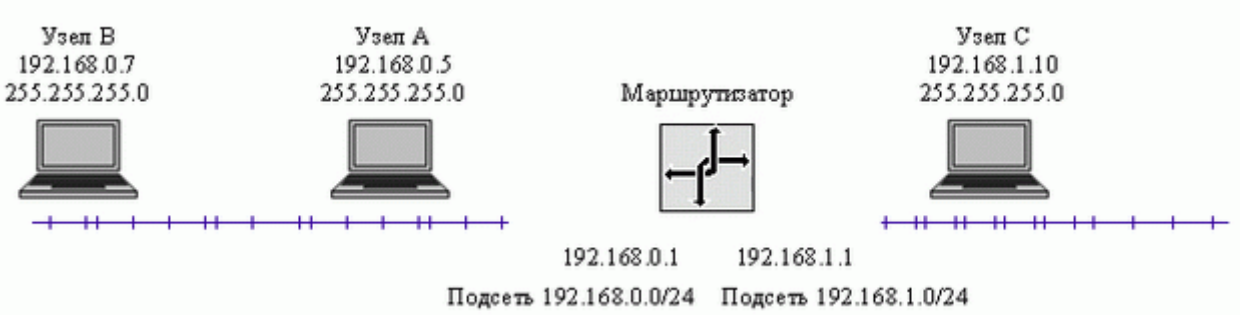


Рис. 4.5.

В данном примере 2 подсети: `192.168.0.0/24` и `192.168.1.0/24`. Подсети объединены в одну сеть маршрутизатором. Интерфейс маршрутизатора в первой подсети имеет IP-адрес `192.168.0.1`, во второй подсети - `192.168.1.1`. В первой подсети имеются 2 узла: узел A (`192.168.0.5`) и узел B (`192.168.0.7`). Во второй подсети имеется узел C с IP-адресом `192.168.1.10`.

Если узел A будет отправлять пакет узлу B, то сначала он вычислит, что узел B находится в той же подсети, что и узел A (т.е. в локальной подсети), затем узел A выполнит ARP-запрос для IP-адреса `192.168.0.7`. После этого содержимое IP-пакета будет передано на канальный уровень, и информация будет передана сетевым адаптером узла A сетевому адаптеру узла B. Это пример прямой доставки данных (или прямой маршрутизации, direct delivery).

Если узел A будет отправлять пакет узлу C, то сначала он вычислит, что узел C находится в другой подсети (т.е. в удаленной подсети). После этого узел A отправит пакет узлу, который в его конфигурации указан в качестве основного шлюза (в данном случае это интерфейс маршрутизатора с IP-адресом `192.168.0.1`). Затем маршрутизатор с интерфейса `192.168.1.1` выполнит прямую доставку узлу C. Это пример непрямой доставки (или косвенной маршрутизации, indirect delivery) пакета от узла A узлу C. В данном случае процесс косвенной маршрутизации состоит из двух операций прямой маршрутизации.

В целом процесс IP-маршрутизации представляет собой серии отдельных операций прямой или косвенной маршрутизации пакетов.

Каждый сетевой узел принимает решение о маршрутизации пакета на основе таблицы маршрутизации, которая хранится в оперативной памяти данного узла. Таблицы маршрутизации существуют не только у маршрутизаторов с несколькими интерфейсами, но и у рабочих станций, подключаемых к сети через сетевой адаптер. Таблицу маршрутизации в системе Windows можно посмотреть по команде `route print`. Каждая таблица маршрутизации содержит набор записей. Записи могут формироваться различными способами:

- записи, созданные автоматически системой на основе конфигурации протокола TCP/IP на каждом из сетевых адаптеров;
- статические записи, созданные командой `route add` или в консоли службы *Routing and Remote Access Service* ;
- динамические записи, созданные различными протоколами маршрутизации (RIP или OSPF).

Рассмотрим два примера: таблицу маршрутизации типичной рабочей станции, расположенной в локальной сети компании, и таблицу маршрутизации сервера, имеющего несколько сетевых интерфейсов.

Рабочая станция.

В данном примере имеется рабочая станция с системой Windows XP, с одним сетевым адаптером и такими настройками протокола TCP/IP: IP-адрес — `192.168.1.10`, маска подсети —`255.255.255.0`, основной шлюз — `192.168.1.1`.

Введем в командной строке системы Windows команду `route print`, результатом работы команды будет следующий экран (рис. 4.6; в скобках приведен текст для английской версии системы):

```
C:\>route print

IPv4 таблица маршрута
===========================================================================
Список интерфейсов (Interface List)
0x1 ...........................          MS TCP Loopback interface
0x10002 ...00 c0 26 a1 6e 05 ......      Realtek   RTL8139   Family   PCI   Fast
Ethernet NIC
===========================================================================
===========================================================================
Активные маршруты (Active Routes):

Сетевой адрес     Маска сети       Адрес шлюза     Интерфейс       Метрика
(Network          (Netmask)        (Gateway)       (Interface)     (Metric)
Destination)
        0.0.0.0          0.0.0.0     192.168.1.1    192.168.1.10              1
      127.0.0.0        255.0.0.0       127.0.0.1       127.0.0.1              1
    192.168.1.0    255.255.255.0    192.168.1.10    192.168.1.10             20
   192.168.1.10  255.255.255.255       127.0.0.1       127.0.0.1             20
  192.168.1.255  255.255.255.255    192.168.1.10    192.168.1.10             20
      224.0.0.0        240.0.0.0    192.168.1.10    192.168.1.10             20
255.255.255.255  255.255.255.255    192.168.1.10    192.168.1.10              1
Основной шлюз (Default Gateway):      192.168.1.1
===========================================================================
Постоянные маршруты (Persistent Routes):
  Отсутствует (None)
```

Рис. 4.6.
Список интерфейсов — список сетевых адаптеров, установленных в компьютере. Интерфейс *MS TCP Loopback interface* присутствует всегда и предназначен для обращения узла к самому себе. Интерфейс *Realtek RTL8139 Family PCI Fast Ethernet NIC* — сетевая карта.

Далее идет сама таблица маршрутов. Каждая строка таблицы — это маршрут для какой-либо IP-сети. Ее столбцы:

Сетевой адрес — диапазон IP-адресов, которые достижимы с помощью данного маршрута.

Маска сети — маска подсети, в которую отправляется пакет с помощью данного маршрута.

Адрес шлюза — IP-адрес узла, на который пересылаются пакеты, соответствующие данному маршруту.

Интерфейс — обозначение сетевого интерфейса данного компьютера, на который пересылаются пакеты, соответствующие маршруту.

Метрика — условная стоимость маршрута. Если для одной и той же сети есть несколько маршрутов, то выбирается маршрут с минимальной стоимостью. Как правило, метрика — это количество маршрутизаторов, которые должен пройти пакет, чтобы попасть в нужную сеть.

Проанализируем некоторые строки таблицы.

Первая строка таблицы соответствует значению основного шлюза в конфигурации TCP/IP данной станции. Сеть с адресом "0.0.0.0" обозначает "все остальные сети, не соответствующие другим строкам данной таблицы маршрутизации".

Вторая строка — маршрут для отправки пакетов от узла самому себе.

Третья строка (сеть `192.168.1.0` с маской `255.255.255.0`) — маршрут для отправки пакетов в локальной IP-сети (т.е. той сети, в которой расположена данная рабочая станция).

Последняя строка — широковещательный адрес для всех узлов локальной IP-сети.

Последняя строка на рис. 4.6 — список постоянных маршрутов рабочей станции. Это статические маршруты, которые созданы командой `route add`. В данном примере нет ни одного такого статического маршрута.

Сервер.

Теперь рассмотрим сервер с системой Windows 2003 Server, с тремя сетевыми адаптерами:

- Адаптер 1 — расположен во внутренней сети компании (IP-адрес — `192.168.1.10`, маска подсети — `255.255.255.0`);
- Адаптер 2 — расположен во внешней сети Интернет-провайдера ISP-1 (IP-адрес — `213.10.11.2`, маска подсети — `255.255.255.248`, ближайший интерфейс в сети провайдера — `213.10.11.1`);
- Адаптер 3 — расположен во внешней сети Интернет-провайдера ISP-2 (IP-адрес — `217.1.1.34`, маска подсети — `255.255.255.248`, ближайший интерфейс в сети провайдера — `217.1.1.33`).

IP-сети провайдеров — условные, IP-адреса выбраны лишь для иллюстрации (хотя вполне возможно случайное совпадение с какой-либо существующей сетью).

Кроме того, на сервере установлена Служба маршрутизации и удаленного доступа для *управления маршрутизацией* пакетов между IP-сетями и доступа в сеть компании через модемный пул.

В данном случае команда `route print` выдаст таблицу маршрутизации, изображенную на рис. 4.7.

```
C:\>route print

IPv4 таблица маршрута
===========================================================================
Список интерфейсов (Interface List)
0x1 ...........................          MS TCP Loopback interface
0x10002 ...00 53 45 00 00 00 ...... WAN (PPP/SLIP) Interface
0x10003 ...00 03 47 97 61 81 ...... Intel(R) 10/100 Network Adapter
0x10004 ...00 02 b3 a6 be 48 ...... Intel(R) PRO/100 Adapter
0x10005 ...00 d0 b7 b7 fd df ...... Intel 8255x-based PCI Ethernet Adapter
(10/100)
===========================================================================
===========================================================================
Активные маршруты (Active Routes):

Сетевой адрес     Маска сети       Адрес шлюза     Интерфейс       Метрика
(Network          (Netmask)        (Gateway)       (Interface)     (Metric)
Destination)
        0.0.0.0          0.0.0.0     213.10.11.1     213.10.11.2              20
    196.15.20.16    255.255.255.0      217.1.1.33      217.1.1.33               1
       127.0.0.0        255.0.0.0       127.0.0.1       127.0.0.1               1
     192.168.1.0    255.255.255.0     192.168.1.1     192.168.1.1              20
     192.168.1.1  255.255.255.255       127.0.0.1       127.0.0.1              20
   192.168.1.255  255.255.255.255     192.168.1.1     192.168.1.1              20
    192.168.10.1  255.255.255.255       127.0.0.1       127.0.0.1              50
     213.10.11.0  255.255.255.248     213.10.11.2     213.10.11.2              20
     213.10.11.2  255.255.255.255       127.0.0.1       127.0.0.1              20
   213.10.11.255  255.255.255.255     213.10.11.2     213.10.11.2              20
      217.1.1.32  255.255.255.248      217.1.1.33      217.1.1.33              20
      217.1.1.34  255.255.255.255       127.0.0.1       127.0.0.1              20
     217.1.1.255  255.255.255.255      217.1.1.34      217.1.1.34              20
       224.0.0.0        240.0.0.0     192.168.1.1     192.168.1.1              20
       224.0.0.0        240.0.0.0     213.10.11.2     213.10.11.2              20
       224.0.0.0        240.0.0.0      217.1.1.34      217.1.1.34              20
 255.255.255.255  255.255.255.255     192.168.1.1     192.168.1.1               1
 255.255.255.255  255.255.255.255     213.10.11.2     213.10.11.2               1
 255.255.255.255  255.255.255.255      217.1.1.34      217.1.1.34               1
Основной шлюз (Default Gateway):      213.10.11.1
===========================================================================
Постоянные маршруты (Persistent Routes):
  Отсутствует (None)
```

Рис. 4.7.
В таблице в списке интерфейсов отображены три сетевых адаптера разных моделей, адаптер обратной связи (MS TCP *Loopback interface*) и *WAN (PPP/SLIP) Interface* — интерфейс для доступа в сеть через модемный пул.

Отметим особенности таблицы маршрутов сервера с несколькими сетевыми интерфейсами.

Первая строка похожа на первую строку в таблице рабочей станции. Она также соответствует значению основного шлюза в конфигурации TCP/IP данной станции. Заметим, что только на одном интерфейсе можно задавать параметр "Основной шлюз". В данном случае этот параметр был задан на одном из внешних интерфейсов (это же значение отражено и в конце таблицы в строке "Основной шлюз").

Как и в рабочей станции, для каждого интерфейса есть маршруты как для *unicast*-пакетов, так и для широковещательных (broadcast) для каждой подсети.

Во второй строке содержится статический маршрут, сконфигурированный в консоли *Службы маршрутизации и удаленного доступа*, для пересылки пакетов в сеть `196.15.20.16/24`.

Поддержка таблиц маршрутизации.

Есть два способа поддержки актуального состояния таблиц маршрутизации: ручной и автоматический.

Ручной способ подходит для небольших сетей. В этом случае в таблицы маршрутизации вручную заносятся статические записи для маршрутов. Записи создаются либо командой `route add`, либо в консоли *Службы маршрутизации и удаленного доступа*.

В больших сетях ручной способ становится слишком трудоемким и чреват ошибками. Автоматическое построение и модификация таблиц маршрутизации производится так называемыми*"динамическими маршрутизаторами"*. Динамические маршрутизаторы отслеживают изменения в топологии сети, вносят необходимые изменения в таблицы маршрутов и обмениваются данной информацией с другими маршрутизаторами, работающими по тем же протоколам маршрутизации. В Windows Server реализована динамическая маршрутизация в *Службе маршрутизации и удаленного доступа*. В данной службе реализованы наиболее распространенные протоколы маршрутизации — протокол RIP версий 1 и 2 и протокол OSPF.

4.2 Служба DNS (домены, зоны; зоны прямого и обратного просмотра; основные и дополнительные зоны; рекурсивный и итеративный запросы на разрешение имен).

Историческая справка: Систему доменных имен разработал в 1983 году Пол Мокапетрис. Тогда же было проведено первое успешное тестирование DNS, ставшей позже одним из базовых компонентов сети Internet. С помощью DNS стало возможным реализовать масштабируемый распределенный механизм, устанавливающий соответствие между иерархическими именами сайтов и числовыми IP-адресами.

В 1983 году Пол Мокапетрис работал научным сотрудником института информатики (Information Sciences Institute, *ISI*), входящего в состав инженерной школы университета Южной Калифорнии (*USC*). Его руководитель, Джон Постел, предложил Полу придумать новый механизм, устанавливающий связи между именами компьютеров и адресами Internet, - взамен использовавшемуся тогда централизованному каталогу имен и адресов хостов, который поддерживала калифорнийская компания SRI International.

"Все понимали, что старая схема не сможет работать вечно, - вспоминает Мокапетрис. - Рост Internet становился лавинообразным. К сети, возникшей на основе проекта ARPANET, инициированного Пентагоном, присоединялись все новые и новые компании и исследовательские институты".

Предложенное Мокапетрисом решение - DNS - представляло собой распределенную базу данных, которая позволяла организациям, присоединившимся к Internet, получить свой домен.

"Как только организация подключалась к сети, она могла использовать сколь угодно много компьютеров и сама назначать им имена", - подчеркнул Мокапетрис. Названия доменов компаний получили суффикс `.com`, университетов - `.edu` и так далее.

Первоначально DNS была рассчитана на поддержку 50 млн. записей и допускала безопасное расширение до нескольких сотен миллионов записей. По оценкам Мокапетриса, сейчас насчитывается около 1 млрд. имен DNS, в том числе почти 20 млн. общедоступных имен. Остальные принадлежат системам, расположенным за межсетевыми экранами. Их имена неизвестны обычным Internet-пользователям.

Новая система внедрялась постепенно, в течение нескольких лет. В это время ряд исследователей экспериментировали с ее возможностями, а Мокапетрис занимался в *ISI* обслуживанием и поддержанием стабильной работы "корневого сервера", построенного на мэйнфреймах компании Digital Equipment. Копии таблиц хостов хранились на каждом компьютере, подключенном к Internet, еще примерно до 1986 года. Затем начался массовый переход на использование DNS.

Необходимость отображения имен сетевых узлов в IP-адреса

Компьютеры и другие сетевые устройства, отправляя друг другу пакеты по сети, используют IP-адреса. Однако пользователю (человеку) гораздо проще и удобнее запомнить некоторое символические имена сетевых узлов, чем четыре бессодержательных для него числа. Однако, если люди в своих операциях с сетевыми ресурсами будут использовать имена узлов, а не IP-адреса, тогда должен существовать механизм, сопоставляющий именам узлов их IP-адреса.

Есть два таких механизма - локальный для каждого компьютера файл hosts и централизованная иерархическая служба имен DNS.

Использование локального файла hosts и системы доменных имен DNS для разрешения имен сетевых узлов

На начальном этапе развития сетей, когда количество узлов в каждой сети было небольшое, достаточно было на каждом компьютере хранить и поддерживать актуальное состояние простого текстового файла, в котором содержался список сетевых узлов данной сети. Список устроен очень просто - в каждой строке текстового файла содержится пара "IP-адрес - имя сетевого узла". В системах семейства Windows данный файл расположен в папке `%system root%\system32\drivers\etc` (где `%system root%` обозначает папку, в которой установлена операционная система). Сразу после установки системы Windows создается файл `hosts` с одной записью `127.0.0.1 localhost`.

С ростом сетей поддерживать актуальность и точность информации в файле `hosts` становится все труднее. Для этого надо постоянно обновлять содержимое этого файла на всех узлах сети. Кроме того, такая простая технология не позволяет организовать пространство имен в какую-либо структуру. Поэтому появилась необходимость в централизованной базе данных имен, позволяющей производить преобразование имен в IP-адреса без хранения списка соответствия на каждом компьютере. Такой базой стала DNS (Domain Name System) - система именования доменов, которая начала массовую работу в 1987 году.

Заметим, что с появлением службы DNS актуальность использования файла host совсем не исчезла, в ряде случаев использование этого файла оказывается очень эффективным.

Служба DNS: пространство имен, домены

DNS - это *иерархическая база данных*, сопоставляющая имена сетевых узлов и их сетевых служб IP-адресам узлов. Содержимое этой базы, с одной стороны, распределено по большому количеству серверов службы DNS, а с другой стороны, является централизованно управляемым. В основе *иерархической структуры базы данных* DNS лежит доменное пространство имен (domain namespace), основной структурной единицей которого является домен, объединяющий сетевые узлы (хосты), а также поддомены. Процесс поиска в БД службы DNS имени некоего сетевого узла и сопоставления этому имени IP-адреса называется "разрешением имени узла в пространстве имен DNS".

Служба DNS состоит из трех основных компонент:

- **Пространство имен DNS и соответствующие ресурсные записи (RR, resource record)** - это сама распределенная база данных DNS;
- **Серверы имен DNS** - компьютеры, хранящие базу данных DNS и отвечающие на запросы DNS-клиентов;
- **DNS-клиенты (DNS-clients, DNS-resolvers)** -компьютеры, посылающие запросы серверам DNS для получения ресурсных записей.

Пространство имен.

Пространство имен DNS - иерархическая древовидная структура, начинающаяся с корня, не имеющего имени и обозначаемого точкой ".". Схему построения пространства имен DNS лучше всего проиллюстрировать на примере сети Интернет (рис. 4.8).

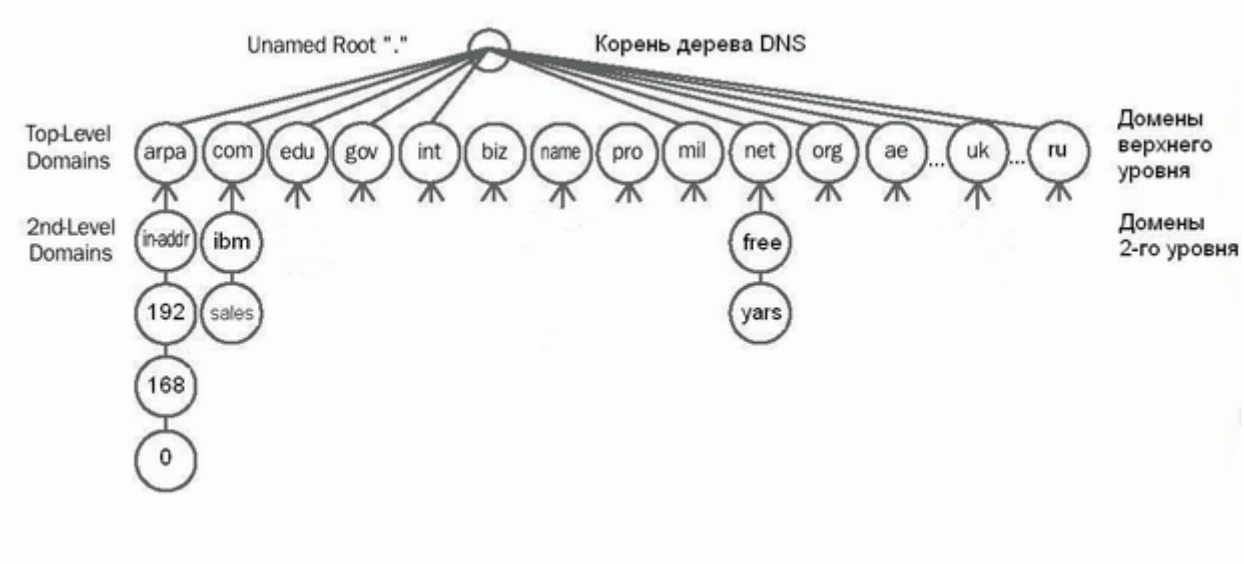


Рис. 4.8.
Для доменов 1-го уровня различают 3 категории имен:

- **ARPA** - специальное имя, используемое для обратного разрешения DNS (из IP-адреса в полное имя узла);
- **Общие (generic) имена 1-го уровня** - 16 (на данный момент) имен, назначение которых приведено в табл. 4.4;
- **Двухбуквенные имена для стран** - имена для доменов, зарегистрированных в соответствующих странах (например, `ru` - для России, `ua` - для Украины, `uk` - для Великобритании и т.д.).

Таблица 4.4.

Имя домена	Назначение
`aero`	Сообщества авиаторов
`biz`	Компании (без привязки к стране)
`com`	Коммерческие организации, преимущественно в США (например, домен microsoft.com для корпорации Microsoft)
`coop`	Кооперативы
`edu`	Образовательные учреждения в США
`gov`	Правительственные учреждения США
`info`	Домен для организаций, предоставляющих любую информацию для потребителей
`int`	международные организации (например, домен *nato*.int для НАТО)

mil	Военные ведомства США
museum	Музеи
name	Глобальный домен для частных лиц
net	Домен для Интернет-провайдеров и других организаций, управляющих структурой сети Интернет
org	Некоммерческие и неправительственные организации, преимущественно в США
pro	Домен для профессиональных объединений (врачей, юристов, бухгалтеров и др.)
job	Кадровые агентства
travel	Туроператоры

Для непосредственного отображения пространства имен в пространство IP-адресов служат т.н. ресурсные записи (RR, resource record). Каждый сервер DNS содержит ресурсные записи для той части пространства имен, за которую он несет ответственность (*authoritative*). табл. 4.5 содержит описание наиболее часто используемых типов ресурсных записей.

Таблица 4.5.

Тип ресурсной записи	Функция записи	Описание использования
A	*Host Address* Адрес хоста, или узла	Отображает имя узла на IP-адрес (например, для домена microsoft.com узлу с именем www.microsoft.com сопоставляется IP-адрес с помощью такой записи: `www A 207.46.199.60`)
CNAME	Canonical Name (alias) Каноническое имя (псевдоним)	Отображает одно имя на другое
MX	Mail Exchanger Обмен почтой	Управляет маршрутизацией почтовых сообщений для протокола SMTP
NS	Name Server Сервер имен	Указывает на серверы DNS, ответственные за конкретный домен и его поддомены
PTR	Pointer Указатель	Используется для обратного разрешения IP-адресов в имена узлов в домене in-addr.arpa
SOA	Start of Authority Начальная запись зоны	Используется для указания основного сервера для данной зоны и описания свойств зоны
SRV	Service Locator Указатель на службу	Используется для поиска серверов, на которых функционируют определенные службы (например, контроллеры доменов Active Directory или *серверы глобального каталога*)

Полное имя узла (FQDN, fully qualified domain name) состоит из нескольких имен, называемых метками (label) и разделенных точкой. Самая левая метка относится непосредственно к узлу, остальные метки - список доменов от домена первого уровня до того домена, в котором находится узел (данный список просматривается справа налево).

Серверы имен DNS.

Серверы имен DNS (или DNS-серверы) - это компьютеры, на которых хранятся те части БД пространства имен DNS, за которые данные серверы отвечают, и функционирует программное обеспечение, которое обрабатывает запросы DNS-клиентов на разрешение имен и выдает ответы на полученные запросы.

DNS-клиенты.

DNS-клиент - это любой сетевой узел, который обратился к DNS-серверу для разрешения имени узла в IP-адрес или, обратно, IP-адреса в имя узла.

Служба DNS: домены и зоны

Как уже говорилось выше, каждый DNS-сервер отвечает за обслуживание определенной части пространства имен DNS. Информация о доменах, хранящаяся в БД сервера DNS, организуется в особые единицы, называемые зонами (zones). Зона - основная единица репликации данных между серверами DNS. Каждая зона содержит определенное количество ресурсных записей для соответствующего домена и, быть может, его поддоменов.

Системы семейства Windows Server поддерживают следующие типы зон:

- **Стандартная основная (standard primary)** - главная копия стандартной зоны; только в данном экземпляре зоны допускается производить какие-либо изменения, которые затем реплицируются на серверы, хранящие дополнительные зоны;
- **Стандартная дополнительная (standard secondary)** - копия основной зоны, доступная в режиме "только-чтение", предназначена для повышения отказоустойчивости и распределения нагрузки между серверами, отвечающими за определенную зону; процесс репликации изменений в записях зон называется "передачей зоны" (*zone transfer*) (информация в стандартных зонах хранится в текстовых файлах, файлы создаются в папке "%system root%\system32\dns", имя файла, как правило, образуется из имени зоны с добавлением расширения файла ".dns"; термин "стандартная" используется только в системах семейства Windows);
- **Интегрированная в Active Directory (Active Directory–integrated)** - вся информация о зоне хранится в виде одной записи в базе данных Active Directory (такие типы зон могут существовать только на серверах Windows, являющихся контроллерами доменов Active Directory; в интегрированных зонах можно более жестко управлять правами доступа к записям зоны; изменения в записях зоны между разными экземплярами интегрированной зоны производятся не по технологии передачи зоны службой DNS, а механизмами репликации службы Active Directory);
- **Зона-заглушка** (**stub** ; только в Windows 2003) - особый тип зоны, которая для данной части пространства имен DNS содержит самый минимальный набор ресурсных записей (начальная запись зоны SOA, список серверов имен, отвечающих за данную зону, и несколько записей типа A для ссылок на серверы имен для данной зоны).

Рассмотрим на примере соотношение между понятиями домена и зоны. Проанализируем информацию, представленную на рис. 4.9.

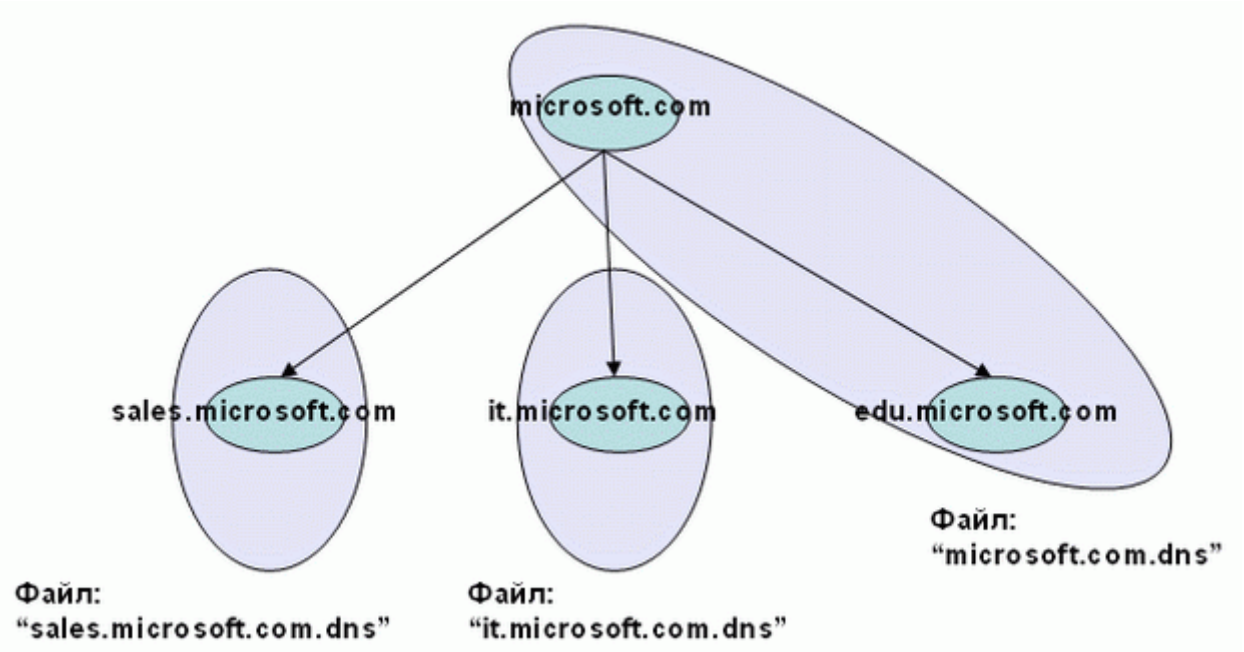


Рис. 4.9.
В данном примере пространство имен DNS начинается с домена microsoft.com, который содержит 3
поддомена: sales.microsoft.com, it.microsoft.com и edu.microsoft.com (домены на рисунке обозначены маленькими горизонтальными овалами). Домен - понятие чисто логическое, относящееся только к распределению имен. Понятие домена никак не связано с технологией хранения информации о *домене. Зона* - это способ представления информации о домене и его поддоменах в хранилище тех серверов DNS, которые отвечают за данный домен и поддомены. В данной ситуации, если для хранения выбрана технология стандартных зон, то размещение информации о доменах может быть реализовано следующим образом:

- записи, относящиеся к доменам microsoft.com и edu.microsoft.com, хранятся в одной зоне в файле `"microsoft.com.dns"` (на рисунке зона обозначена большим наклонным овалом);
- управление доменами sales.microsoft.com и it.microsoft.com делегировано другим серверам DNS, для этих доменов на других серверах созданы соответствующие файлы`"sales.microsoft.com.dns"` и `"it.microsoft.com.dns"` (данные зоны обозначены большими вертикальными овалами).

Делегирование управления - передача ответственности за часть пространства имен другим серверам DNS.

Зоны прямого и обратного просмотра

Зоны, рассмотренные в предыдущем примере, являются *зонами прямого просмотра (forward lookup zones)*. Данные зоны служат для разрешения имен узлов в IP-адреса. Наиболее часто используемые для этого типы записей: `A`, `CNAME`, `SRV`.

Для определения имени узла по его IP-адресу служат зоны обратного просмотра (*reverse lookup* zones), основной тип записи в "обратных" зонах - PTR. Для решения данной задачи создан специальный домен с именем `in-addr.arpa`. Для каждой IP-сети в таком домене создаются соответствующие поддомены, образованные из идентификатора сети, записанного в обратном порядке. Записи в такой зоне будут сопоставлять идентификатору узла полное FQDN-имя данного узла. Например, для IP-сети `192.168.0.0/24` необходимо создать зону с именем `"0.168.192.in-addr.arpa"`.

Для узла с IP-адресом `192.168.0.10` и именем `host.company.ru` в данной зоне должна быть создана запись `"10 PTR host.company.ru"`.

Алгоритмы работы итеративных и рекурсивных запросов DNS

Все запросы, отправляемые DNS-клиентом DNS-серверу для разрешения имен, делятся на два типа:

- итеративные запросы (клиент посылает серверу DNS запрос, в котором требует дать наилучший ответ без обращений к другим DNS-серверам);
- рекурсивные запросы (клиент посылает серверу DNS запрос, в котором требует дать окончательный ответ даже если DNS-серверу придется отправить запросы другим DNS-серверам; посылаемые в этом случае другим DNS-серверам запросы будут итеративными).

Обычные DNS-клиенты (например, рабочие станции пользователей), как правило, посылают рекурсивные запросы.

Рассмотрим на примерах, как происходит взаимодействие DNS-клиента и DNS-сервера при обработке итеративных и рекурсивных запросов.

Допустим, что пользователь запустил программу Обозреватель Интернета и ввел в адресной строке адрес http://www.microsoft.com. Прежде чем Обозреватель установит сеанс связи с веб-сайтом по протоколу HTTP, клиентский компьютер должен определить IP-адрес веб-сервера. Для этого клиентская часть протокола TCP/IP рабочей станции пользователя (так называемый *resolver*) сначала просматривает свой локальный кэш разрешенных ранее имен в попытке найти там имя www.microsoft.com. Если имя не найдено, то клиент посылает запрос DNS-серверу, указанному в конфигурации TCP/IP данного компьютера (назовем данный DNS-сервер "локальным DNS-сервером"), на разрешение имени www.microsoft.com в IP-адрес данного узла. Далее DNS-сервер обрабатывает запрос в зависимости от типа запроса.

Вариант 1 (итеративный запрос).

Если клиент отправил серверу итеративный запрос (напомним, что обычно клиенты посылают рекурсивные запросы), то обработка запроса происходит по следующей схеме:

- сначала локальный DNS-сервер ищет среди зон, за которые он отвечает, зону microsoft.com;

  если такая зона найдена, то в ней ищется запись для узла `www` ; если запись найдена, то результат поиска сразу же возвращается клиенту;

  в противном случае локальный DNS-сервер ищет запрошенное имя www.microsoft.com в своем кэше разрешенных ранее DNS-запросов;

  если искомое имя есть в кэше, то результат поиска возвращается клиенту; если локальный DNS-сервер не нашел в своей базе данных искомую запись, то клиенту посылается IP-адрес одного из корневых серверов DNS;

- клиент получает IP-адрес корневого сервера и повторяет ему запрос на разрешение имени www.microsoft.com;

  корневой сервер не содержит в своей БД зоны "microsoft.com", но ему известны DNS-серверы, отвечающие за зону "com", и корневой сервер посылает клиенту IP-адрес одного из серверов, отвечающих за эту зону;

- клиент получает IP-адрес сервера, отвечающего за зону "com", и посылает ему запрос на разрешение имени www.microsoft.com;

  сервер, отвечающий за зону com, не содержит в своей БД зоны microsoft.com, но ему известны DNS-серверы, отвечающие за зону microsoft.com, и данный DNS-сервер посылает клиенту IP-адрес одного из серверов, отвечающих уже за зону microsoft.com;

- клиент получает IP-адрес сервера, отвечающего за зону microsoft.com, и посылает ему запрос на разрешение имени www.microsoft.com;

  сервер, отвечающий за зону microsoft.com, получает данный запрос, находит в своей базе данных IP-адрес узла www, расположенного в зоне microsoft.com, и посылает результат клиенту;

  клиент получает искомый IP-адрес, сохраняет разрешенный запрос в своем локальном кэше и передает IP-адрес веб-сайта программе Обозреватель Интернета (после чего Обозреватель устанавливает связь с веб-сайтом по протоколу HTTP).

Вариант 2 (*рекурсивный запрос*).

Если клиент отправил серверу *рекурсивный запрос*, то обработка запроса происходит по такой схеме:

  сначала локальный DNS-сервер ищет среди зон, за которые он отвечает, зону microsoft.com; если такая зона найдена, то в ней ищется запись для узла www ; если запись найдена, то результат поиска сразу же возвращается клиенту;

  в противном случае локальный DNS-сервер ищет запрошенное имя www.microsoft.com в своем кэше разрешенных ранее DNS-запросов; если искомое имя есть в кэше, то результат поиска возвращается клиенту;

- если локальный DNS-сервер не нашел в своей базе данных искомую запись, то сам локальный DNS-сервер выполняет серию итеративных запросов на разрешение имени www.microsoft.com, и клиенту посылается либо найденный IP-адрес, либо сообщение об ошибке.

Реализация службы DNS в системах семейства Windows Server

Главная особенность службы DNS в системах семейства Windows Server заключается в том, что служба DNS разрабатывалась для поддержки службы каталогов Active Directory. Для выполнения этой функции требуются обеспечение двух условий:

- поддержка службой DNS динамической регистрации (dynamic updates);
- поддержка службой DNS записей типа SRV.

Служба DNS систем Windows Server удовлетворяет обоим условиям, и реализация служб каталогов Active Directory может быть обеспечена только серверами на базе систем Windows Server.

Рассмотрим несколько простых примеров управления службой DNS:

- установка службы DNS;
- создание основной и дополнительной зоны прямого просмотра;
- создание зоны обратного просмотра;

- выполнение динамической регистрации узлов в зоне.

Все рассматриваемые далее в пособии примеры были выполнены в следующей конфигурации:

- сеть состоит из двух серверов Windows 2003 Server;
- операционная система - ограниченная по времени 120-дневная русская версия Windows 2003 Server Enterprise Edition;
- первый сервер установлен на ПК с процессором Intel Pentium-4 3Ггц и оперативной памятью 512 МБ, имя сервера - DC1, IP-адрес - `192.168.0.1/24` ;
- второй сервер работает в качестве виртуальной системы с помощью Microsoft VirtualPC 2004, имя сервера -DC2, IP-адрес - `192.168.0.2/24` ;
- имя домена в пространстве DNS и соответствующее имя в службе каталогов Active Directory - world.ru (сеть полностью изолирована от других сетей, поэтому в данном примере авторы были свободны в выборе имени домена; в реальной обстановке конкретного учебного заведения преподавателю нужно скорректировать данную информацию).

Подробные рекомендации по организации сети для изучения данного курса (как под руководством преподавателя в организованной группе, так и при самостоятельном изучении) изложены в указаниях к выполнению упражнений лабораторных работ в конце пособия.

Установка службы DNS

Установка службы DNS (как и других компонент системы) производится достаточно просто с помощью мастера установки компонент Windows:

1. Откройте *Панель управления*.
2. Выберите пункт *"Установка и удаление программ"*.
3. Нажмите кнопку *"Установка компонентов Windows"*.
4. Выберите *"Сетевые службы"* - кнопка *"Дополнительно"* (ни в коем случае не снимайте галочку у названия *"Сетевые службы"*).
5. Отметьте службу DNS.

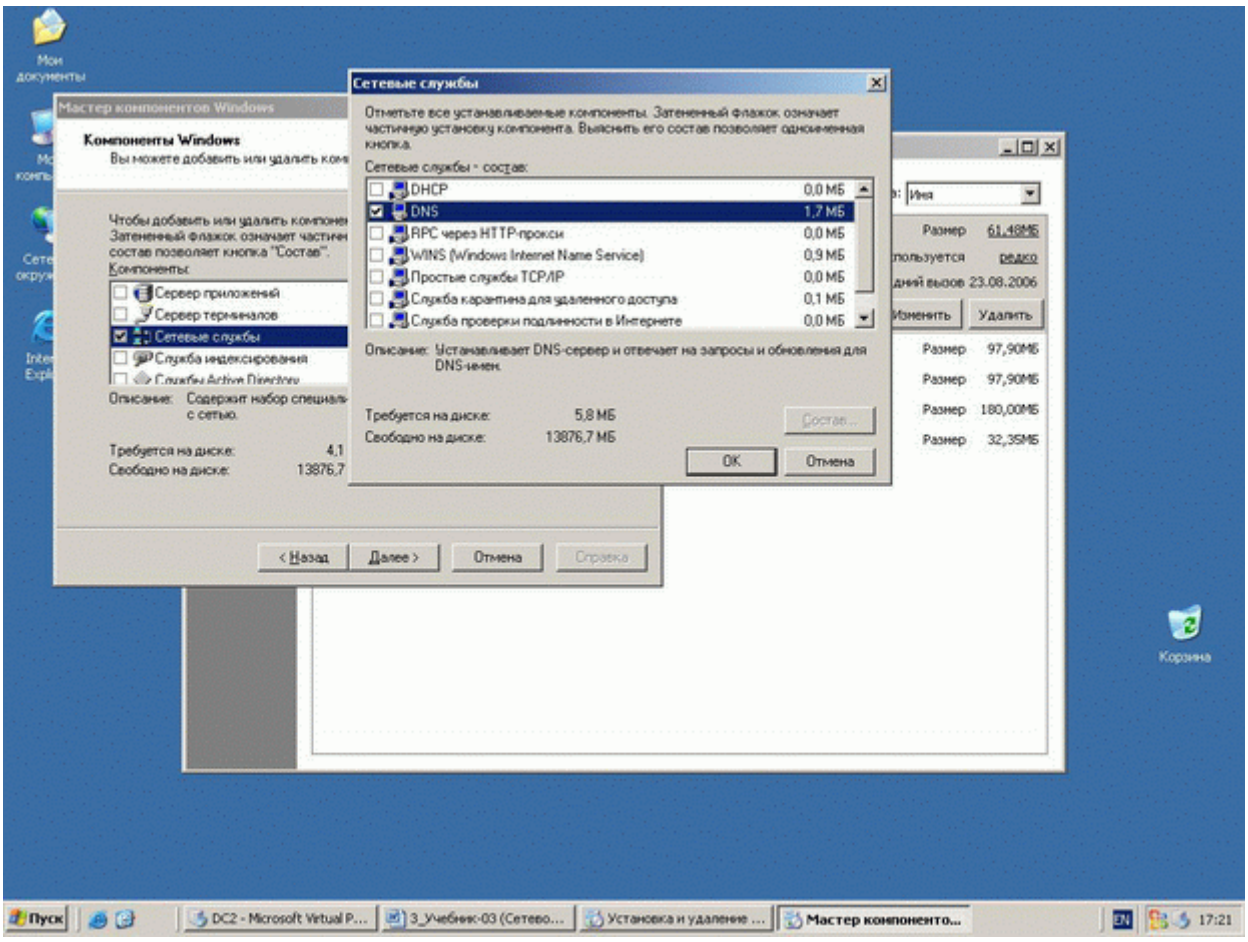


Рис. 4.10.

6. Кнопка *"OK"*, кнопка *"Далее"*, кнопка *"Готово"*.

   Если система попросит указать путь к дистрибутиву системы, введите путь к папке с дистрибутивом.

Выполним данное действие на обоих серверах.

Создание основной зоны прямого просмотра.

На сервере DC1 создадим стандартную основную зону с именем world.ru.

1. Откроем консоль DNS.
2. Выберем раздел *"Зоны прямого просмотра"*.
3. Запустим мастер создания зоны (тип зоны - *"Основная"*, динамические обновления - разрешить, остальные параметры - по умолчанию).
4. Введем имя зоны - `world.ru`.
5. Разрешим передачу данной зоны на любой сервер DNS (`Консоль DNS` - зона `world.ru` - *Свойства* - Закладка *"Передачи зон"* - Отметьте *"Разрешить передачи"* и *"На любой сервер"*).

Создание дополнительной зоны прямого просмотра.

На сервере DC2 создадим стандартную дополнительную зону с именем world.ru.

1. Откроем консоль DNS.
2. Выберем раздел *"Зоны прямого просмотра"*

3. Запустим мастер создания зоны (выбрать: тип зоны - *"Дополнительная"*, IP-адрес master-сервера (с которого будет копироваться зона) - адрес сервера DC1, остальные параметры - по умолчанию)
4. Введем имя зоны - `world.ru`.
5. Проверим в консоли DNS появление зоны.

Настройка узлов для выполнения динамической регистрации на сервер DNS.

Для выполнения данной задачи нужно выполнить ряд действий как на сервере DNS, так и в настройках клиента DNS.

Сервер DNS.

1. Создать соответствующую зону.
2. Разрешить динамические обновления.

Это нами уже выполнено.

Клиент DNS.

1. Указать в настройках протокола TCP/IP адрес предпочитаемого DNS-сервера - тот сервер, на котором разрешены динамические обновления (в нашем примере - сервер DC1).
2. В полном имени компьютера указать соответствующий DNS-суффикс (в нашем примере - `world.ru`). Для этого - *"Мой компьютер"* - *"Свойства"* - Закладка *"Имя компьютера"* - Кнопка*"Изменить"* - Кнопка *"Дополнительно"* - в пустом текстовом поле впишем название домена `world.ru` - кнопка *"ОК"* (3 раза)).

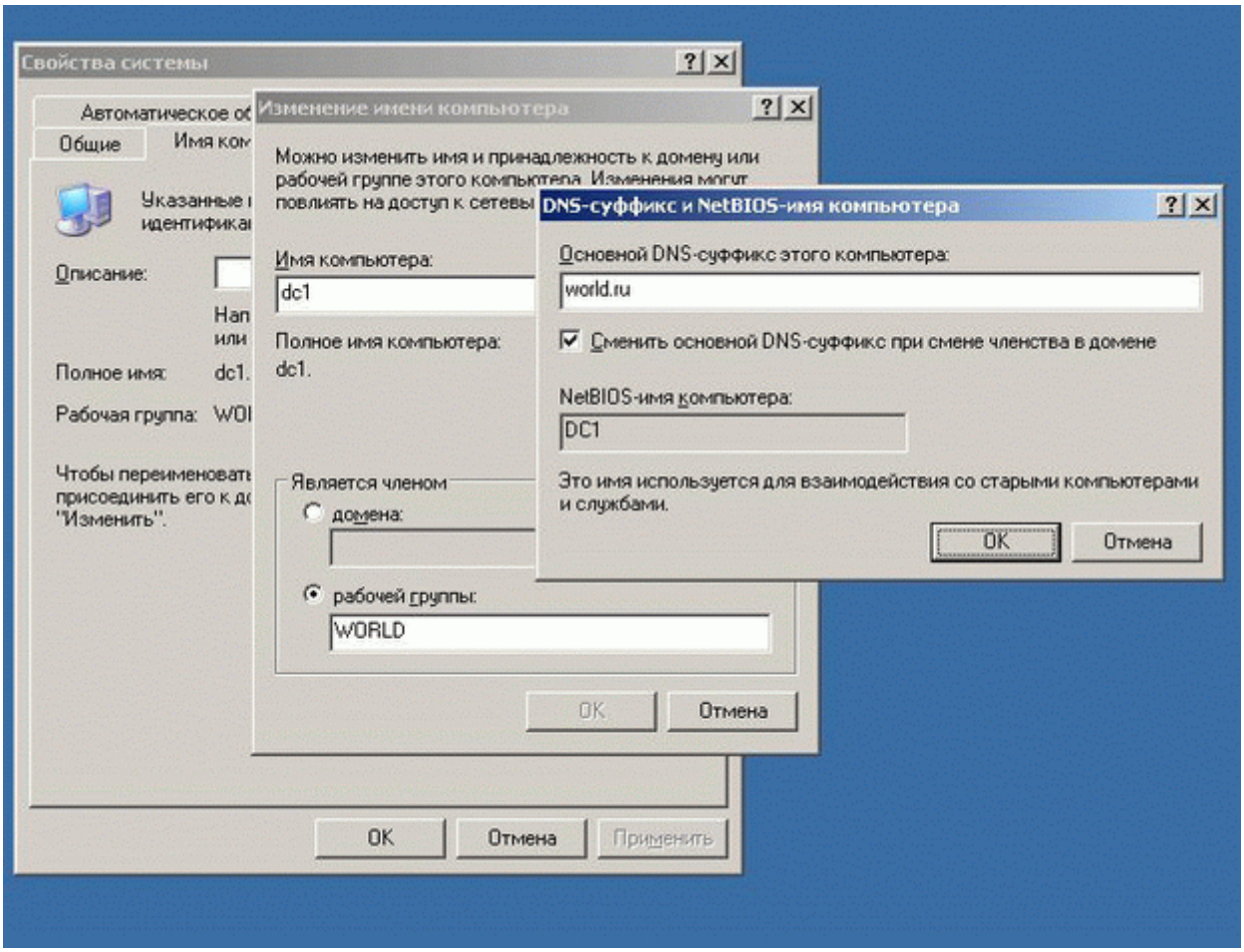


Рис. 4.11.
После этого система предложит перезагрузить компьютер. После выполнения перезагрузки на сервер DNS в зоне `world.ru` автоматически создадутся записи типа `A` для наших серверов (рис. 4.12).

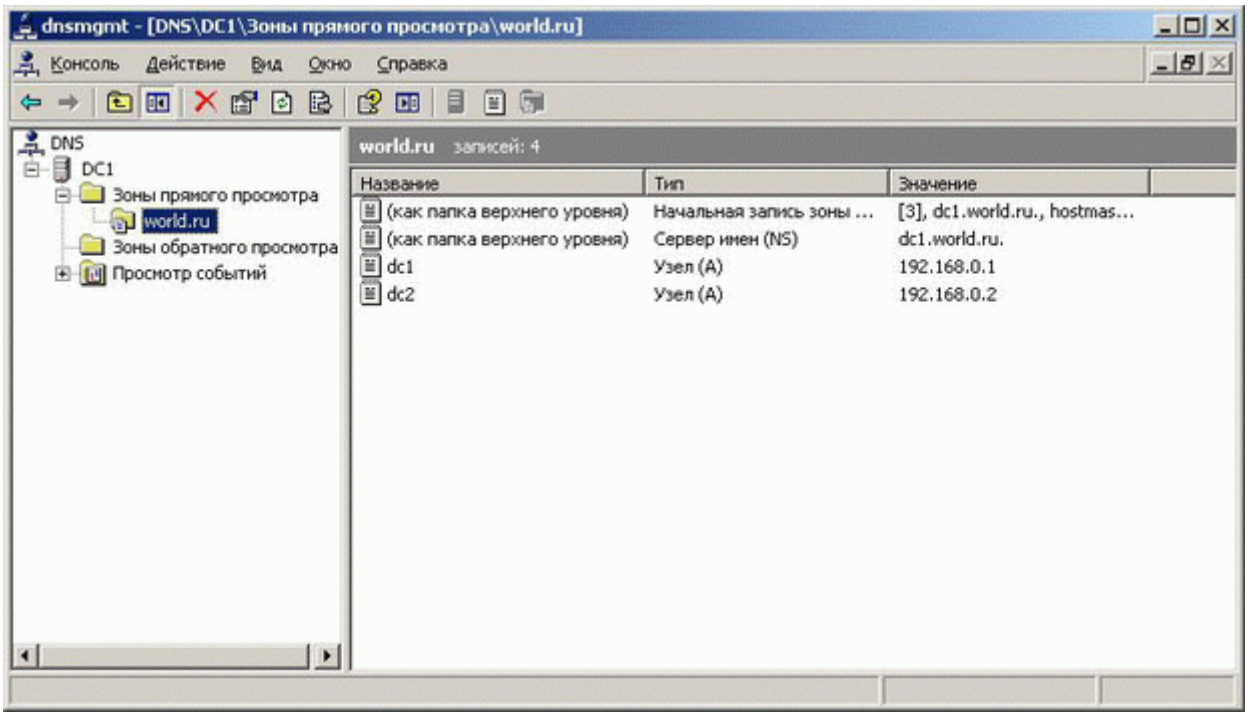


Рис. 4.12.
Создание зоны обратного просмотра.

1. Откроем консоль DNS.
2. Выберем раздел *"Зоны обратного просмотра"*.
3. Запустим мастер создания зоны (выбрать: тип зоны - *"Основная"*, динамические обновления - разрешить, остальные параметры - по умолчанию)
4. В поле *"Код сети (ID)"* введем параметры идентификатора сети - `192.168.0`.
5. Выполним команду принудительной регистрации клиента на сервере DNS - `ipconfig /registerdns`.

Наши серверы зарегистрируются в *обратной зоне* DNS (рис. 4.13):

4.3 Диагностические утилиты TCP/IP и DNS.

Любая операционная система имеет набор диагностических утилит для тестирования сетевых настроек и функционирования коммуникаций. Большой набор диагностических средств есть и в системах семейства Windows (как графических, так и режиме командной строки).

Перечислим утилиты командной строки, являющиеся инструментами первой необходимости для проверки настроек протокола TCP/IP и работы сетей и коммуникаций. Подробное описание данных утилит содержится в системе интерактивной помощи Windows. В Таблице 6 укажем основные и наиболее часто используемые параметры этих команд и дадим их краткое описание.

Таблица 4.6.

Название	Параметры	Комментарии

утилиты		
ipconfig	/? - Отобразить справку по команде /all - Отобразить полную информацию о настройке параметров всех адаптеров /release - Освободить динамическую IP-конфигурацию /renew - Обновить динамическую IP-конфигурацию с DHCP-сервера /flushdns - Очистить кэш разрешений DNS /registerdns - Обновить регистрацию на DNS-сервере /displaydns - Отобразить содержимое кэша разрешений DNS	Служит для отображения всех текущих параметров сети TCP/IP и обновления параметров DHCP и DNS. При вызове команды ipconfig без параметров выводятся IP-адрес, маска подсети и основной шлюз для каждого сетевого адаптера.
arp	-a - Отображает текущие ARP-записи	Отображение и изменение ARP-таблиц.
ping	Формат команды: "ping <сетевой узел> параметры" Параметры: -t - Бесконечная (до нажатия клавиш <Ctrl>+<Break>) отправка пакетов на указанный узел -a - Определение имени узла по IP-адресу -n <число> - Число отправляемых запросов -l <размер> - Размер буфера отправки -w <таймаут> - Таймаут ожидания каждого ответа в миллисекундах	Мощный инструмент диагностики (с помощью протокола ICMP). Команда ping позволяет проверить: • работоспособность IP-соединения; • правильность настройки протокола TCP/IP на узле; • работоспособность маршрутизаторов; • работоспособность системы разрешения имен FQDN или NetBIOS; • доступность и работоспособность какого-либо сетевого ресурса.
tracert	-d - Без разрешения IP-адресов в имена узлов -h <максЧисло> - Максимальное число прыжков при поиске узла -w <таймаут> - Таймаут каждого ответа в миллисекундах	Служебная программа для трассировки маршрутов, используемая для определения пути, по которому IP-дейтаграмма доставляется по месту назначения.
pathping	-n - Без разрешения IP-адресов в имена узлов	Средство трассировки маршрута, сочетающее функции программ ping и tracert и обладающее

	`-h максЧисло` - Максимальное число прыжков при поиске узла `-q <число_запросов>` - Число запросов при каждом прыжке `-w <таймаут>` - Таймаут каждого ответа в миллисекундах	дополнительными возможностями. Эта команда показывает степень потери пакетов на любом маршрутизаторе или канале, с ее помощью легко определить, какие маршрутизаторы или каналы вызывают неполадки в работе сети.
`netstat`	`-a` - Отображение всех подключений и ожидающих (слушающих) портов `-n` - Отображение адресов и номеров портов в числовом формате `-o` - Отображение кода (ID) процесса каждого подключения `-r` - Отображение содержимого локальной таблицы маршрутов	Используется для отображения статистики протокола и текущих TCP/IP-соединений.
`nbtstat`	`-n` - Выводит имена пространства имен NetBIOS, зарегистрированные *локальными процессами* `-c` - Отображает кэш имен NetBIOS (разрешение NetBIOS-имен в IP-адреса) `-R` - Очищает кэш имен и перезагружает его из файла Lmhosts `-RR` - Освобождает имена NetBIOS, зарегистрированные на WINS-сервере, а затем обновляет их регистрацию	Средство диагностики разрешения имен NetBIOS

Примеры использования утилит командной строки.

Пример 1. Команда `ipconfig` (без параметров и с параметром `/all`).

```
C:\>ipconfig

Настройка протокола IP для Windows

Подключение по локальной сети - Ethernet адаптер:

   DNS-суффикс этого подключения . . :
   IP-адрес  . . . . . . . . . . . . : 192.168.0.1
   Маска подсети . . . . . . . . . . : 255.255.255.0
   Основной шлюз . . . . . . . . . . :

C:\> ipconfig /all

     Настройка протокола IP для Windows

        Имя компьютера  . . . . . . . . . : dc1
        Основной DNS-суффикс  . . . . . . : world.ru
        Тип узла. . . . . . . . . . . . . : неизвестный
        IP-маршрутизация включена . . . . : нет
        WINS-прокси включен . . . . . . . : нет
        Порядок просмотра суффиксов DNS . : world.ru

     Подключение по локальной сети - Ethernet адаптер:

        DNS-суффикс этого подключения . . :
        Описание  . . . . . . . . . . . . : Realtek RTL8139 Family PCI Fast
Ethernet NIC
        Физический адрес. . . . . . . . . : 00-11-D8-E7-14-F4
        DHCP включен. . . . . . . . . . . : нет
        IP-адрес  . . . . . . . . . . . . : 192.168.0.1
        Маска подсети . . . . . . . . . . : 255.255.255.0
        Основной шлюз . . . . . . . . . . :
        DNS-серверы . . . . . . . . . . . : 192.168.0.1
```

Рис. 4.14.
Пример 2. Команда `arp`.

Поскольку в нашей сети только два узла, то в кэше сервера DC1 будет только одна запись - отображение IP-адреса сервера DC2 на MAC-адрес сетевого адаптера.

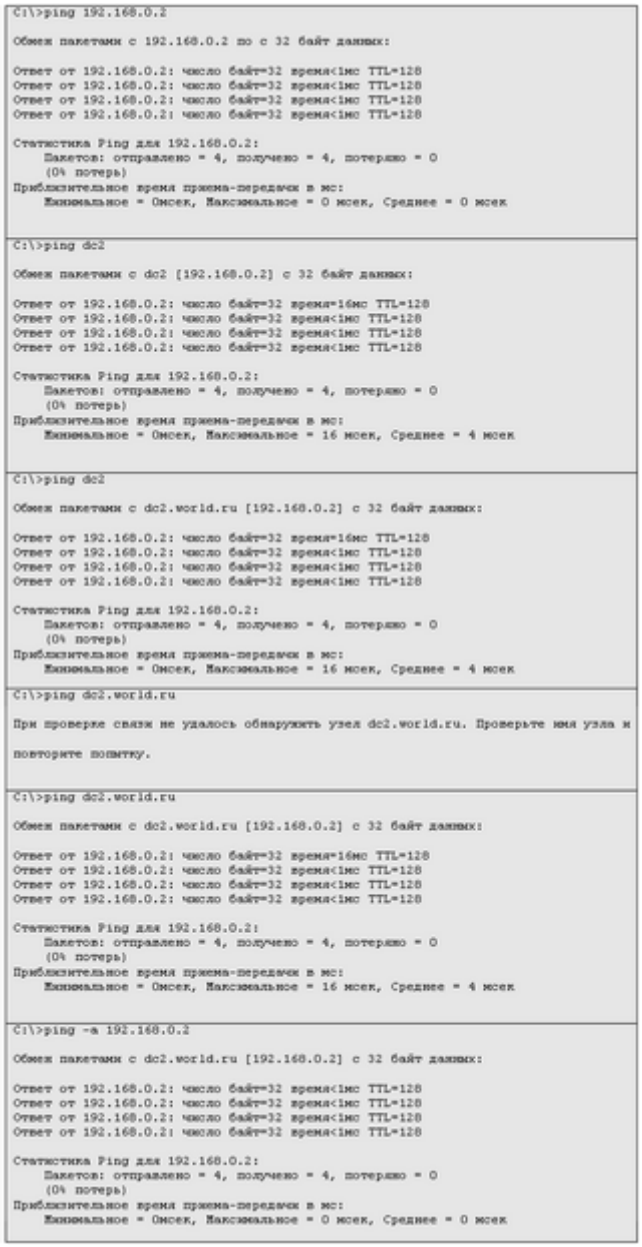
```
C:\>arp -a
    Интерфейс: 192.168.0.1 --- 0x10003
      IP-адрес              Физический адрес      Тип
      192.168.0.2           00-03-ff-e7-14-f4     динамический
```

Рис. 4.15.
Пример 3. Команда `ping`.

Варианты использования:

- `ping <IP-адрес>` ;
- `ping <NetBIOS-имя узла>`, когда в зоне сервера DNS нет записи для сервера DC2 (поиск IP-адреса производится широковещательным запросом);
- `ping <NetBIOS-имя узла>`, когда в зоне сервера DNS есть запись для сервера DC2 (надо обратить внимание на подстановку клиентом DNS суффикса домена в запросе на имя узла, т.е в команде используется краткое NetBIOS-имя сервера, а в статистике команды выводится полное имя);
- `ping <FQDN-имя узла>`, когда в зоне сервера DNS нет записи для сервера DC2 (узел DC2 не будет найдет в сети);

- `ping <FQDN-имя узла>`, когда в зоне сервера DNS есть запись для сервера DC2 (узел успешно найден);
- `ping -a <IP-адрес>` (обратное разрешение IP-адреса в имя узла)



увеличить изображение

Рис. 4.16.

Пример 4. Команда `tracert`.

Трассировка маршрута до узла www.ru (если в вашем распоряжении только одна IP-сеть, то изучить работу данной команды будет невозможно).

```
C:\>tracert -d www.ru

Трассировка маршрута к www.ru [194.87.0.50]
с максимальным числом прыжков 30:

  1    17 ms    <1 мс    <1 мс  192.168.0.1
  2     1 ms    <1 мс     1 ms  217.1.1.33
  3     3 ms     3 ms     3 ms  217.1.10.1
  4     *        *        *     Превышен интервал ожидания для запроса.
  5     *        *        *     Превышен интервал ожидания для запроса.
  6    10 ms    11 ms    10 ms  217.150.36.190
  7     *        *        *     Превышен интервал ожидания для запроса.
  8    13 ms    13 ms    15 ms  194.87.0.83
  9    17 ms    12 ms    12 ms  194.87.0.50

Трассировка завершена.
```

Рис. 4.17.
Пример 5. Команда `pathping`.

Аналогичная задача (трассировка маршрута до узла www.ru), выполненная командой `pathping`.

```
C:\> pathping -n www.ru

Трассировка маршрута к www.ru [194.87.0.50]
с максимальным числом прыжков 30:
  0  192.168.0.1
  1  217.1.1.33
  2  217.1.10.1
  3     *        *        *
Подсчет статистики за: 100 сек. ...
           Исходный узел     Маршрутный узел
Прыжок  RTT    Утер./Отпр.   %   Утер./Отпр.  %   Адрес
  0    0мс      0/ 100 =  0%     0/ 100 =  0%  192.168.0.1
                                 0/ 100 =  0%   |
  1    2мс      0/ 100 =  0%     0/ 100 =  0%  217.1.1.33
                                 0/ 100 =  0%   |
  2    5мс      0/ 100 =  0%     0/ 100 =  0%  217.1.10.1
                                100/ 100 =100%   |
  3   ---     100/ 100 =100%     0/ 100 =  0%  0.0.0.0

Трассировка завершена.
```

Рис. 4.18.
Пример 6. Команда `netstat` (с параметрами `-an` - отображение в числовой форме списка активных подключений и слушающих портов).

```
C:\> netstat -an

Активные подключения

  Имя    Локальный адрес         Внешний адрес          Состояние
  TCP    0.0.0.0:53              0.0.0.0:0              LISTENING
  TCP    0.0.0.0:135             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1029            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1030            0.0.0.0:0              LISTENING
  TCP    192.168.0.1:139      0.0.0.0:0              LISTENING
  UDP    0.0.0.0:445             *:*
  UDP    0.0.0.0:500             *:*
  UDP    0.0.0.0:1025            *:*
  UDP    0.0.0.0:1028            *:*
  UDP    0.0.0.0:1035            *:*
  UDP    0.0.0.0:4500            *:*
  UDP    127.0.0.1:53            *:*
  UDP    127.0.0.1:123           *:*
  UDP    127.0.0.1:1026          *:*
  UDP    127.0.0.1:1027          *:*
  UDP    192.168.0.1:53       *:*
  UDP    192.168.0.1:123      *:*
  UDP    192.168.0.1:137      *:*
  UDP    192.168.0.1:138      *:*
```

Рис. 4.19.
Пример 7. Команда `nbtstat` (с параметром `-n` - отображение локальных имен NetBIOS).

```
C:\> nbtstat -n

Подключение по локальной сети:
Адрес IP узла: [192.168.0.1] Код области: []

                Локальная таблица NetBIOS-имен

       Имя                Тип          Состояние
    ---------------------------------------------------
    DC1             <00>  Уникальный   Зарегистрирован
    DC1             <20>  Уникальный   Зарегистрирован
    WORLD           <00>  Группа       Зарегистрирован
    WORLD           <1E>  Группа       Зарегистрирован
```

Рис. 4.20.
В упражнениях лабораторных работ к данному разделу предусмотрено выполнение аналогичных и дополнительных примеров с комментариями преподавателя.

Резюме

Первая часть данного раздела посвящена базовым понятиям протокола TCP/IP, являющегося в настоящее время основным протоколом корпоративных сетей и глобальной сети Интернет:

- адресация узлов в IP-сетях (понятие IP-адреса узла, маски подсети, структуры IP-адреса сетевого узла, основного шлюза, классов IP-сетей);
- публичные и приватные IP-сети;
- протокол ARP;
- разбиение на подсети с помощью маски подсети;
- понятие IP-маршрутизации (доставка пакетов между узлами, находящимися в одной подсети, и узлами, находящимися в различных подсетях).

Во второй части изложены основы функционирования службы DNS:

- иерархическое древовидное пространство имен DNS;
- распределенная база данных DNS;
- серверы DNS и клиенты DNS;
- связь между доменами и зонами;
- типы зон - основные (primary) и дополнительные (secondary), репликация изменений между зонами;
- зоны прямого и обратного просмотра;
- рекурсивные и итеративные запросы DNS;
- динамическая регистрация узлов в зонах DNS;
- особенности службы DNS в системах семейства Windows Server;
- установка и настройка службы DNS в системах семейства Windows Server.

Третья часть раздела содержит краткое описание наиболее часто используемых утилит командной строки для диагностики и настройки протокола TCP/IP и службы DNS с примерами применения этих утилит.

Задачи сетевого администратора при управлении инфраструктурой протокола TCP/IP:

- планирование пространства IP-адресов для внутренних и внешних сегментов корпоративной сети;
- планирование подсетей главного офиса и подразделений компании или организации;
- планирование (если есть необходимость) разбиения сетей на подсети с помощью маски подсети;
- настройка маршрутизации между отдельными подсетями корпоративной сети.

Задачи сетевого администратора при управлении инфраструктурой службы DNS:

- планирование пространства доменных имен компании или организации, как для внешних, так и для внутренних сегментов сети;
- планирование, установка и настройка серверов DNS;
- обеспечение высокой эффективности функционирования серверов DNS;
- обеспечение отказоустойчивой работы серверов DNS;
- защита серверов DNS от внешних и внутренних атак.

```
C:\>ipconfig

Настройка протокола IP для Windows

Подключение по локальной сети - Ethernet адаптер:

   DNS-суффикс этого подключения . . :
   IP-адрес  . . . . . . . . . . . . : 192.168.0.1
   Маска подсети . . . . . . . . . . : 255.255.255.0
   Основной шлюз . . . . . . . . . . :

C:\> ipconfig /all

     Настройка протокола IP для Windows

        Имя компьютера  . . . . . . . . . : dc1
        Основной DNS-суффикс  . . . . . . : world.ru
        Тип узла. . . . . . . . . . . . . : неизвестный
        IP-маршрутизация включена . . . . : нет
        WINS-прокси включен . . . . . . . : нет
        Порядок просмотра суффиксов DNS . : world.ru

     Подключение по локальной сети - Ethernet адаптер:

        DNS-суффикс этого подключения . . :
        Описание  . . . . . . . . . . . . : Realtek RTL8139 Family PCI Fast
Ethernet NIC
        Физический адрес. . . . . . . . . : 00-11-D8-E7-14-F4
        DHCP включен. . . . . . . . . . . : нет
        IP-адрес  . . . . . . . . . . . . : 192.168.0.1
        Маска подсети . . . . . . . . . . : 255.255.255.0
        Основной шлюз . . . . . . . . . . :
        DNS-серверы . . . . . . . . . . . : 192.168.0.1
```

Рис. 4.13.