

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
Кафедра компьютерных наук и экспериментальной математики
Кафедра информационных систем и компьютерного моделирования

Н.М. Полубоярова, В.В. Полубояров

ОПЕРАЦИОННЫЕ СИСТЕМЫ

*Учебно-методическое пособие к лабораторному практикуму
для бакалавров, обучающихся по направлениям подготовки
230100.62 Информатика и вычислительная техника,
230400.62 Информационные системы и технологии,
231000.62 Программная инженерия,
230700.62 Прикладная информатика,
010500.62 Математическое обеспечение
и администрирование информационных систем*

Волгоград 2013

УДК 004.45(075.8)
ББК 32.973-018.2я73
П53

Печатается по решению редакционно-издательского совета
Волгоградского государственного университета

Рецензенты:

А.М. Дворянкин, профессор, доктор технических наук, проректор
по учебной работе, заведующий кафедрой программного обеспечения
автоматизированных систем Волгоградского государственного технического
университета; **Е.А. Петрова**, доктор экономических наук,
заведующий кафедрой экономической информатики и управления
Волгоградского государственного университета

Полубоярова, Н. М.

П53 Операционные системы [Текст]: учеб.-метод. пособие к лаб.
практикуму для бакалавров, обучающихся по направлениям подгот.
230100.62 Информатика и вычисл. техника, 230400.62 Информ. системы и
технологии, 231000.62 Прогр. инженерия, 230700.62 Прикл. информатика,
010500.62 Мат. обеспечение и администрирование информ. систем
/ Н. М. Полубоярова, В. В. Полубояров ; Федер. гос. авт. образоват.
учреждение высш. проф. образования «Волгогр. гос. ун-т», Ин-т
математики и информ. технологий, Каф. компьютер. наук и эксперим.
математики, Каф. информ. систем и компьютер. моделирования. –
Волгоград : Изд-во ВолГУ, 2013. – 90 с. : ил.

ISBN 978-5-9669-1173-7

Пособие к лабораторному практикуму включает две лабораторные работы,
нацеленные на приобретение практических навыков установки операционной
системы Windows 2008 Server и конфигурирования базовых сетевых служб.

Предназначено для студентов старших курсов, обладающих базовыми навыками
в области администрирования и практическими навыками работы на персональном
компьютере.

УДК 004.45(075.8)
ББК 32.973-018.2я73

ISBN 978-5-9669-1173-7



- © Полубоярова Н.М., Полубояров В.В., 2013
- © ФГАОУ ВПО «Волгоградский государственный университет», 2013
- © Оформление. Издательство Волгоградского государственного университета, 2013

ВВЕДЕНИЕ

Учебно-методическое пособие направлено на улучшение усвоения теоретических положений курса «Операционные системы» бакалаврами, обучающимися по направлениям подготовки 230100.62 Информатика и вычислительная техника, 230400.62 Информационные системы и технологии, 231000.62 Программная инженерия, 230700.62 Прикладная информатика, 010500.62 Математическое обеспечение и администрирование информационных систем.

В пособии приведены рекомендации для выполнения следующих лабораторных работ:

- лабораторная работа № 1. СТЕК ПРОТОКОЛОВ TCP/IP;
- лабораторная работа № 2. СЛУЖБА DNS.

Лабораторная работа № 1

СТЕК ПРОТОКОЛОВ TCP/IP

Цель работы

Установка, настройка и конфигурация сетевой подсистемы ОС Windows 2008 Server как одиночного сервера инфраструктуры.

Теоретическое введение

Стек TCP/IP (Transmission Control Protocol/Internet Protocol) – это набор протоколов, предназначенный для поддержки внутри- и межсетевых связей. Является базовым протоколом сети Internet. TCP/IP поддерживается всеми современными сетевыми операционными системами. Предоставляет также отказоустойчивую, масштабируемую и кросс-платформенную инфраструктуру.

Архитектура TCP/IP

Протоколы TCP/IP соответствуют четырехуровневой концептуальной модели, известной как модель DARPA. Архитектура протоколов TCP/IP показана на рисунке 1.

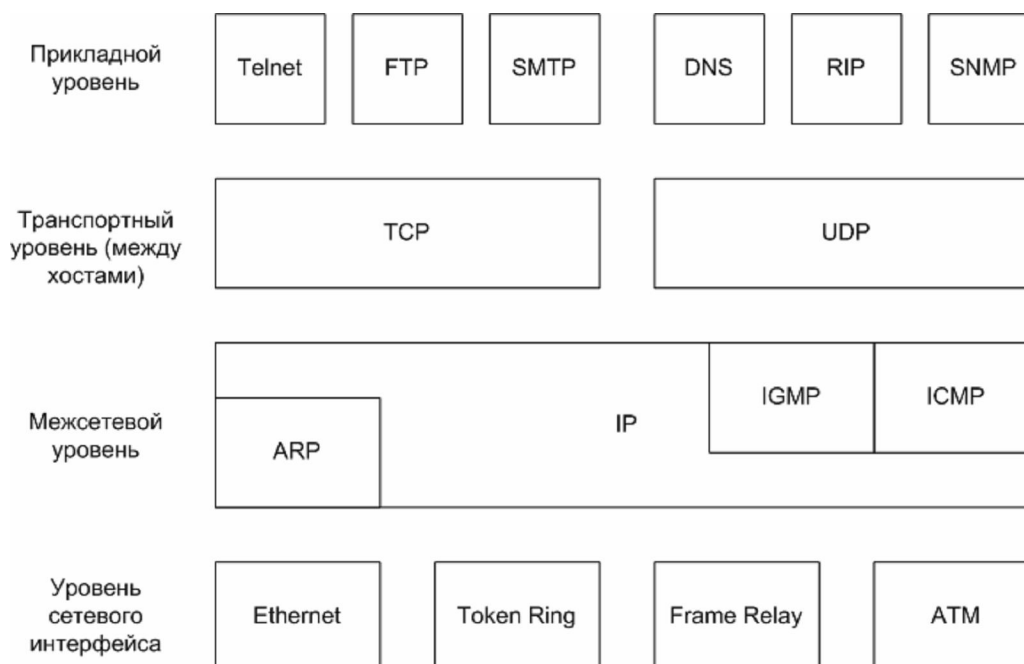


Рис. 1. Архитектура TCP/IP

Уровень сетевого интерфейса отвечает за передачу TCP/IP-пакетов в сетевую среду и прием этих пакетов из сетевой среды. TCP/IP независим от способа доступа к сети, формата кадров и сетевой среды. Благодаря этому TCP/IP можно использовать для соединения сетей разных типов. Независимость от сетевой технологии позволяет адаптировать TCP/IP к новым технологиям передачи данных. Уровень сетевого интерфейса считается ненадежным – за поддержание надежной коммуникационной связи отвечает транспортный уровень.

Межсетевой уровень отвечает за поддержку адресации, пакетов и маршрутизации.

Базовые протоколы этого уровня:

- IP (Internet Protocol) – маршрутизируемый протокол, отвечающий за IP-адресацию, маршрутизацию, а также за фрагментацию и восстановление пакетов.
- ARP (Address Resolution Protocol) – обеспечивает преобразование адресов меж сетевого уровня в адреса уровня сетевого интерфейса.
- ICMP (Internet Control Message Protocol) – поддерживает диагностические функции и сообщает об ошибках в случае неудачной доставки IP-пакетов.
- IGMP (Internet Group Management Protocol) – управляет группами IP-рассылки (IP multicast groups).

Транспортный уровень, также известный как «уровень транспорта между хостами» (host-to-host transport layer), предоставляет прикладному уровню сеансовые коммуникационные службы и обеспечивает поддержку дейтаграмм. Базовые протоколы этого уровня:

– TCP (Transmission Control Protocol) – обеспечивает надежную, требующую логического соединения коммуникационную связь по типу «один-к-одному». TCP отвечает за установление TCP-соединения, упорядочение посылаемых пакетов, подтверждение приема поступающих пакетов и восстановление пакетов, потерянных в процессе передачи.

– UDP (User Datagram Protocol) – обеспечивает ненадежную, не требующую логического соединения коммуникационную связь по типу «один-к-одному» или «один-ко-многим». UDP используется,

когда объем передаваемых данных невелик (например, данные могут уместиться в единственном пакете), когда издержки установления TCP-соединения нежелательны либо когда приложения или протоколы верхних уровней гарантируют надежную доставку.

Прикладной уровень обеспечивает приложениям доступ к сервисам других уровней и определяет протоколы, по которым приложения могут обмениваться данными. На прикладном уровне предусмотрено большое количество протоколов, и постоянно разрабатываются новые.

Следующие протоколы прикладного уровня предназначены для обмена пользовательской информацией:

- HTTP (Hypertext Transfer Protocol) – протокол для передачи файлов, образующих содержимое Web-страниц в World Wide Web;
- FTP (File Transfer Protocol) – протокол для интерактивной передачи файлов;
- SMTP (Simple Mail Transfer Protocol) – протокол для передачи почтовых сообщений и вложений;
- Telnet – протокол эмуляции терминала – используется для регистрации на удаленных сетевых хостах.

Следующие протоколы прикладного уровня предназначены для управления IP-сетями:

- DNS (Domain Name System) – предназначен для разрешения хост-имен в IP-адреса;
- RIP (Routing Information Protocol) – применяется маршрутизаторами для обмена соответствующей информацией;
- SNMP (Simple Network Management Protocol) – обеспечивает взаимодействие между консолью управления сетью и сетевыми устройствами (маршрутизаторами, мостами, «интеллектуальными» хабами), позволяя собирать информацию, необходимую для управления сетью, и обмениваться ею.

Прикладной уровень имеет API-интерфейсы для приложений, использующих TCP/IP, – Windows Sockets и NetBIOS.

IP-адресация

TCP/IP-хост идентифицируется логическим IP-адресом. IP-адрес – это адрес сетевого межсетевого уровня, независимый от адреса сетевого уровня (например, от MAC-адреса сетевого адаптера). Уникальный IP-адрес необходим для каждого хоста и сетевого компонента, использующего коммуникационную связь по TCP/IP. IP-адреса должны быть глобально уникальными и иметь единый формат.

Любой IP-адрес включает:

- идентификатор сети (network ID), также известный как сетевой адрес, определяет системы, которые находятся в одной физической сети, ограниченной маршрутизаторами. Идентификатор сети должен быть одинаков у всех систем в одной физической сети и уникален в межсетевом пространстве;
- идентификатор хоста (host ID), также известный как адрес хоста, определяет рабочую станцию, сервер, маршрутизатор или другой TCP/IP-хост в сети. Адрес каждого хоста должен быть уникален для данного идентификатора сети.

IP-адрес состоит из 32 битов, которые разбиваются на четыре октета – поля по 8 битов.

Каждый октет преобразуется в десятичное число в диапазоне 0...255 и отделяется точкой.

Такой формат называется точечно-десятичной нотацией. В таблице 1 дан пример IP-адреса в двоичном и точечно-десятичном форматах.

Таблица 1. **Пример IP-адреса в двоичном и точечно-десятичном формате**

Двоичный формат	Десятичный формат
11000000 10101000 00000011 00011000	192.168.3.24

Классы IP-адресов

Определено пять классов адресов, соответствующих сетям различных размеров. Microsoft TCP/IP поддерживает адреса классов А, В и С. Класс адреса задает, сколько битов в IP-адресе отводится

под идентификаторы сети и хоста «А», значит, класс адреса также определяет максимальное количество сетей данного класса и хостов в каждой из этих сетей. Допустимые диапазоны идентификаторов сети, основанные на классах IP-адресов, приведены в таблице 2.

Таблица 2. Диапазоны идентификаторов сети для разных классов IP-адресов

Класс	Первый идентификатор сети	Последний идентификатор сети	Комментарий
A	1.0.0.0	126.0.0.0	Сети с очень большим (до 16 млн) числом хостов
B	128.0.0.0	191.255.0.0	Сети большого и среднего размера (до 65,5 тыс. хостов)
C	192.0.0.0	223.255.255.0	Малые сети (до 254 хостов)

Идентификатор хоста определяет TCP/IP-хост в сети. Комбинация идентификаторов сети и хоста образует IP-адрес. Допустимые диапазоны идентификаторов хоста, основанные на классах IP-адресов, перечислены в таблице 3.

Таблица 3. Допустимые диапазоны идентификаторов хоста, основанные на классах IP-адресов

Класс	Первый идентификатор хоста	Последний идентификатор хоста
A	w.0.0.1	w.255.255.254
B	w.x.0.1	w.x.255.254
C	w.x.y.1	w.x.y.254

Подсети и маски подсетей

Сети класса А и В содержат огромное количество узлов, а все хосты в физической сети делят один и тот же широкополосный трафик. В связи с этим IP-сеть можно разбить на несколько меньших сетей (подсетей), разграничив их IP-маршрутизаторами и присвоив каждой из них свой идентификатор сети, включающий идентификатор

подсети (subnetted network ID). Последний формируется из части битов, отводимых под идентификатор хоста в данном классе IP-адресов.

Маска подсети (также называемая маской адреса) определена в RFC 950 как 32-битное значение, используемое для того, чтобы отличить идентификатор сети от идентификатора хоста в произвольном IP-адресе. Биты маски подсети задаются следующим образом:

- все биты, соответствующие идентификатору сети, устанавливаются в 1;
- все биты, соответствующие идентификатору хоста, устанавливаются в 0.

Любой хост в TCP/IP-сети требует наличия маски подсети, даже если эта сеть состоит из единственного сегмента. Поэтому в каждом TCP/IP-узле применяется либо стандартная маска подсети (для идентификаторов сетей на основе классов), либо нестандартная (для идентификаторов сетей, включающих идентификаторы подсетей или надсетей).

Маски подсетей часто записываются в точечно-десятичной нотации. Установив все биты для идентификаторов сети и хоста, полученное 32-битное значение преобразуют в точечно-десятичную форму.

Стандартная маска подсети основана на классе IP-адреса и используется в TCP/IP-сетях, не разбитых на подсети. Стандартные маски подсетей в точечно-десятичной нотации перечислены в таблице 4.

Таблица 4. Стандартные маски подсетей

Класс	Биты маски подсети	Маска подсети	Префикс сети
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

Поскольку биты идентификатора сети всегда начинаются со старших битов IP-адреса, маску подсети можно выразить в более краткой нотации, просто указав число битов идентификатора сети. Это

число записывается в виде /<количество битов> и называется префиксом сети (network prefix). Как выглядят стандартные маски IP-сетей в виде префиксов, показано в Таблица 4. Представление маски подсети в виде префикса сети также называется нотацией CIDR (Classless Interdomain Routing). Так как у всех хостов в одной сети должен быть одинаковый идентификатор сети, они должны использовать и одинаковую маску подсети.

Определение идентификатора сети

Чтобы извлечь идентификатор сети из IP-адреса с применением произвольной маски подсети, IP использует логическое сравнение AND. При этом результат сравнения двух элементов является истинным, только если истинны оба сравниваемых элемента; в ином случае результат ложен. Если применить этот принцип к битам, то результат сравнения равен 1, когда оба сравниваемых бита равны 1; в остальных случаях результат равен 0. IP сравнивает таким образом 32-битные IP-адрес и маску подсети. Это называется побитовой операцией AND. Результатом сравнения данных значений является идентификатор сети. Например, требуется определить идентификатор сети для IP-узла с адресом 129.56.189.41 и маской подсети 255.255.240.0. Чтобы получить результат, преобразуем обе величины в двоичные значения, а затем выполним над ними побитовую операцию AND:

10000001	00111000	10111101	00101001	IP-адрес
11111111	11111111	11110000	00000000	Маска подсети
10000001	00111000	10110000	00000000	ID сети

Итак, логическое сравнение AND битов IP-адреса и маски подсети дает идентификатор сети, равный 129.56.176.0.

Создание надсетей и метод CIDR

В результате быстрого развития Интернета стало ясно, что идентификаторы сетей класса В вскоре будут исчерпаны. Для большинства организаций класс С дает слишком мало

идентификаторов хостов, тогда как класс В обеспечивает требуемую гибкость в формировании подсетей.

Поэтому был придуман новый метод присвоения идентификаторов сетей. Вместо того чтобы назначать идентификатор сети класса В, InterNIC выделяет организации какой-либо диапазон идентификаторов сетей класса С, который предоставляет достаточное для нее количество идентификаторов как сетей, так и хостов. Этот метод называется формированием надсетей (supernetting).

Например, вместо закрепления идентификатора сети класса В за организацией, в которой имеется до 2 000 хостов, InterNIC выделяет ей диапазон из восьми идентификаторов сетей класса С. Каждый такой идентификатор допускает наличие до 254 хостов, что в сумме дает 2 032 хоста.

Но, помогая экономить идентификаторы сетей класса В, этот метод создает новую проблему. Для пересылки IP-пакетов такой организации у Интернет-маршрутизаторов в таблицах маршрутизации должно быть по восемь записей с идентификаторами сетей класса С. Чтобы избежать переполнения таблиц маршрутизации, применяется метод CIDR (Classless Interdomain Routing); при этом набор записей с идентификаторами сетей сворачивается в одну запись, соответствующую всем выделенным организации идентификаторам сетей класса С.

С концептуальной точки зрения, CIDR создает в таблице маршрутизации запись вида **[начальный ID сети, счетчик]**, где начальный ID сети – первый идентификатор сети класса С, а счетчик – количество выделенных идентификаторов сетей класса С. На практике же применяется маска подсети, содержащая информацию о надсетях (supernetted subnet mask). Вот как она выглядит в ситуации, когда выделено 8 идентификаторов сетей класса С, начиная с 220.78.168.0:

- Начальный ID сети 220.78.168.0 11011100 01001110 10101000
00000000
- Конечный ID сети 220.78.175.0 11011100 01001110 10101111
00000000

Первые 21 бит во всех этих идентификаторах одинаковы. Последние три бита в третьем октете варьируются от 000 до 111. В итоге CIDR-запись в таблице маршрутизации выглядит так:

Идентификатор сети	Маска подсети	Маска подсети в двоичной форме
222.78.168.0	255.255.255.248.0	11111111 11111111 11111000 00000000

В нотации CIDR (то есть с указанием префикса сети) CIDR-запись представляет собой 220.78.168.0/21. Блок адресов, определяемых методом CIDR, называется CIDR-блоком.

Общие и частные адреса

Если интрасеть не подключена к Интернету, можно использовать любой вид IP-адресации. Но, если она должна быть прямо (через маршрутизатор) или косвенно (через прокси или транслятор) связана с Интернетом, приходится оперировать с адресами двух типов, применяемых в Интернете: общими (public) и частными (private).

Общие адреса назначаются InterNIC и состоят из идентификаторов сетей на основе класса или CIDR-блоков, глобально уникальных в Интернете. При выделении общих адресов маршруты программируются на маршрутизаторы Интернета, чтобы трафик, направляемый на эти адреса, мог быть доставлен по назначению. Этот трафик доступен в Интернете. Например, когда организации назначается CIDR-блок в виде идентификатора сети и маски подсети, то пара **[идентификатор сети, маска подсети]** записывается как маршрут в таблицы на маршрутизаторах Интернета. IP-пакеты, направляемые на один из адресов CIDR-блока, пересылаются требуемому адресату.

В частных сетях, которые не планируется соединять с Интернетом, можно выбирать любые адреса, даже общие, назначаемые комитетом InterNIC. Однако, если организация решит подключиться к Интернету, может получиться так, что схема адресации включает адреса, уже выделенные InterNIC другим организациям. Такие продублированные, или конфликтующие, адреса называются недопустимыми (illegal addresses). Интернет-соединения с этих адресов невозможны.

Каждому IP-узлу необходим IP-адрес, глобально уникальный в межсетевой IP-среде, а в случае Интернета – IP-адрес, глобально

уникальный в Интернете. Любой организации, подключающейся к Интернету, требовался общий адрес для каждого узла в ее интрасети. Из-за этого возникла нехватка общих адресов.

Анализируя потребности организаций в адресах, проектировщики Интернета заметили, что в интрасетях многих из них большинство хостов не требует прямого соединения с Интернет-хостами. А те хосты, которым действительно необходим специфический набор Интернет-услуг, например доступ в World Wide Web и электронная почта, обычно обращаются за такими услугами через шлюзы прикладного уровня – прокси-серверы, серверы электронной почты и т. д. В итоге основной массе организаций нужны лишь небольшие диапазоны общих адресов для узлов, напрямую подключенных к Интернету, – прокси-серверов, маршрутизаторов, брандмауэров (firewalls) и трансляторов.

Хостам, которым не требуется прямой доступ в Интернет, должны быть назначены IP-адреса, не дублирующие уже выделенные общие адреса. Чтобы решить эту проблему, проектировщики Интернета зарезервировали часть адресного пространства и назвали ее частным адресным пространством (private address space). IP-адрес в таком пространстве никогда не выделяется как общий. IP-адреса в частном адресном пространстве называются частными адресами (private addresses). Поскольку общее и частное адресные пространства не перекрываются, частные адреса никогда не дублируют общие.

Частное адресное пространство, описанное в RFC 1918, определяется следующими тремя адресными блоками:

- 10.0.0.0/8 – идентификатор сети класса А, допускающий IP-адреса в диапазоне от 10.0.0.1 до 10.255.255.254 и предусматривающий 24 бита для идентификатора хоста, которые могут быть использованы в любой схеме разбиения на подсети в рамках частной организации;
- 172.16.0.0/12 – интерпретируется либо как блок из 16 идентификаторов сетей класса В, либо как 20-битное частное адресное пространство, которое может быть использовано в любой схеме разбиения на подсети в рамках частной организации. Частная сеть 172.16.0.0/12 допускает IP-адреса в диапазоне от 172.16.0 до 172.31.255.254;

- 192.168.0.0/16 – интерпретируется либо как блок из 256 идентификаторов сетей класса С, либо как 16-битное частное адресное пространство, которое может быть использовано в любой схеме разбиения на подсети в рамках частной организации. Частная сеть 192.168.0.0/16 допускает IP-адреса в диапазоне от 192.168.0.1 до 192.168.255.254.

Таким образом, многие организации используют одно и то же частное адресное пространство, что помогает избежать нехватки общих адресов.

Поскольку IP-адреса в частном адресном пространстве никогда не выделяются InterNIC в качестве общих, на Интернет-маршрутизаторах нет маршрутов к этим частным адресам, и они недоступны в Интернете. Следовательно, Интернет-трафик от хоста с частным адресом должен поступать либо шлюзу прикладного уровня (например, прокси-серверу) с корректным общим адресом, либо транслятору сетевых адресов (network address translator, NAT), который перед отправкой трафика в Интернет преобразует частный адрес хоста в допустимый общий адрес.

Команда тестирования настроек сетевого интерфейса

Для отображения настроек сетевого интерфейса в операционных системах семейства Windows используется команда **ipconfig**. Ее полный синтаксис следующий:

```
ipconfig [/all] [/renew [адаптер]] [/release [адаптер]] [/flushdns] [/displaydns]
[/registerdns] [/showclassid адаптер] [/setclassid адаптер [код_класса]]
```

Параметры:

/all – вывод полной конфигурации TCP/IP для всех адаптеров. Без этого параметра команда **ipconfig** выводит только IP-адреса, маску подсети и основной шлюз для каждого адаптера. Адаптеры могут представлять собой физические интерфейсы, такие как установленные сетевые адаптеры, или логические интерфейсы, такие, как подключения удаленного доступа.

/renew [адаптер] – обновление конфигурации DHCP для всех адаптеров (если адаптер не задан) или для заданного адаптера. Данный параметр доступен только на компьютерах с адаптерами, настроенными для автоматического получения IP-адресов. Чтобы

указать адаптер, введите без параметров имя, выводимое командой **ipconfig**.

/release [адаптер] – отправка сообщения **DHCPRELEASE** серверу DHCP для освобождения текущей конфигурации DHCP и удаление конфигурации IP-адресов для всех адаптеров (если адаптер не задан) или для заданного адаптера. Этот параметр отключает протокол TCP/IP для адаптеров, настроенных для автоматического получения IP-адресов. Чтобы указать адаптер, введите без параметров имя, выводимое командой **ipconfig**.

/flushdns – сброс и очистка содержимого кэша сопоставления имен DNS клиента. Во время устранения неполадок DNS эту процедуру используют для удаления из кэша записей отрицательных попыток сопоставления и других динамически добавляемых записей.

/displaydns – отображение содержимого кэша сопоставления имен DNS клиента, включающего записи, предварительно загруженные из локального файла **Hosts**, а также последние полученные записи ресурсов для запросов на сопоставление имен. Эта информация используется службой DNS клиента для быстрого сопоставления часто встречаемых имен без обращения к указанным в конфигурации DNS-серверам.

/registerdns – динамическая регистрация вручную имен DNS и IP-адресов, настроенных на компьютере. Этот параметр полезен при устранении неполадок в случае отказа в регистрации имени DNS или при выяснении причин неполадок динамического обновления между клиентом и DNS-сервером без перезагрузки клиента. Имена, зарегистрированные в DNS, определяются параметрами DNS в дополнительных свойствах протокола TCP/IP.

/showclassid адаптер – отображение кода класса DHCP для указанного адаптера. Чтобы просмотреть код класса DHCP для всех адаптеров, вместо параметра **адаптер** укажите звездочку (*). Данный параметр доступен только на компьютерах с адаптерами, настроенными для автоматического получения IP-адресов.

/setclassid адаптер [код_класса] – задание кода класса DHCP для указанного адаптера. Чтобы задать код класса DHCP для всех адаптеров, вместо параметра **адаптер** укажите звездочку (*). Данный параметр доступен только на компьютерах с адаптерами,

настроенными для автоматического получения IP-адресов. Если код класса DHCP не задан, текущий код класса удаляется.

Для проверки настроек TCP/IP на компьютере CLIENT в командной строке необходимо выполнить команду **ipconfig**. Результаты ее работы представлены на рисунке 35.

Если выдаваемые значения IP-адреса и маски подсети совпадают с требуемыми, то это означает, что TCP/IP на этом компьютере сконфигурирован правильно. Для проверки возможности установления соединения с другим компьютером, на котором настроен стек протоколов TCP/IP, используется утилита командной строки **ping**. Эта утилита посылает эхо-запросы по протоколу ICMP. После каждой передачи выводится соответствующее сообщение с эхо-ответом.

Команда тестирования работоспособности сетевого соединения между двумя узлами *ping*

Для тестирования работоспособности сетевого соединения между двумя узлами используется команда **ping**. Ее синтаксис следующий:

```
ping [-t] [-a] [-n счетчик] [-l размер] [-f] [-i TTL] [-v тип] [-r счетчик] [-s  
счетчик] [{-j список_узлов | -k список_узлов}] [-w интервал]  
[имя_конечного_компьютера]
```

Параметры:

-t – задает для команды **ping** отправку сообщений с эхо-запросом к точке назначения до тех пор, пока команда не будет прервана. Для прерывания команды и вывода статистики нажмите комбинацию **CTRL-BREAK**. Для прерывания команды **ping** и выхода из нее нажмите клавиши **CTRL-C**.

-a – задает разрешение обратного имени по IP-адресу назначения. В случае успешного выполнения выводится имя соответствующего узла.

-n счетчик – задает число отправляемых сообщений с эхо-запросом. По умолчанию – 4.

-l размер – задает длину (в байтах) поля данных в отправленных сообщениях с эхо-запросом. По умолчанию – 32 байта. Максимальный размер – 65 527.

-f – задает отправку сообщений с эхо-запросом с флагом «Don't Fragment» в IP-заголовке, установленном на 1. Сообщения с эхо-запросом не фрагментируются маршрутизаторами на пути к месту назначения. Этот параметр полезен для устранения проблем, возникающих с максимальным блоком данных для канала (Maximum Transmission Unit).

-i TTL – задает значение поля TTL в IP-заголовке для отправляемых сообщений с эхо-запросом. По умолчанию берется значение TTL, заданное по умолчанию для узла. Для узлов Windows XP это значение обычно равно 128. Максимальное значение TTL – 255.

-v тип – задает значение поля типа службы (TOS) в IP-заголовке для отправляемых сообщений с эхо-запросом. По умолчанию это значение равно 0. Тип – это десятичное значение от 0 до 255.

-r счетчик – задает параметр записи маршрута (Record Route) в IP-заголовке для записи пути, по которому проходит сообщение с эхо-запросом и соответствующее ему сообщение с эхо-ответом. Каждый переход в пути использует параметр записи маршрута. По возможности значение счетчика задается равным или большим, чем количество переходов между источником и местом назначения. Параметр счетчик имеет значение от 1 до 9.

-s счетчик – указывает вариант штампа времени Интернета (Internet Timestamp) в заголовке IP для записи времени прибытия сообщения с эхо-запросом и соответствующего ему сообщения с эхо-ответом для каждого перехода. Параметр счетчик имеет значение от 1 до 4.

-j список_узлов – указывает для сообщений с эхо-запросом использование параметра свободной маршрутизации в IP-заголовке с набором промежуточных точек назначения, указанным в списке_узлов. При свободной маршрутизации последовательные промежуточные точки назначения могут быть разделены одним или несколькими маршрутизаторами. Максимальное число адресов или имен в списке узлов – 9.

Список_узлов – это набор IP-адресов (в точечно-десятичной нотации), разделенных пробелами.

-k список_узлов – указывает для сообщений с эхо-запросом использование параметра строгой маршрутизации в IP-заголовке

с набором промежуточных точек назначения, указанным в списке_узлов. При строгой маршрутизации следующая промежуточная точка назначения должна быть доступной напрямую (она должна быть соседней в интерфейсе маршрутизатора). Максимальное число адресов или имен в списке узлов равно 9.

Список_узлов – это набор IP-адресов (в точечно-десятичной нотации), разделенных пробелами.

-w интервал – определяет в миллисекундах время ожидания получения сообщения с эхо-ответом, которое соответствует сообщению с эхо-запросом. Если сообщение с эхо-ответом не получено в пределах заданного интервала, то выдается сообщение об ошибке **Request timed out**. Интервал по умолчанию равен 4 000 (4 секунды).

Имя_конечного_компьютера – задает точку назначения, идентифицированную IP-адресом или именем узла.

Практические задания

Постановка задачи

В данной лабораторной работе с использованием специального программного обеспечения – эмулятора – будут созданы 2 виртуальные машины, в которых будут установлены операционные системы Microsoft Windows 2008 Server и Microsoft Windows XP Professional, а также настроены их сетевые подсистемы в части конфигурирования стека протоколов TCP/IP.

Создание новой виртуальной машины

В любом эмуляторе, например Oracle VM VirtualBox, необходимо создать новую виртуальную машину. Для этого требуется установить эмулятор (рис. 2), а затем создать новую виртуальную машину нажав на кнопку **Создать** (рис. 3).

На первом шаге работы мастера создания виртуальной машины следует выбрать тип и версию создаваемой виртуальной машины (Microsoft Windows, Windows 2008), а также задать имя для виртуальной машины (рис. 4).

Далее следует указать выделяемый виртуальной машине объем оперативной памяти (рис. 5).



Рис. 2. Установка эмулятора

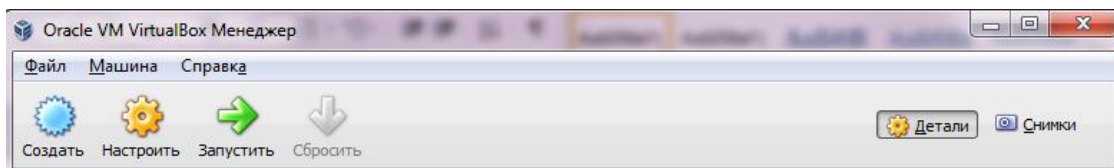


Рис. 3. Панель инструментов эмулятора Oracle VM VirtualBox

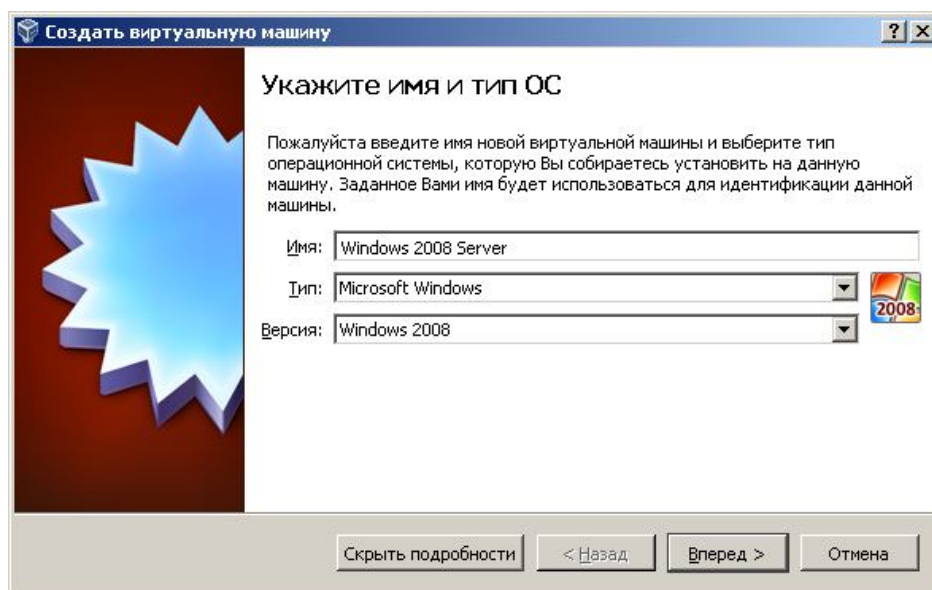


Рис. 4. Указание имени и типа ОС

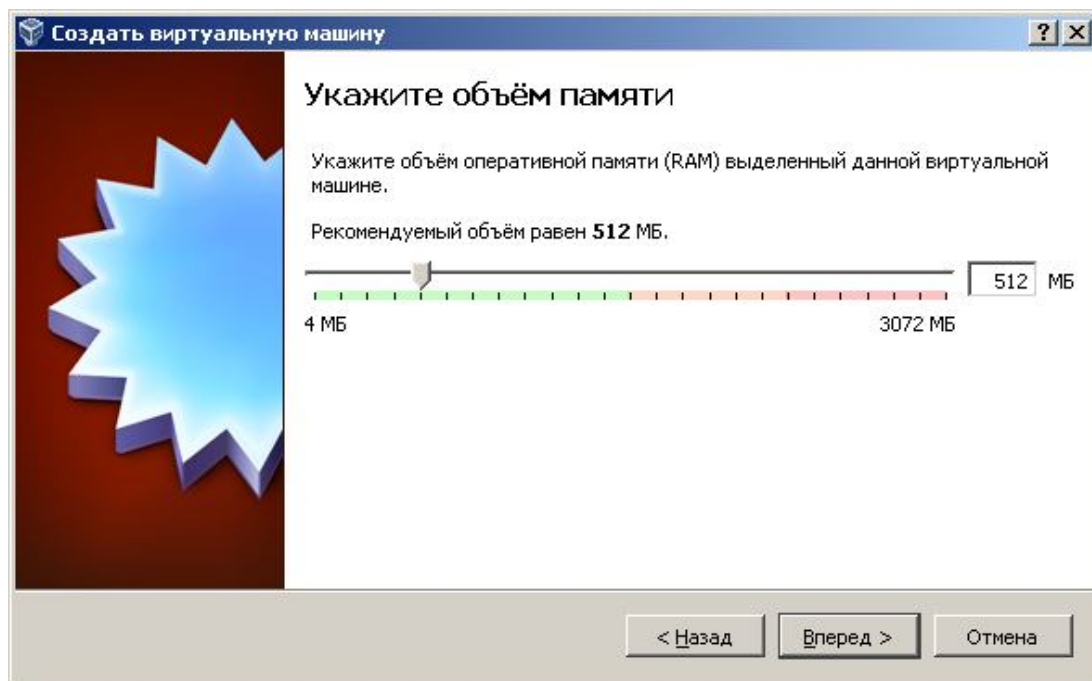


Рис. 5. Задание выделяемого виртуальной машине объема оперативной памяти

На следующем этапе необходимо создать новый виртуальный жесткий диск (рис. 6) и выбрать его тип (рис. 7).

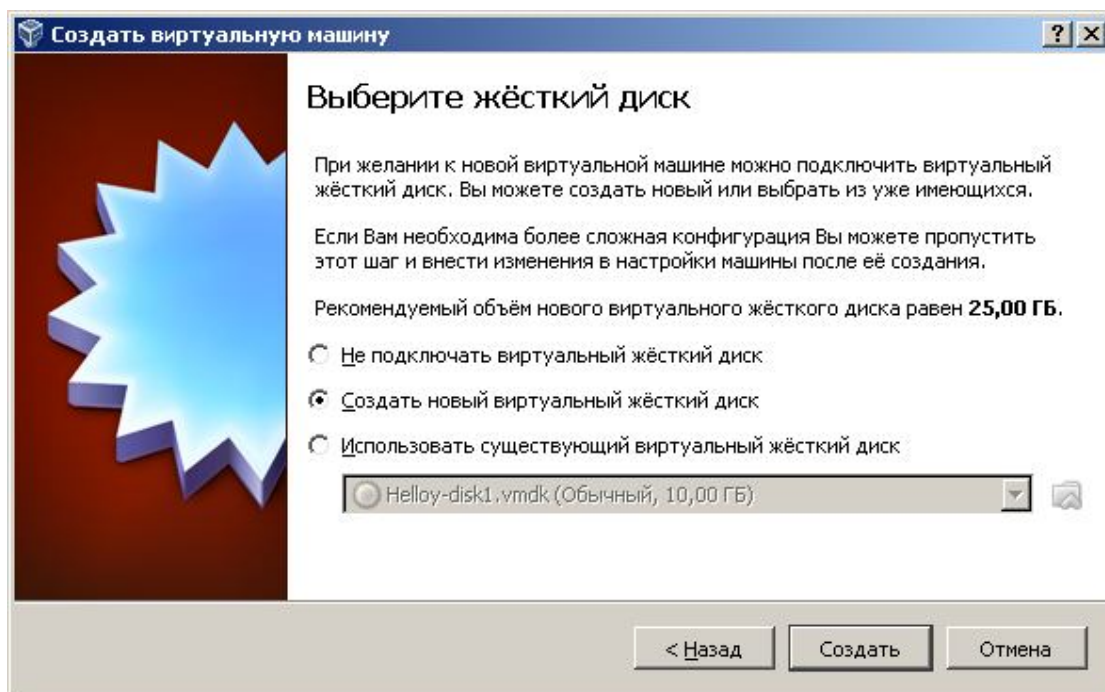


Рис. 6. Создание нового виртуального жесткого диска

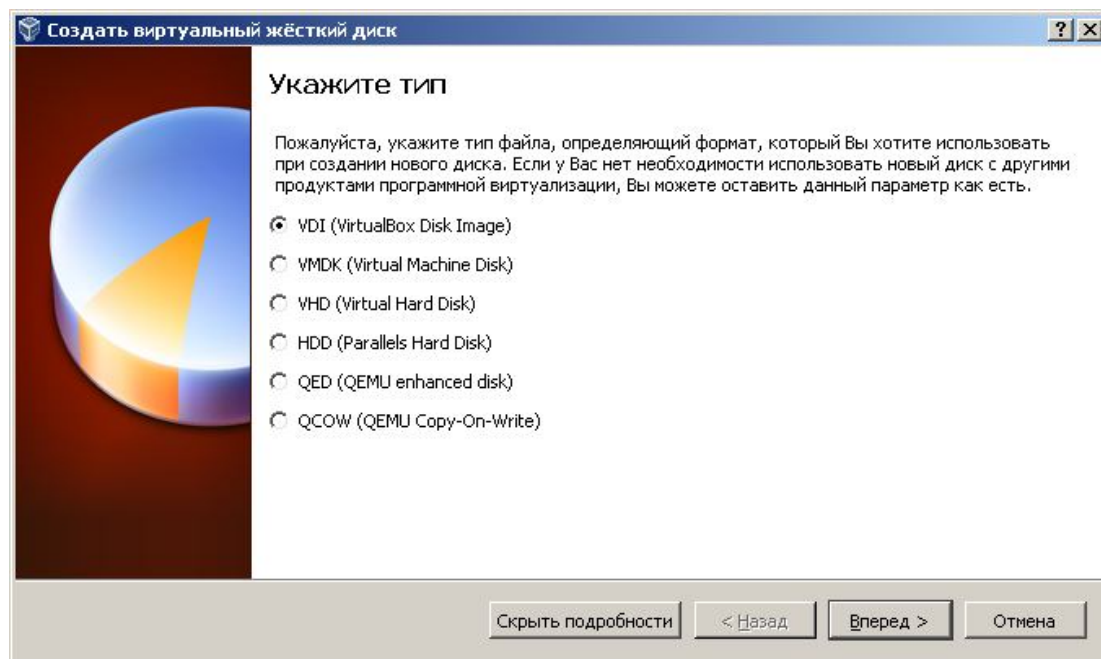


Рис. 7. Выбор типа создаваемого виртуального жесткого диска

Далее необходимо выбрать формат хранения на создаваемом виртуальном жестком диске (рис. 8). Для большинства случаев, в том числе и данной лабораторной работы, оптимальным решением будет выбор динамического виртуального жесткого диска.

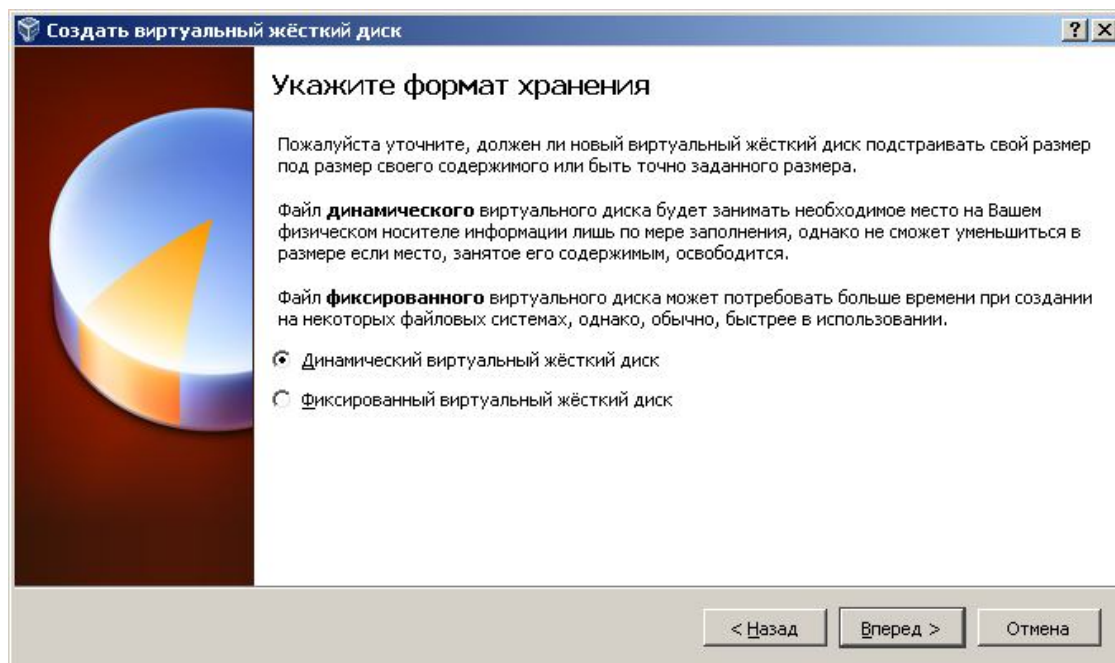


Рис. 8. Выбор формата хранения

На следующем этапе необходимо выбрать имя файла на физическом компьютере и объем создаваемого виртуального жесткого диска (рис. 9).

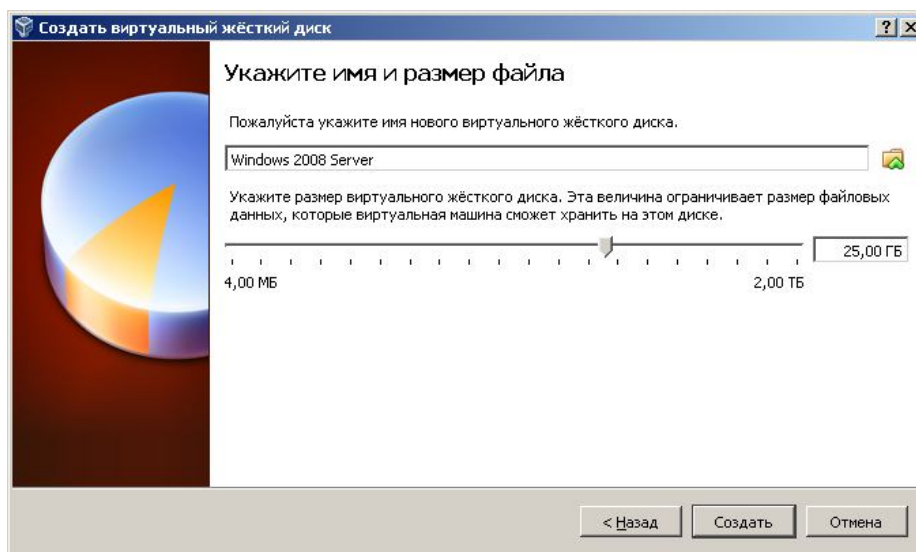


Рис. 9. Выбор имени файла виртуального жесткого диска и его максимального размера

В результате будет создана новая виртуальная машина, в которой будет производиться установка операционной системы Windows 2008 Server (рис. 10).

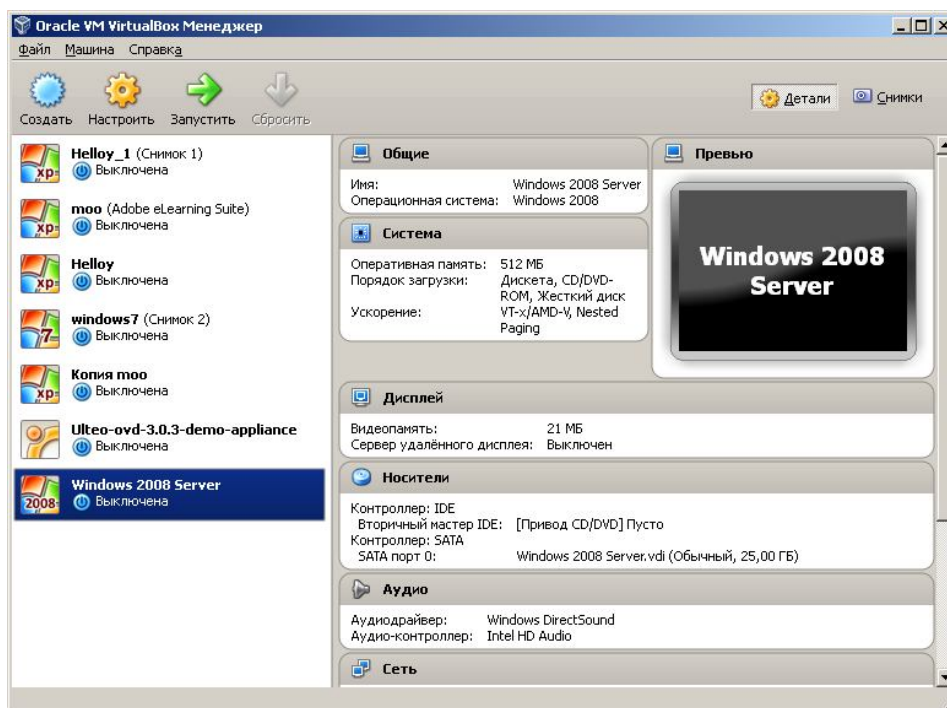



Рис. 10. Созданная виртуальная машина

Установка Windows 2008 Server

Далее в созданной виртуальной машине необходимо произвести установку операционной системы Windows 2008 Server. Для этого следует включить виртуальную машину, для чего необходимо нажать на кнопку **Запустить** на панели инструментов программы VM VirtualBox (рис. 11).

Далее необходимо выбрать с помощью кнопки  путь к образу устанавливаемой операционной системы (рис. 11), а затем нажать кнопку **Продолжить**.

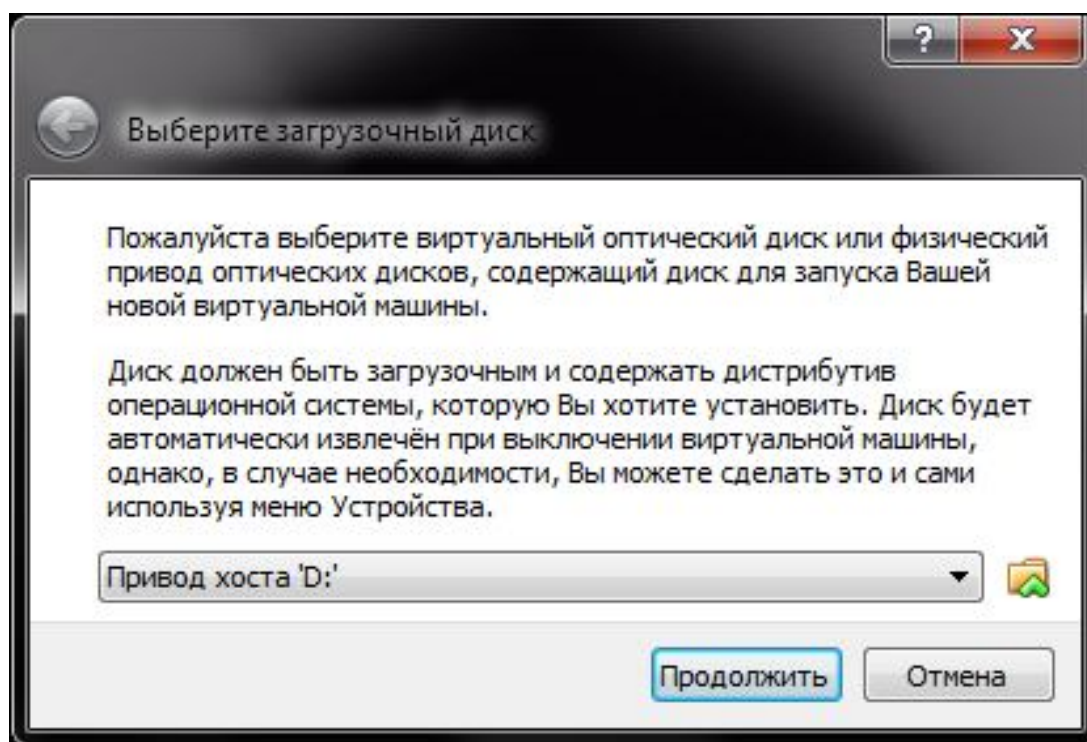


Рис. 11. Выбор загрузочного диска

Произойдет загрузка программы установки. В процессе установки вначале предлагается выбрать язык операционной системы и язык ввода (для удобства рекомендуется в раскрывающемся списке **Раскладка клавиатуры** выбрать значение – **США**) (рис. 12).

Для начала установки следует нажать кнопку **Установить** (рис. 13).

На следующем этапе работы программы установки следует выбрать версию Windows Server 2008 (рис. 14). Следует выбрать вариант **Windows Server 2008 Standard (полная установка)** и нажать кнопку **Далее**.



Рис. 12. Выбор раскладки клавиатуры



Рис. 13. Начало установки операционной системы

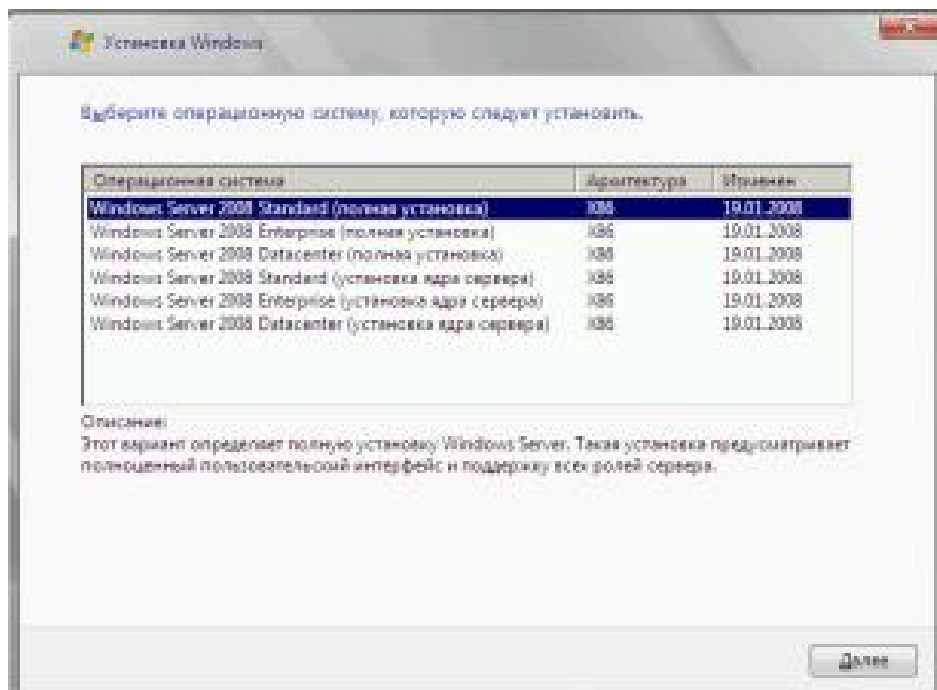


Рис. 14. Выбор версии устанавливаемой операционной системы

Далее на экран выводится информация о лицензионном соглашении. Для его принятия следует поставить галочку и нажать кнопку **Далее** (рис. 15).

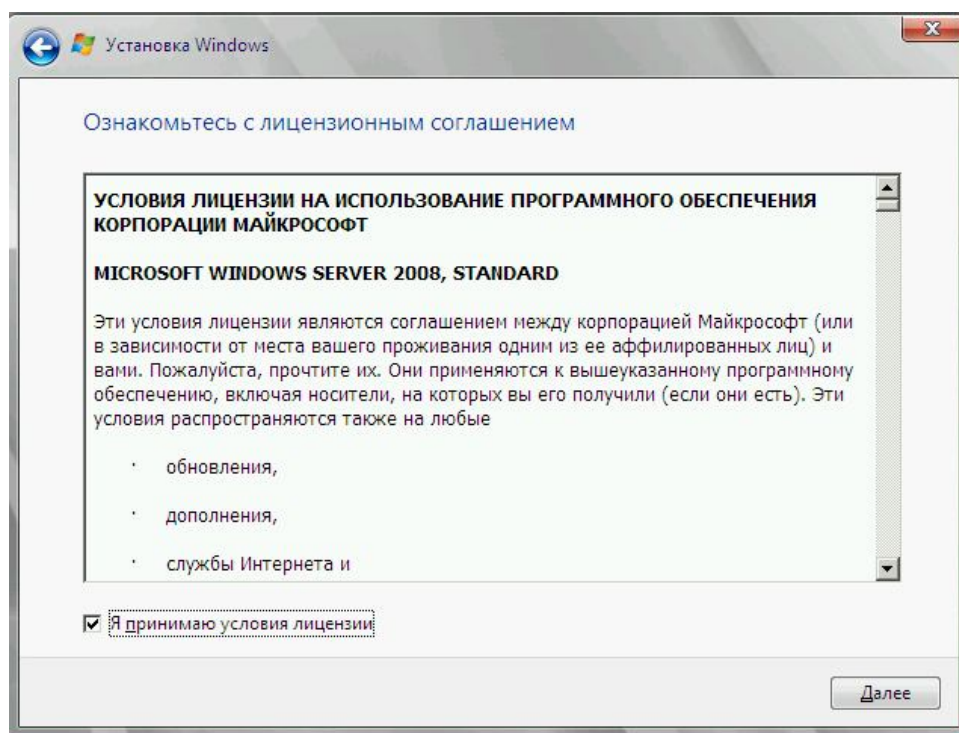


Рис. 15. Условия лицензионного соглашения

Следующий шаг работы программы установки предназначен для выбора типа установки (обновление или полная установка). В данном случае необходимо выбрать **Полная установка** (рис. 16).

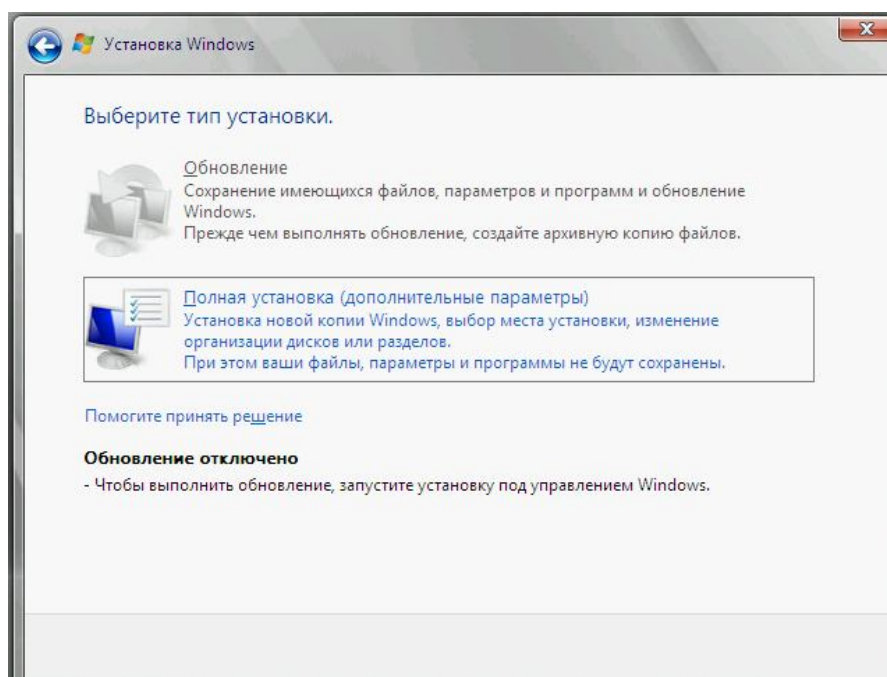


Рис. 16. Выбор типа установки

В следующем окне требуется указать размер создаваемого раздела. Необходимо оставить значение, предлагаемое по умолчанию, и нажать кнопку **Далее** (рис. 17).

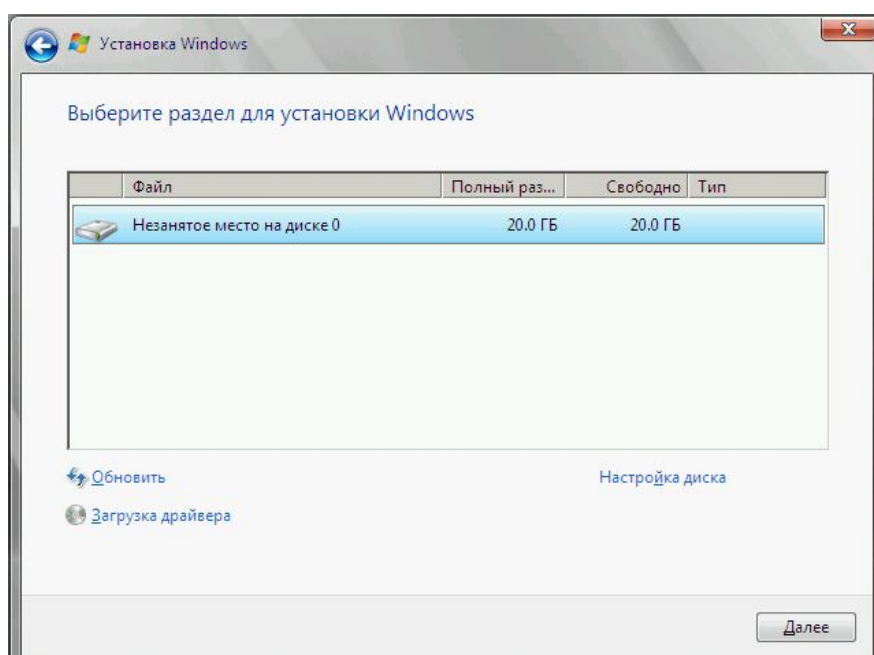


Рис. 17. Выбор раздела для установки

Далее начнется процедура копирования файлов операционной системы (рис. 18).

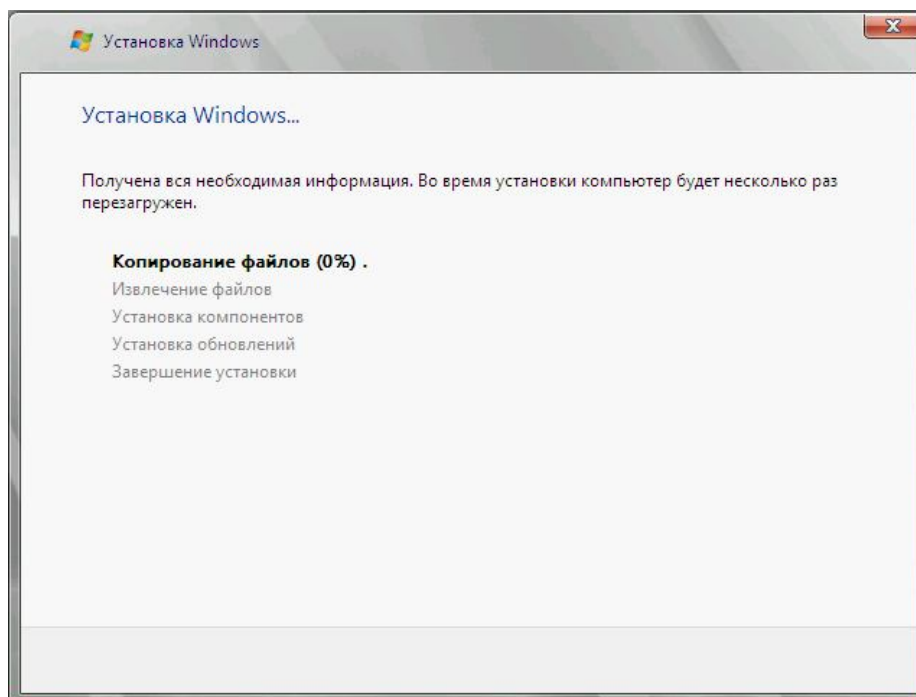


Рис. 18. Установка Windows

По ее завершении появится окно (рис. 19), которое уведомляет о необходимости перезагрузки компьютера.

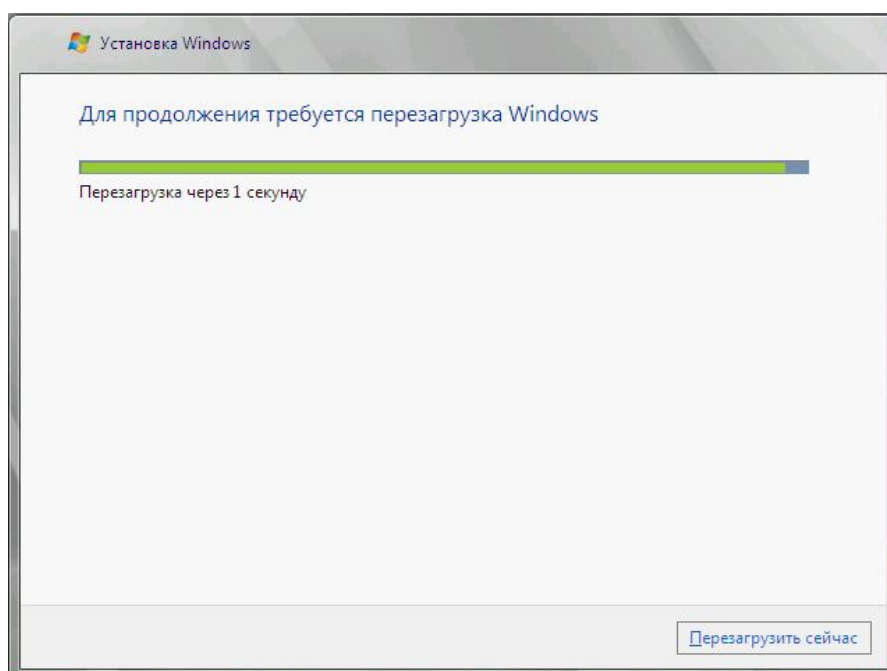


Рис. 19. Требование перезагрузки

После перезагрузки появится предупреждение о необходимости смены пароля перед первым входом в систему (рис. 20). Следует сменить пароль на содержащий как буквы, так и цифры (рис. 21), а затем записать пароль администратора в лабораторную тетрадь.

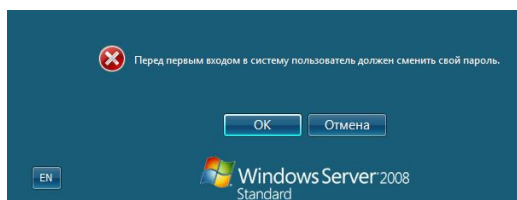


Рис. 20. Предупреждение о смене пароля

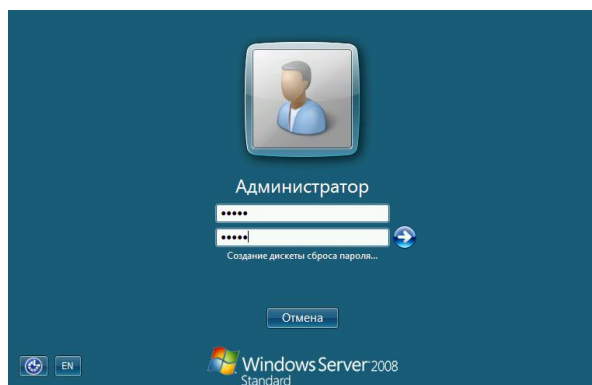


Рис. 21. Ввод нового пароля

Далее появится уведомление о смене пароля (рис. 22).

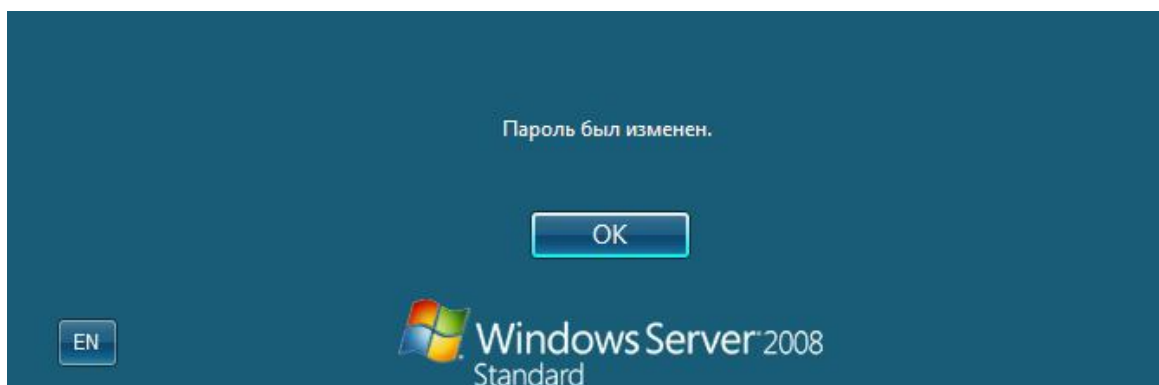


Рис. 22. Подтверждение смены пароля

После входа в систему автоматически откроется окно **Задачи начальной настройки** (рис. 23).

На этом процесс установки Windows 2008 Server завершен.

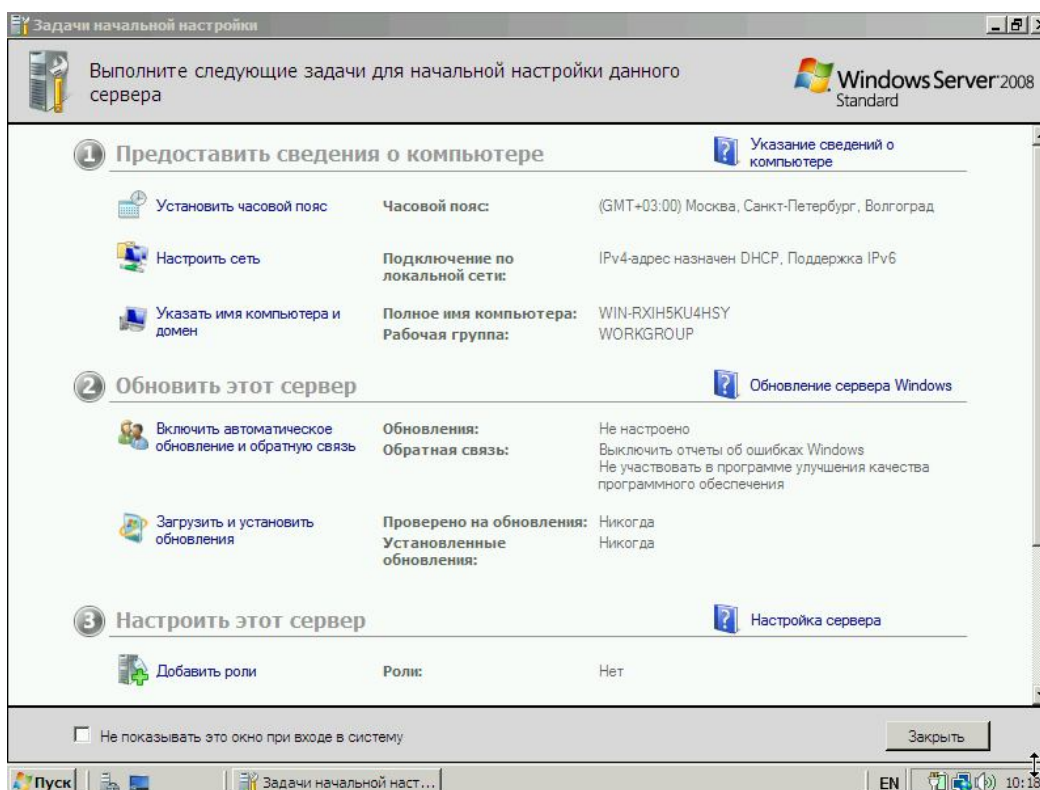


Рис. 23. Задачи начальной настройки

Установка Windows XP Professional

Далее по аналогии необходимо создать вторую виртуальную машину, выбрав в качестве операционной системы Windows XP, и запустить ее, а затем указать путь к файлу образа устанавливаемой операционной системы. Для того чтобы приступить к установке, на начальном этапе работы программы установки следует нажать **Enter** (рис. 24).

Следующий этап установки предназначен для выбора типа файловой системы, которая будет использоваться в создаваемом разделе жесткого диска. Необходимо выбрать вариант **Форматировать раздел в системе NTFS** (рис. 25).

На следующем этапе установки необходимо задать имя компьютера (CLIENT) и пароль для учетной записи администратора. Следует записать в лабораторную тетрадь имя компьютера и пароль администратора. При выборе членства компьютера в рабочей группе или домене (рис. 26) необходимо выбрать вариант с членством в рабочей группе, название которой следует оставить предложенным по умолчанию – WORKGROUP.

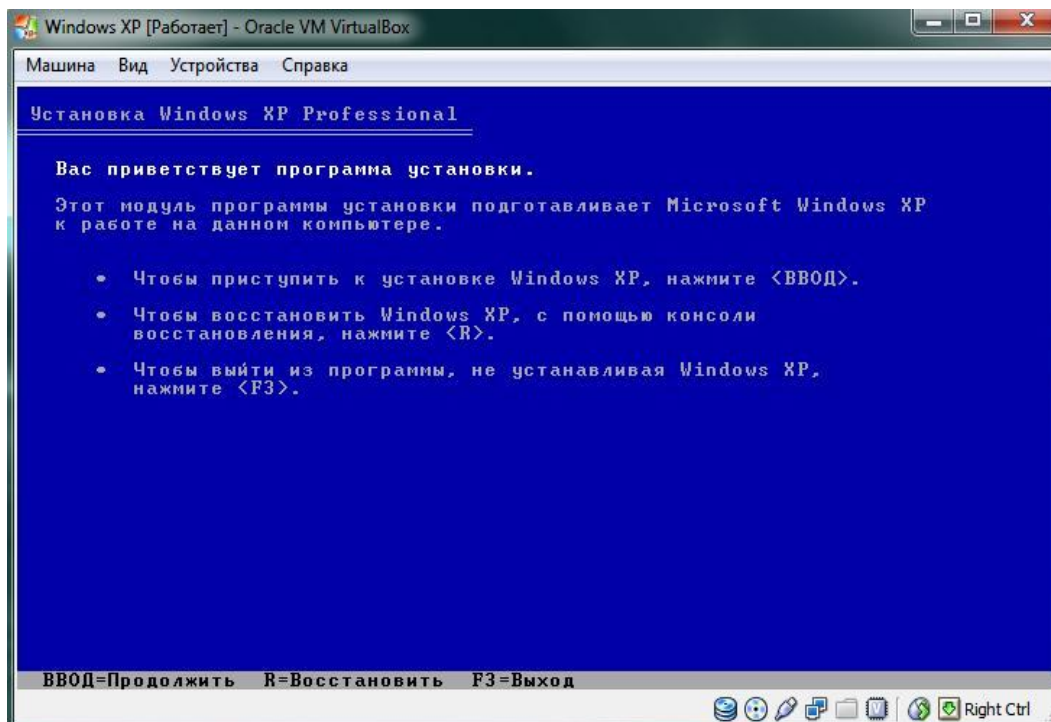


Рис. 24. Начало установки Windows XP

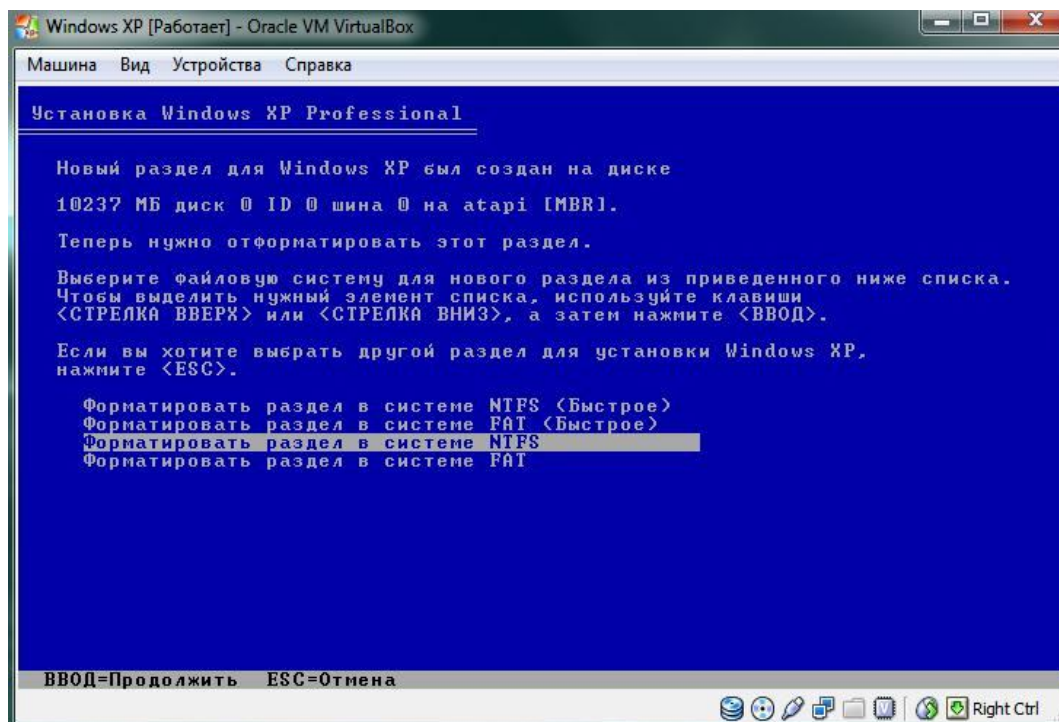


Рис. 25. Форматирование раздела

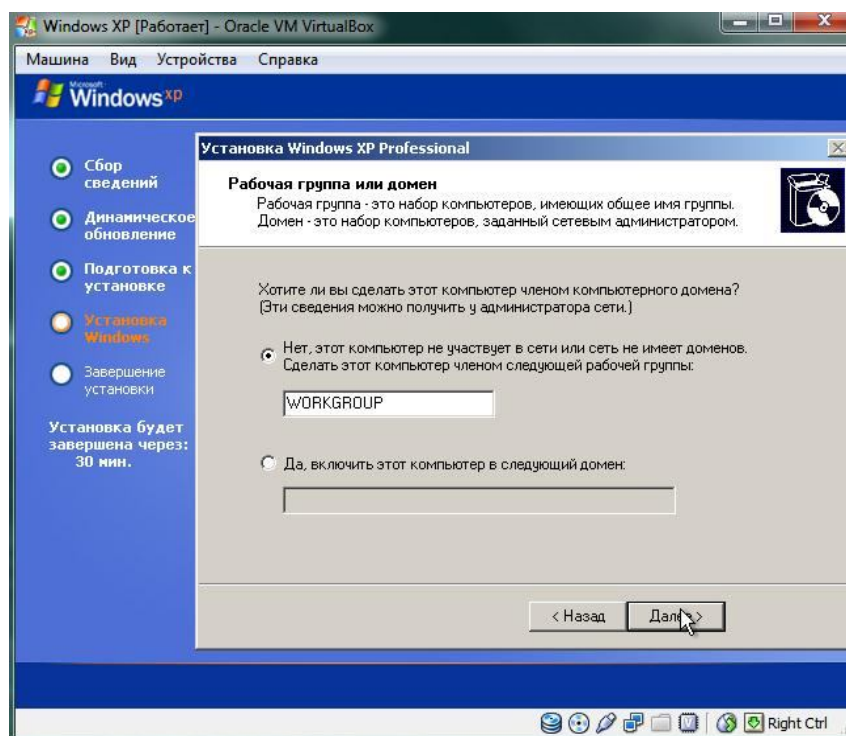


Рис. 26. Выбор членства компьютера в рабочей группе или домене

Далее установка Windows XP проходит в автоматическом режиме. По окончании процесса будет загружен рабочий стол Windows XP (рис. 27).

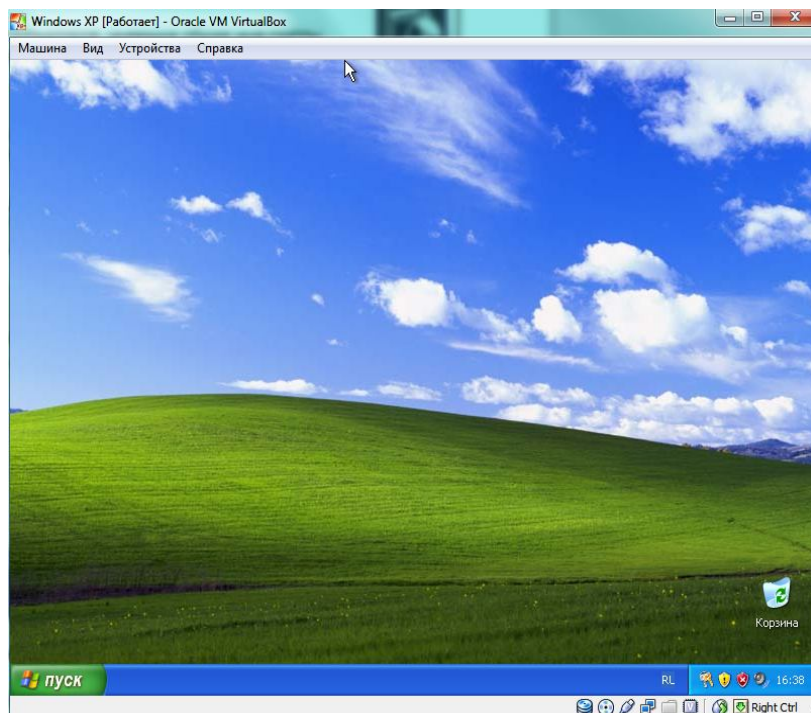


Рис. 27. Рабочий стол Windows XP

Настройка статического IP-адреса сетевого интерфейса виртуальной машины под управлением Windows XP Professional

Далее следует настроить стек протоколов TCP/IP в Windows XP, для чего вначале необходимо открыть окно **Сетевые подключения** (**Пуск | Панель управления | Сетевые подключения**, рис. 28).

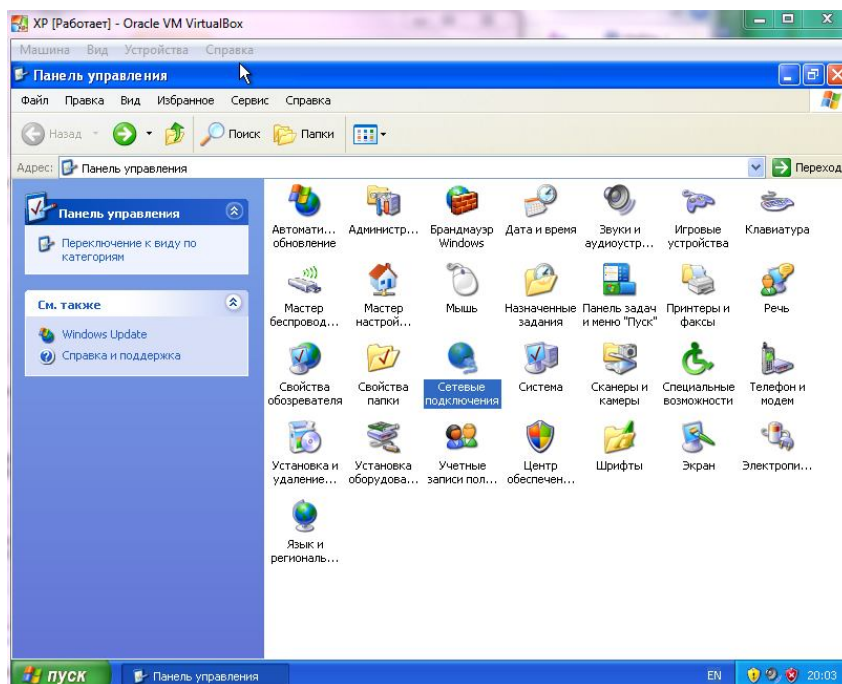


Рис. 28. Панель управления Windows XP

Далее, щелкнув правой кнопкой мыши по **Подключение по локальной сети**, в контекстном меню следует выбрать пункт **Свойства** (рис. 29).

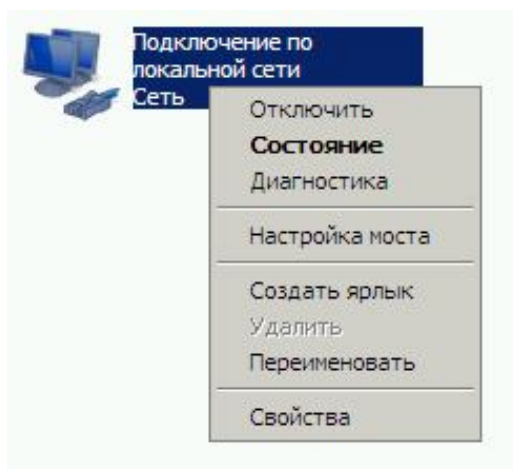


Рис. 29. Контекстное меню объекта **Подключение по локальной сети**

В открывшемся окне свойств необходимо выделить пункт **Протокол Интернета (TCP/IP)** и нажать кнопку **Свойства** (рис. 30).

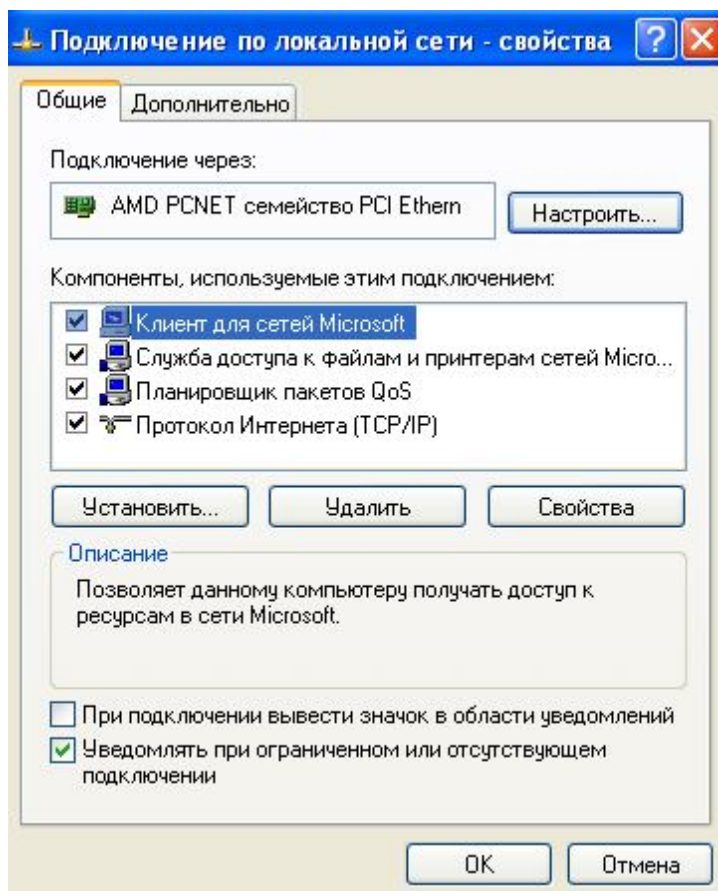


Рис. 30. Диалоговое окно **Подключение по локальной сети – свойства**

В открывшемся окне необходимо выбрать способ задания IP-адреса. Возможны два варианта: получение IP-адреса автоматически (с использованием службы DHCP) и ручное назначение IP-адреса. Следует выбрать второй вариант и указать следующие параметры (рис. 31):

- IP-адрес: 192.168.1.2;
- Маска подсети: 255.255.255.0;
- Основной шлюз: 192.168.1.1;
- Предпочитаемый DNS-сервер: 192.168.1.1;

IP-адрес **192.168.1.1/24** позднее будет присвоен сетевому интерфейсу виртуальной машины под управлением операционной системы Windows 2008 Server. Затем необходимо нажать кнопку **ОК** и

закреть диалоговое окно **Подключение по локальной сети – свойства**.

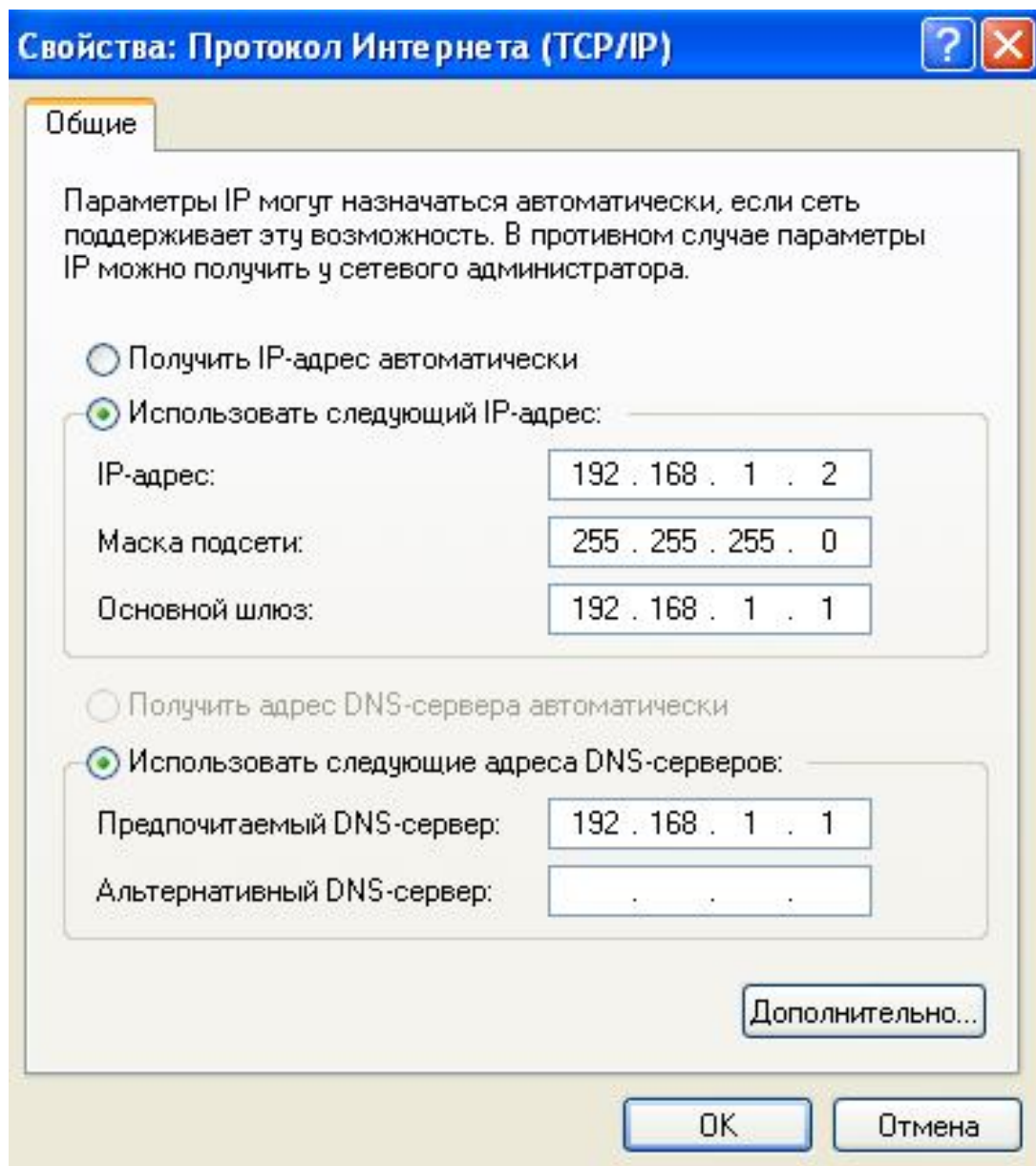


Рис. 31. Диалоговое окно **Свойства: Протокол Интернета**

Отключение брандмауэра в Windows XP

Далее следует отключить брандмауэр (межсетевой экран, который блокирует часть сетевого трафика между компьютерами). Для этого нужно зайти в **Панель управления** (рис. 28), выбрать **Брандмауэр Windows** и в открывшемся окне выбрать вариант **Выключить** (рис. 32).

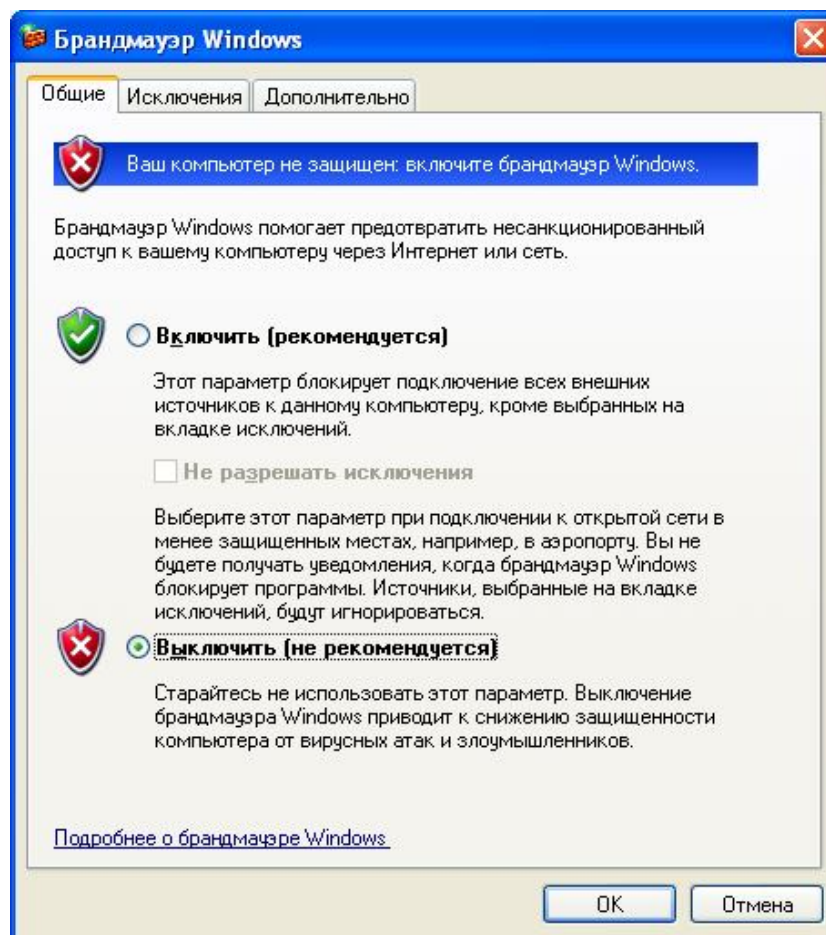


Рис. 32. Отключение брандмауэра

Выбор типа сетевого подключения виртуальной машины

Далее необходимо убедиться, что в настройках виртуальной машины XP задан тип сетевого подключения **Внутренняя сеть**. Для этого на панели управления эмулятора при включенной виртуальной машине необходимо нажать кнопку **Настроить** (рис. 33).

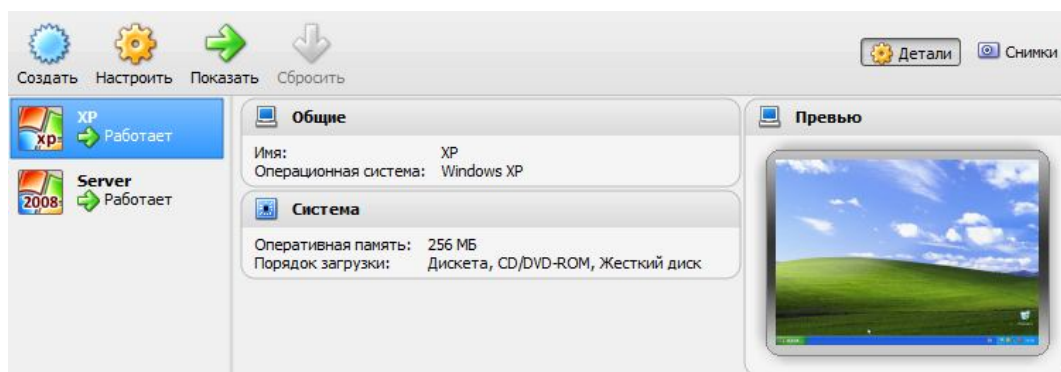


Рис. 33. Окно управления виртуальными машинами эмулятора VirtualBox

В открывшемся окне настроек виртуальной машины на левой панели необходимо выбрать пункт **Сеть**, а затем в раскрывающемся списке **Тип подключения** выбрать вариант **Внутренняя сеть** (рис. 34).

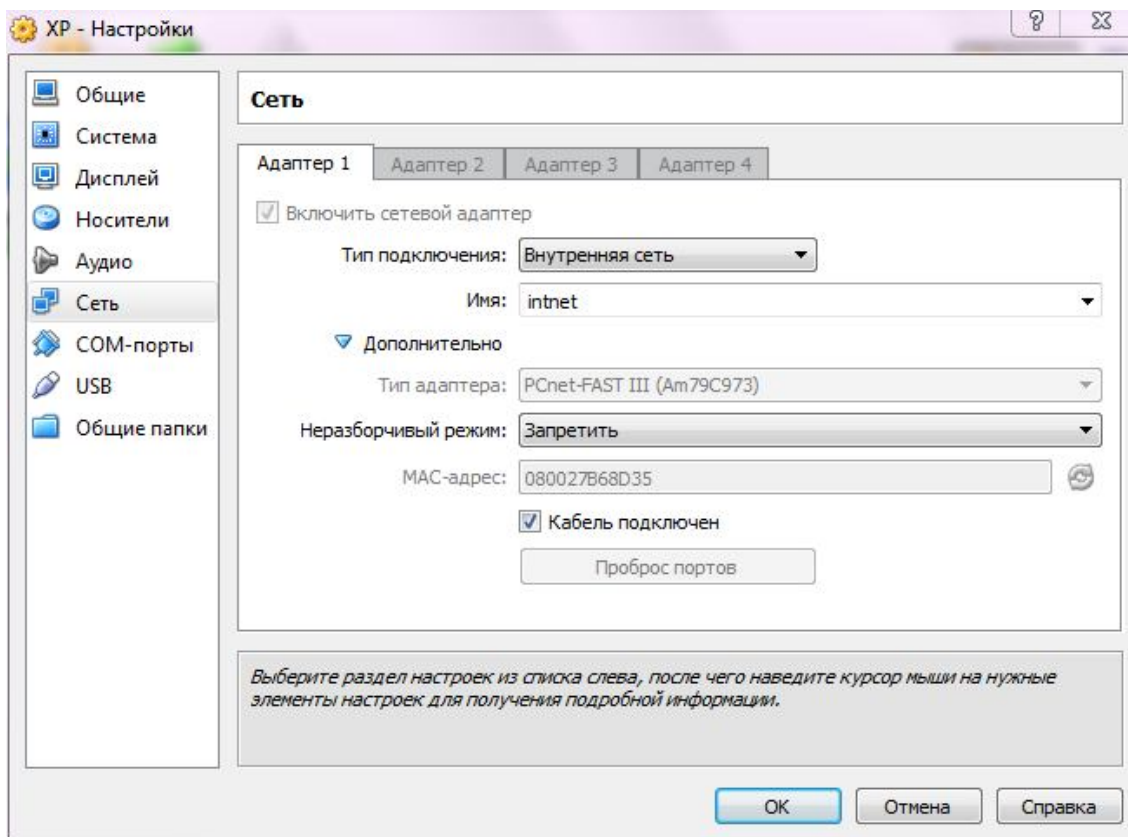


Рис. 34. Задание типа сетевого подключения виртуальной машины

Аналогичную операцию следует проделать с виртуальной машиной под управлением операционной системы Windows 2008 Server.

Тестирование параметров сетевого интерфейса виртуальной машины под управлением Windows XP

Чтобы проверить, применились ли заданные настройки сетевого интерфейса виртуальной машины под управлением операционной системы Windows XP, необходимо в меню **Пуск виртуальной машины** выбрать элемент **Командная строка** и выполнить в командной строке команду **ipconfig** (рис. 35).

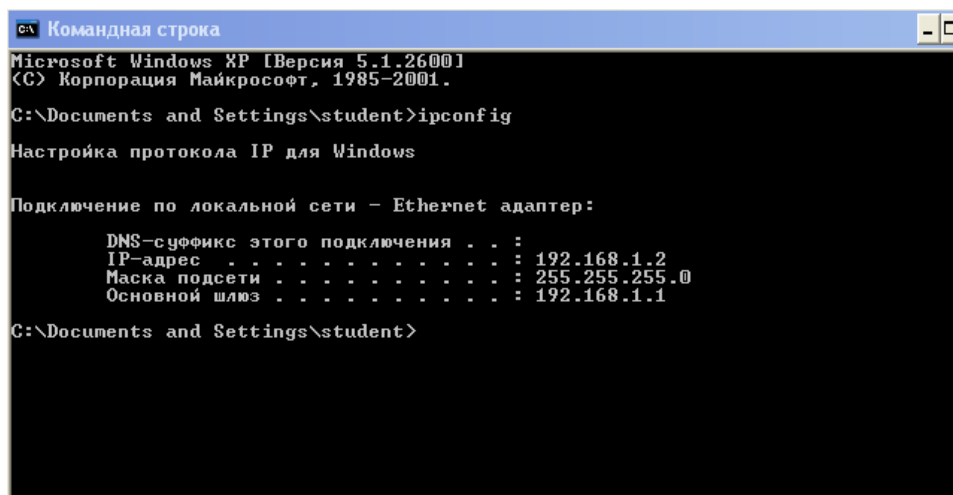


Рис. 35. Результаты работы утилиты **ipconfig**

Настройка статического IP-адреса сетевого интерфейса виртуальной машины под управлением Windows 2008 Server

Аналогичным образом следует настроить статический IP-адрес и для сетевого интерфейса виртуальной машины под управлением операционной системы Windows 2008 Server. Для этих целей в Windows 2008 Server используется **Центр управления сетями и общим доступом** (Пуск | Панель управления | Сеть и Интернет | Центр управления сетями и общим доступом) (рис. 36).

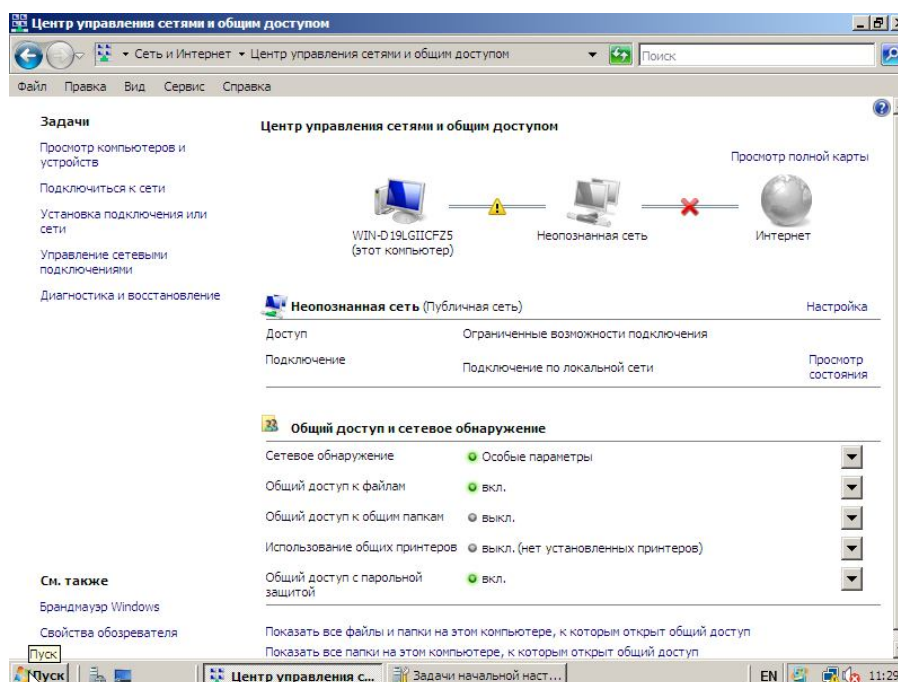


Рис. 36. Диалоговое окно **Центр управления сетями и общим доступом**

В окне Центр управления сетями и общим доступом, слева в меню Задачи, необходимо выбрать ссылку Управление сетевыми подключениями.

Далее необходимо нажать правой кнопкой мыши на значке **Подключение по локальной сети** и в контекстном меню выбрать пункт **Свойства**.

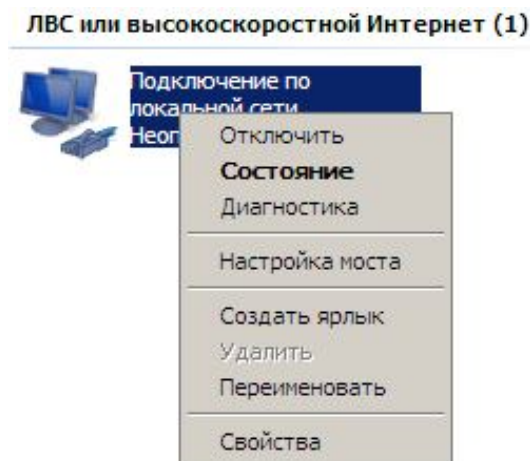


Рис. 37. Контекстное меню **Подключение по локальной сети**

В окне свойств подключения следует убрать галочку напротив пункта **Протокол Интернета версии 6 (TCP/IPv6)**, выделить пункт **Протокол Интернета версии 4 (TCP/IPv4)** и нажать кнопку **Свойства** (рис. 38).

В открывшемся окне необходимо выбрать способ задания IP-адреса. Возможны два варианта: получение IP-адреса автоматически (при помощи службы DHCP) и ручное назначение IP-адреса. Следует выбрать второй вариант и указать следующие параметры (рис. 39):

- IP-адрес: 192.168.1.1;
 - Маска подсети: 255.255.255.0;
 - Предпочитаемый DNS-сервер: 192.168.1.1
- Остальные поля следует оставить пустыми.

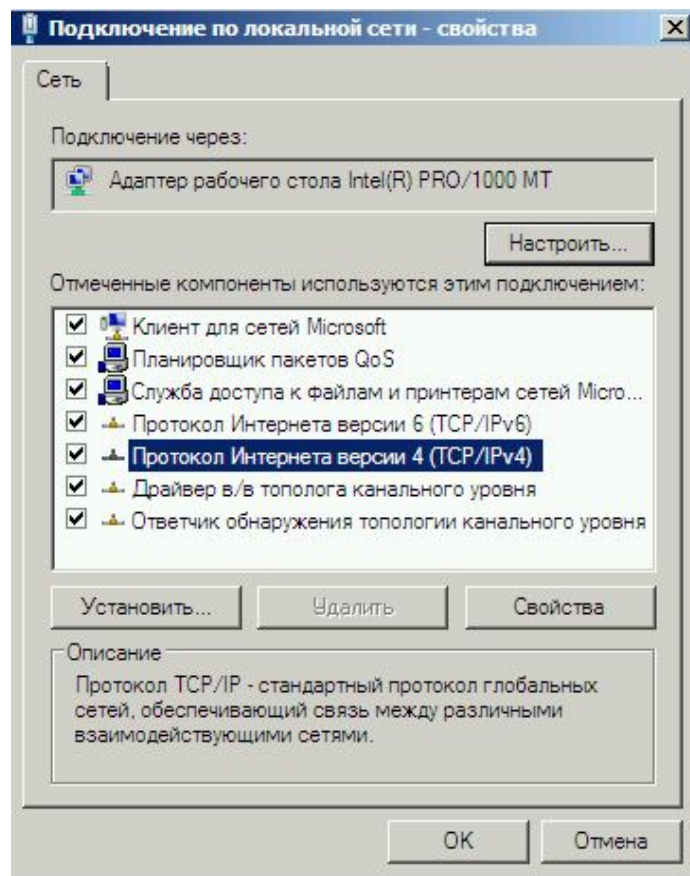


Рис. 38. Диалоговое окно **Подключение по локальной сети – свойства**

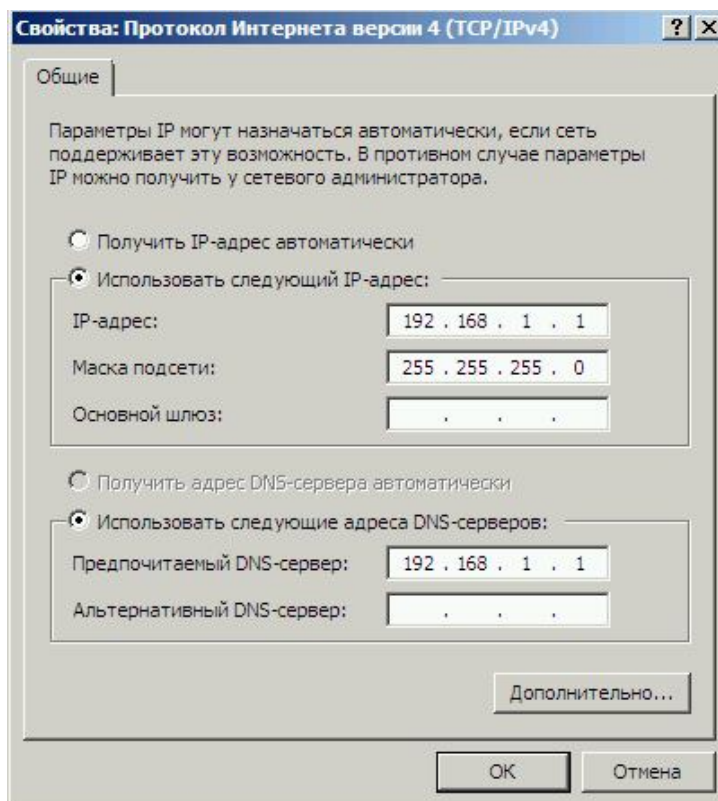


Рис. 39. Диалоговое окно **Свойства: Протокол Интернета версии 4 (TCP/ IPv4)**

Отключение брандмауэра в Windows 2008 Server

Далее следует отключить брандмауэр в операционной системе Windows 2008 Server. Для этого необходимо открыть **Панель управления** (рис. 40) и нажать значок **Безопасность**.

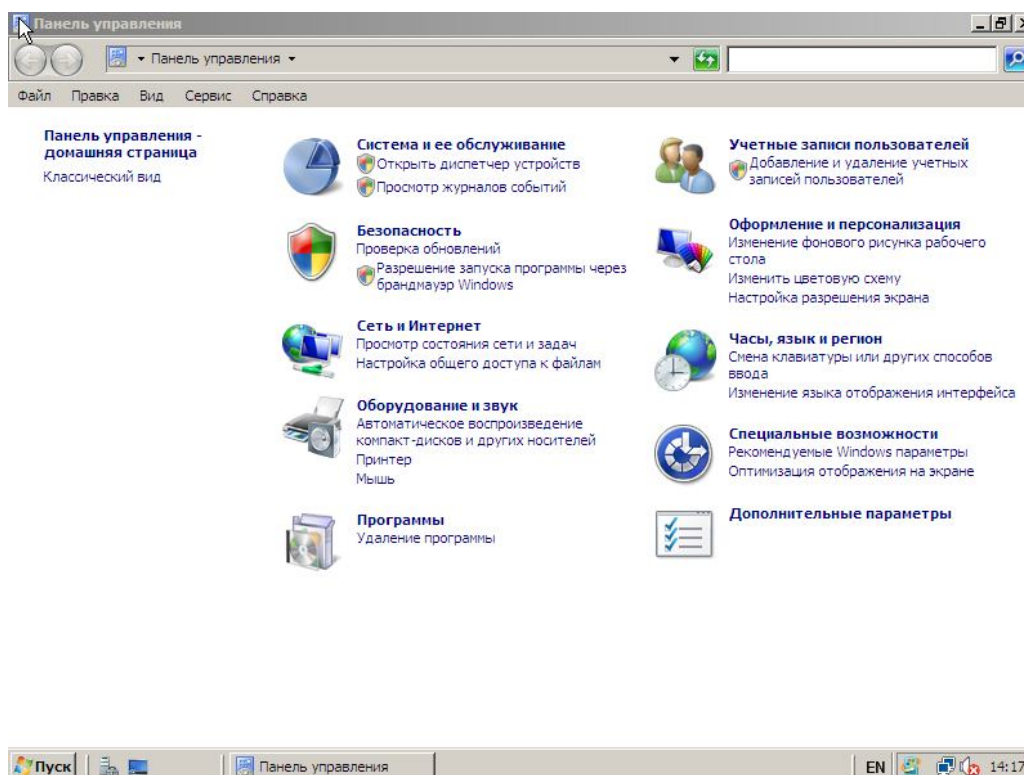


Рис. 40. Панель управления Windows 2008 Server

Далее следует перейти по ссылке **Включение и отключение брандмауэра Windows** (рис. 41).

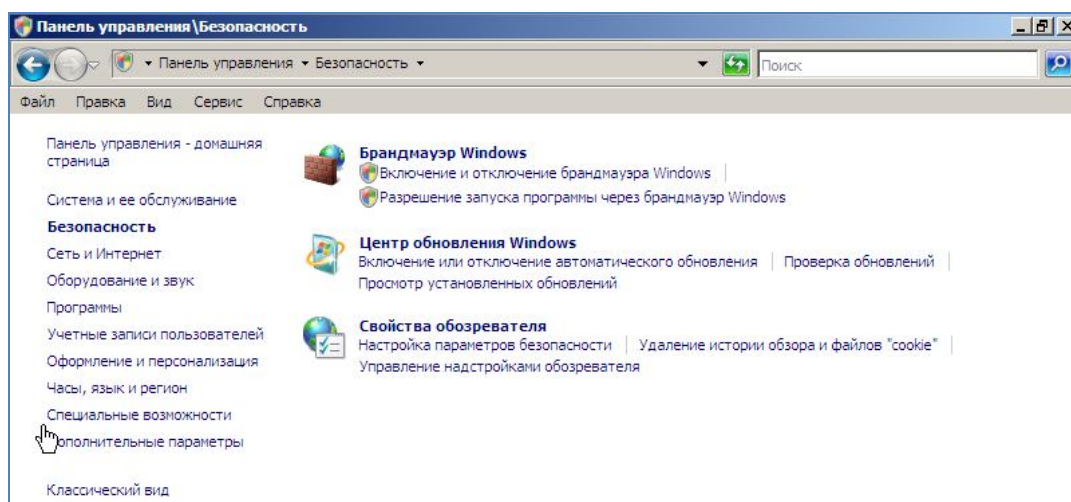


Рис. 41. Оснастка **Безопасность** Панели управления Windows 2008 Server

В открывшемся окне **Параметры брандмауэра Windows** (рис. 42) следует установить состояние **Выкл.**

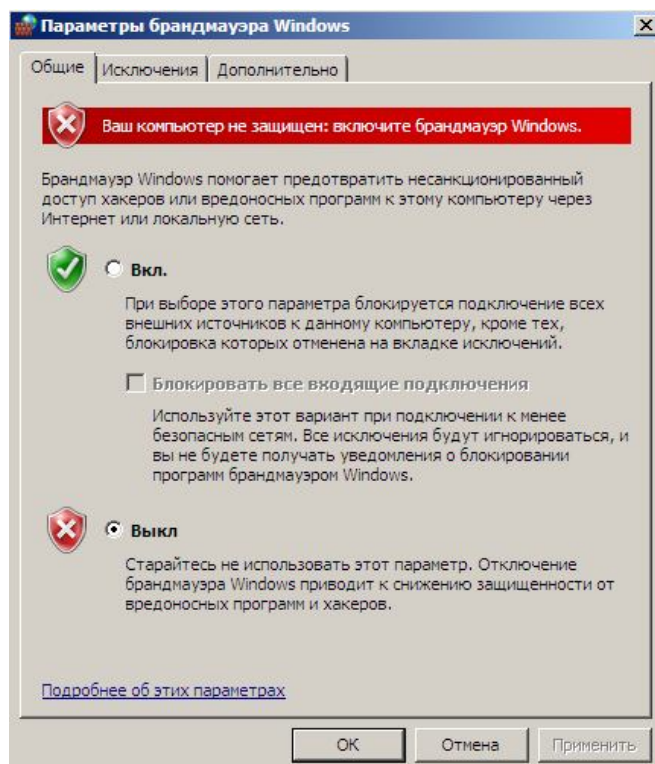


Рис. 42. Диалоговое окно **Параметры брандмауэра Windows**

Тестирование параметров сетевого интерфейса виртуальной машины под управлением Windows 2008 Server

Для проверки настроек статического IP-адреса в командной строке следует выполнить команду **ipconfig** (рис. 43).

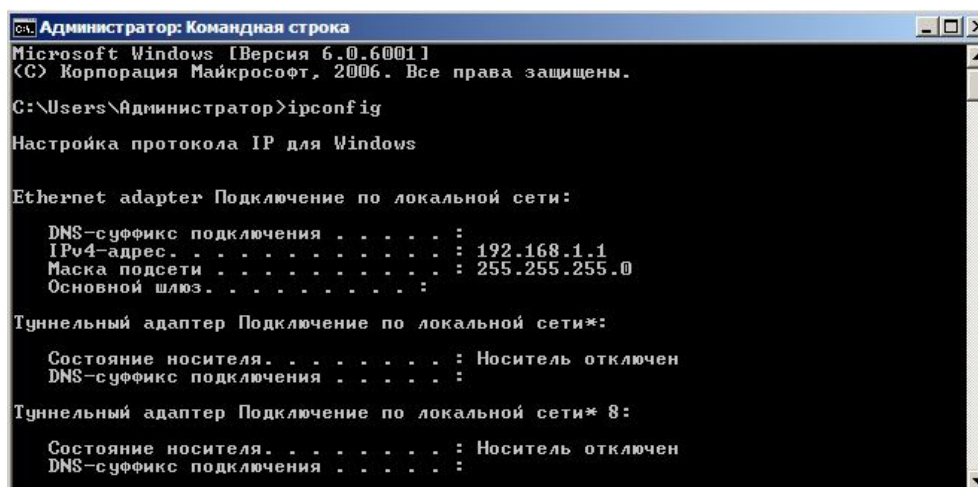
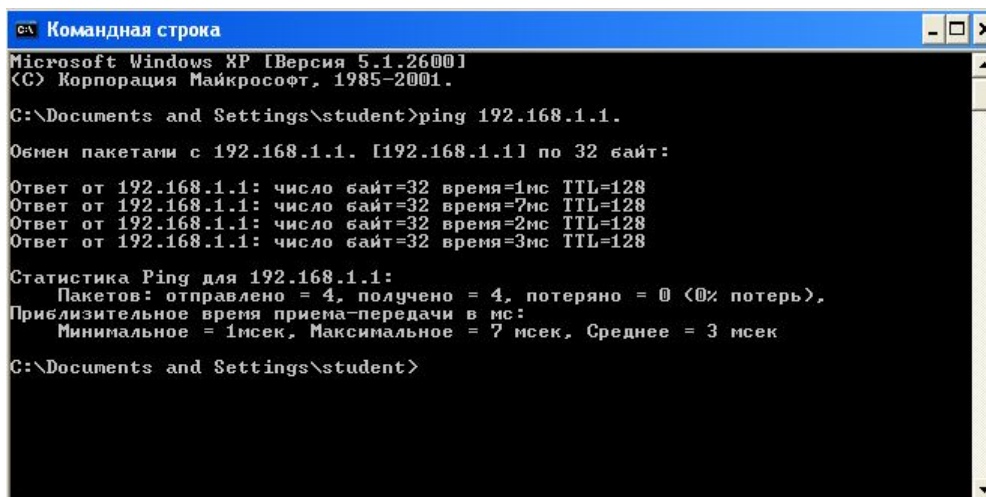


Рис. 43. Тестирование настроек сетевого интерфейса виртуальной машины под управлением Windows 2008 Server

Проверка работоспособности сетевого соединения между виртуальными машинами

Для проверки работоспособности сетевого соединения между виртуальными машинами CLIENT (под управлением Windows XP) и SERVER (под управлением Windows 2008 Server) на компьютере CLIENT в командной строке необходимо выполнить команду **ping 192.168.1.1**. Результаты ее выполнения приведены на рисунке 44.



```
Microsoft Windows XP [Версия 5.1.2600]
(C) Корпорация Майкрософт, 1985-2001.

C:\Documents and Settings\student>ping 192.168.1.1.

Обмен пакетами с 192.168.1.1: [192.168.1.1] по 32 байт:

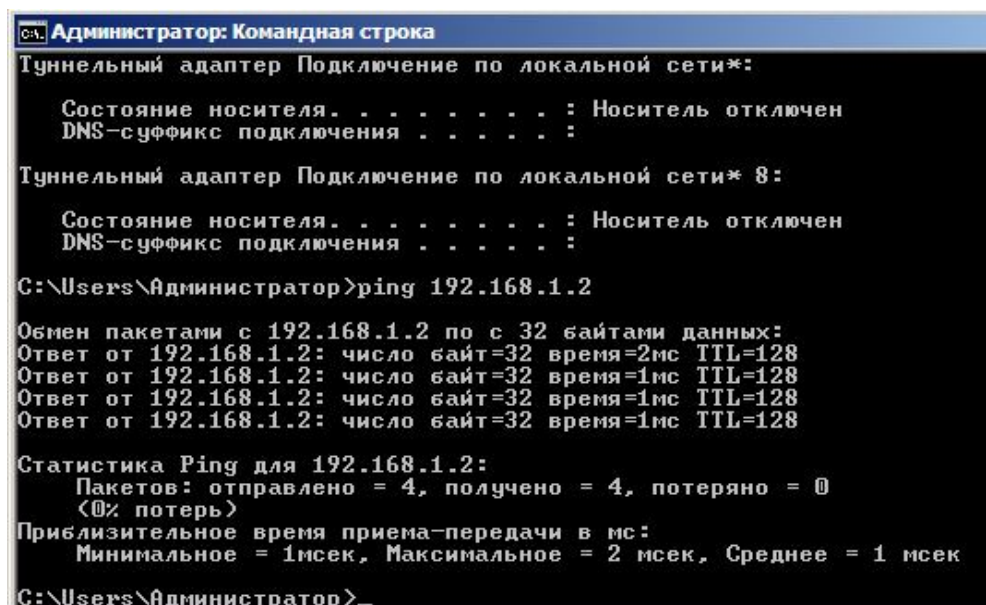
Ответ от 192.168.1.1: число байт=32 время=1мс TTL=128
Ответ от 192.168.1.1: число байт=32 время=7мс TTL=128
Ответ от 192.168.1.1: число байт=32 время=2мс TTL=128
Ответ от 192.168.1.1: число байт=32 время=3мс TTL=128

Статистика Ping для 192.168.1.1:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь).
Приблизительное время приема-передачи в мс:
    Минимальное = 1мсек, Максимальное = 7 мсек, Среднее = 3 мсек

C:\Documents and Settings\student>
```

Рис. 44. Результаты выполнения команды **ping** на CLIENT

Вместо этого можно также на компьютере SERVER в командной строке выполнить команду **ping 192.168.1.2**. Результаты ее выполнения приведены на рисунке 45.



```
Администратор: Командная строка
Туннельный адаптер Подключение по локальной сети*:
    Состояние носителя. . . . . : Носитель отключен
    DNS-суффикс подключения . . . . . :

Туннельный адаптер Подключение по локальной сети* 8:
    Состояние носителя. . . . . : Носитель отключен
    DNS-суффикс подключения . . . . . :

C:\Users\Администратор>ping 192.168.1.2

Обмен пакетами с 192.168.1.2 по с 32 байтами данных:
Ответ от 192.168.1.2: число байт=32 время=2мс TTL=128
Ответ от 192.168.1.2: число байт=32 время=1мс TTL=128
Ответ от 192.168.1.2: число байт=32 время=1мс TTL=128
Ответ от 192.168.1.2: число байт=32 время=1мс TTL=128

Статистика Ping для 192.168.1.2:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 1мсек, Максимальное = 2 мсек, Среднее = 1 мсек

C:\Users\Администратор>
```

Рис. 45. Результаты выполнения команды **ping** на SERVER

Контрольные вопросы

1. Что представляет собой стек TCP/IP?
2. Перечислите уровни модели DARPA и опишите их функции.
3. Опишите функции протоколов IP, ARP, ICMP и IGMP.
4. Какие существуют протоколы транспортного уровня? В чем отличие между ними?
5. Перечислите наиболее распространенные протоколы прикладного уровня и опишите их функции.
6. Что представляет собой IP-адрес? Из каких частей он состоит?
7. Каким образом формируется идентификатор сети и идентификатор хоста у сетей классов А, В и С?
8. Для каких целей используется маска подсети? Как формируются ее биты? Какие существуют формы представления маски подсети? Как перевести одну форму в другую?
9. Как извлечь идентификатор сети из IP-адреса при помощи маски подсети?
10. Для каких целей используется метод CIDR? Опишите принцип формирования маски подсети при использовании этого метода.
11. Дайте определения общих и частных IP-адресов. Перечислите пространства частных адресов.

Лабораторная работа № 2

СЛУЖБА DNS

Цели работы

Получить теоретические и практические навыки по следующим вопросам:

- развертывание и настройка служб DNS;
- настройка клиентов DNS.

Теоретическое введение

Назначение службы DNS. Преимущества доменных имен над IP-адресами

В TCP/IP поиск хостов и подключение к ним осуществляется по IP-адресам. Однако пользователи предпочитают для тех же целей понятные имена, например **files.imit.local** вместо **192.168.1.200**. Система доменных имен DNS (Domain Name System), определенная в RFC 1034 и 1035, предоставляет стандартную схему именования компьютеров в IP-сетях. DNS позволяет использовать понятные имена компьютеров и других ресурсов в IP-сетях. DNS представляет собой распределенную базу данных, связывающую доменные имена с данными. DNS определяет:

- механизм запроса и обновления базы данных;
- механизм репликации информации баз данных между серверами;
- схему базы данных.

Служба DNS применяется в частных сетях и Интернет для разрешения имен узлов в IP-адреса и определения местоположения компьютеров в ЛВС и Интернете. DNS преобразует имена узлов в IP-адреса. Доменные имена узлов обеспечивают следующие преимущества:

- они более дружелюбны, то есть запоминать доменные имена легче, чем IP-адреса;
- они более стабильны, чем IP-адреса. IP-адрес сервера может измениться, а имя сервера останется прежним;

- доменные имена хостов позволяют пользователям подключаться к локальным серверам по тем же правилам именования, что и в Интернете.

Пространство доменных имен

Пространство доменных имен представляет собой схему именования, обеспечивающую иерархичную структуру БД DNS. Каждый узел представляет раздел БД DNS и называется доменом.

БД DNS индексируется по имени, то есть у каждого домена должно быть имя. При добавлении доменов в иерархичную структуру имя родительского домена добавляется к именам его дочерних доменов (поддоменов). Следовательно, имя домена определяет его положение в иерархии.

Каждая вершина дерева доменов представлена DNS-именем, присвоенным домену или хосту. DNS-домен – это ветвь, исходящая из вершины доменного дерева. DNS-домен может включать как хосты, так и другие домены, называемые поддоменами (subdomains). Каждой организации назначаются полномочия для ее части пространства доменных имен, и она отвечает за администрирование, деление на подмножества и присвоение имен DNS-доменам и компьютерам в пределах своей части пространства имен.

Деление на подмножества является важной концепцией DNS. Создание подмножеств в пространстве доменных имен и в DNS-доменах частных TCP/IP-сетей обеспечивает дальнейший рост Интернета и возможность непрерывного увеличения количества имен и административных групп. Подмножества обычно отражают организационную или географическую структуру (рис. 46).

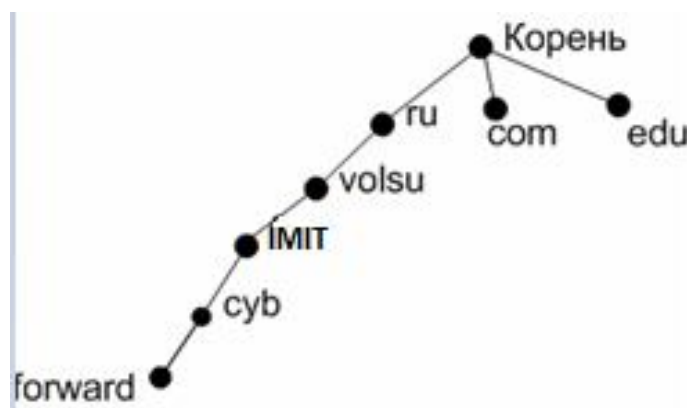


Рис. 46. Система доменных имен

Например, имя IMIT.VOLSU.RU указывает, что домен IMIT состоит в домене VOLSU, а домен VOLSU является поддоменом домена RU.

Понятие «домен» в контексте DNS имеет несколько иное значение, чем в контексте службы каталогов Microsoft Windows 2008. В Windows 2008 домен – это объединение компьютеров и устройств, администрируемых как одно целое. В DNS домен – это узел, представляющий раздел БД DNS.

Структура пространства имен домена включает корневой домен, домены верхнего и последующих уровней и имена узлов.

Корневой домен находится на вершине иерархии и обозначается точкой (.).

Домены верхнего уровня являются 2- или 3-символьными кодами имен. Домены верхнего уровня распределяются по типу или географическому расположению организации.

Домен второго уровня может включать как узлы, так и поддомены.

Пусть VOLSU.RU включает компьютеры, например FTP.VOLSU.RU, и поддомены, например IMIT.VOLSU.RU. Поддомен IMIT.VOLSU.RU. может включать узлы, например:

PRINTSERVER.IMIT.VOLSU.RU.

Имена узлов ссылаются на конкретные компьютеры в Интернете или частной сети.

Например, имя COMPUTER1 является именем хоста. Имя хоста – это левая часть полного доменного имени (fully qualified domain name, FQDN), описывающего точное положение узла в иерархии домена. Например, COMPUTER1.IMIT.VOLSU.RU. (включая последнюю точку, представляющую корневой домен) – полное доменное имя. DNS использует полное доменное имя хоста при разрешении имени в IP-адрес.

Имя узла не обязательно должно совпадать с именем компьютера. По умолчанию программа установки протокола TCP/IP в качестве имени узла использует имя компьютера, заменяя недопустимые символы, например знаки подчеркивания (_), дефисами (-).

Правила именования доменов

При создании пространства имен домена следует придерживаться следующих ограничений:

- обычно записи узлов должны стоять на 3 или 4 уровня, но не более чем на 5 уровней, ниже по иерархии DNS. При увеличении числа уровней увеличивается объем задач администрирования;
- необходимо использовать уникальные имена. Чтобы в пространстве имен DNS были лишь уникальные имена, в домене не должно быть поддоменов с идентичными именами;
- следует использовать простые уникальные имена. Простые имена доменов легче запоминаются и делают возможным интуитивный поиск Web-узлов и других компьютеров в Интернете и интрасети;
- необходимо избегать длинных имен. Доменное имя может включать до 63 символов с учетом точек. Общая длина полного доменного имени не может превышать 255 символов. В именах не учитывается регистр. Используются стандартные символы DNS и Unicode. Windows 2008 поддерживает стандартные символы DNS, определенные в RFC 1035: A–Z, a–z, 0–9 и дефис (-).

Зоны

Зона – непрерывная область в пространстве имен DNS, содержащая набор записей, которые хранятся на DNS-сервере. Каждая зона соответствует определенной вершине в дереве доменов. Однако зоны не являются доменами. DNS-домен – это ветвь дерева, тогда как зона – это область пространства имен DNS, которая обычно хранится в файле и может включать несколько доменов. Домен может быть разбит на несколько разделов, или зон, каждая из которых контролируется своим DNS-сервером. При использовании зон DNS-сервер отвечает на запросы, относящиеся к хостам в зоне, для которой он полномочен.

Зоны бывают основными и дополнительными. Основная зона – та, в которой производятся обновления, а дополнительная – копия, реплицируемая с главного сервера.

Зоны могут храниться по-разному – например, в виде зонных файлов. На серверах с Windows 2008 они могут также храниться в службе каталогов Active Directory. Некоторые дополнительные серверы хранят их в памяти и выполняют зонные передачи при перезапуске.

Для распространения административных задач по группам пространство имен домена делится на несколько зон.

Например, на рисунке 47 пространство имен домена VOLSU.RU разделено на две зоны. Благодаря этому, один администратор может управлять доменами VOLSU и IC, а другой – доменом IMIT. Зона должна охватывать непрерывное пространство имен домена. Например, можно создать зону, охватывающую IC.VOLSU.RU и родительский домен VOLSU.RU, поскольку эти зоны связаны. Однако создать зону, содержащую только домены IMIT.VOLSU.RU и IC.VOLSU.RU, нельзя, поскольку эти домены не связаны. Используемые в зоне привязки «IP-адрес/имя» хранятся в файле БД зоны. Каждая зона прикреплена к определенному домену – корневому домену зоны. Файл БД зоны может содержать сведения не обо всех поддоменах корневого домена зоны.

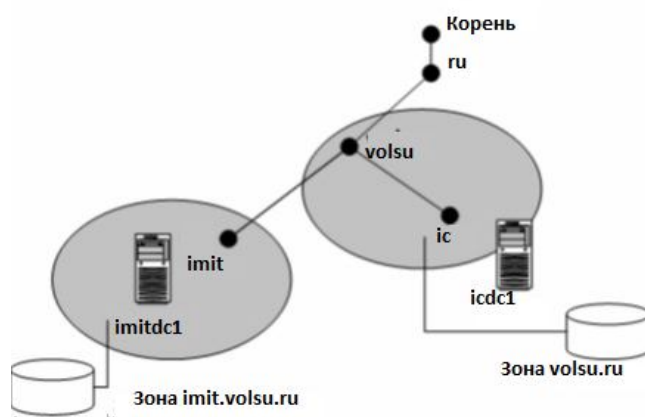


Рис. 47. Пример разделения пространства домена на две зоны

На рисунке 47 VOLSU – корневой домен зоны 1, файл БД которой содержит привязки «IP-адрес/имя» для доменов VOLSU и IC. Корневым доменом зоны 2 является домен IMIT, и файл БД этой зоны содержит привязки «IP-адрес/имя» лишь для домена IMIT. Файл БД зоны 1 не содержит привязок «IP-адрес/имя» для домена IMIT, хотя тот и является поддоменом домена VOLSU.

Серверы имен DNS.

DNS-серверы (DNS servers) – компьютеры, на которых выполняются серверные программы DNS, содержащие информацию о древовидной структуре DNS-домена. DNS-серверы также обрабатывают клиентские запросы. Получив запрос, DNS-сервер может предоставить запрашиваемую информацию или указатель на другой сервер, способный выполнить запрос, а также сообщить, что не располагает запрашиваемой информацией или что такой информации нет.

DNS-интерпретаторы (DNS resolvers) – программы, использующие запросы DNS для получения информации от серверов. Интерпретаторы могут взаимодействовать как с удаленными DNS-серверами, так и с серверными программами DNS, работающими на локальном компьютере. Интерпретаторы обычно встроены в служебные программы либо вызываются через библиотечные функции. Могут работать на любых компьютерах, включая DNS-серверы.

DNS-серверы либо вообще не хранят сведения о зонах, либо хранят информацию об одной или нескольких зонах. DNS-сервер, получая DNS-запрос, пытается найти запрашиваемую информацию в своих локальных зонах. Если ему это не удастся из-за того, что он не является полномочным сервером для запрашиваемого домена, он может:

- проверить свой кэш,
- связаться с другими DNS-серверами для выполнения запроса,
- указать клиенту DNS-сервер, который, возможно, ответит на запрос.

DNS-серверы управляют основными и дополнительными зонами. Сервер можно настроить на управление произвольным количеством основных или дополнительных зон, например:

- одной основной зоной и дополнительной копией другой,
- только основными зонами,
- только дополнительными копиями.

Сервер, управляющий основной зоной, считается основным сервером этой зоны, а сервер, управляющий дополнительными зонами, – дополнительным сервером этих зон.

Основные зоны обновляются локально. При модификации данных в зоне (например, при делегировании части зоны другому серверу или добавлении записей ресурсов в зону) эти изменения должны быть сделаны на основном сервере этой зоны, чтобы новая информация была внесена в локальную зону.

Дополнительные зоны реплицируются с другого сервера. При определении зоны на дополнительном сервере ее настраивают на IP-адрес того сервера, с которого она будет реплицироваться. Сервер, с которого реплицируется зонный файл, может быть как основным, так и дополнительным сервером этой зоны и иногда называется основным сервером дополнительной зоны.

При запуске дополнительный сервер связывается с основным сервером своей зоны и инициирует передачу информации из зоны (зонную передачу). Кроме того, дополнительный сервер периодически обращается к основному серверу, чтобы определить, не изменилась ли информация в зоне. Если да, он также может инициировать зонную передачу.

В зоне должен иметься хотя бы один сервер имен, но их может быть и несколько. Один из них (основной сервер зоны) содержит мастер-файл БД этой зоны, или первичный файл БД зоны. Изменения в конфигурации зоны, например добавление доменов или хостов, обрабатываются на сервере, содержащем первичный файл БД зоны. Все остальные серверы имен (дополнительные серверы зоны), связанные с данной зоной, являются резервными и содержат вторичные файлы БД.

Наличие множества серверов имен дает следующие преимущества:

- избыточность – Дополнительные серверы имен получают от сервера, содержащего первичный файл БД зоны, копию этого файла. Этот процесс называется передачей зоны. Резервные серверы периодически обращаются к серверу, содержащему первичный файл БД, за обновленными сведениями о конфигурации зоны. При сбое сервера, содержащего первичный файл БД зоны, в работу включаются резервные серверы;
- повышение скорости доступа удаленных клиентов – при наличии удаленных клиентов дополнительные серверы имен

позволят снизить трафик запросов в низкоскоростных каналах связи с Интернет;

- снижение нагрузки – дополнительные серверы имен уменьшают нагрузку на сервер, содержащий первичный файл БД зоны. Кроме того, благодаря БД Active Directory, Windows 2008 поддерживает хранилище зоны, интегрированное с каталогом. Зоны, хранимые подобным образом, содержатся в дереве Active Directory в объекте-контейнере Domain. Каждая зона, интегрированная с каталогом, хранится в объекте-контейнере зоны DNS, которому присваивается имя зоны.

Серверы кэширования

Кэширование выполняют все DNS-серверы: получив информацию от других серверов, они определенное время ее хранят. Это ускоряет разрешение имен, уменьшает трафик DNS-запросов и повышает надежность.

Некоторые DNS-серверы, называемые серверами кэширования, просто выдают запросы, кэшируют ответы и возвращают результаты. Они не имеют полномочий ни в каких DNS-доменах и не управляют зонами, а лишь кэшируют информацию, получаемую в ответ на запросы.

Серверы кэширования полезны тем, что не создают трафик, связанный с зонными передачами. Однако им свойствен один недостаток: при начальном запуске такой сервер не располагает никакой информацией и должен какое-то время накапливать ее, кэшируя обслуживаемые запросы.

Серверы пересылок и ведомые серверы

Получив запрос, DNS-сервер пытается найти запрашиваемую информацию в своих локальных зонах и в кэше. Если ему не удастся обнаружить запрашиваемую информацию и он не имеет полномочий для соответствующих зон, этот сервер должен обратиться к другим серверам. Однако в некоторых случаях сетевые администраторы предпочитают, чтобы такой сервер не взаимодействовал с другими серверами напрямую. Так, если организация подключена к Интернету по низкоскоростному каналу связи, не желательно, чтобы каждый ее DNS-сервер напрямую соединялся с DNS-серверами в Интернете. Для

решения этой проблемы DNS позволяет использовать серверы пересылок – DNS-серверы, предназначенные для пересылки запросов другим серверам.

Например, можно назначить сервером пересылки один из DNS-серверов организации и использовать его для разрешения имен компьютеров в Интернете, а остальные серверы настроить так, чтобы они обращались к нему для разрешения имен, по которым они не полномочны.

Компьютер, используемый как сервер пересылок, не требует специальных настроек.

Необходимо лишь сконфигурировать DNS-серверы, которым необходима пересылка запросов, указав IP-адрес сервера пересылок.

Сервер может использовать сервер пересылок в неэксклюзивном или эксклюзивном режиме.

В неэксклюзивном режиме сервер, получив запрос DNS, на который у него нет полномочий и который он не в состоянии выполнить, передает запрос одному из серверов пересылок. Сервер пересылок связывается с другими серверами и передает результат запросившему серверу, который в свою очередь возвращает этот результат хосту, выдавшему запрос. Если сервер пересылок не может выполнить запрос, исходный сервер пытается обработать этот запрос самостоятельно.

В эксклюзивном режиме серверы полностью полагаются на серверы пересылок и называются ведомыми. Когда ведомый сервер получает запрос, который он не может разрешить через свои зоны, он передает запрос одному из серверов пересылок. Тот отсылает запрос полномочному серверу и передает результат запросившему серверу, который в свою очередь возвращает результат хосту, выдавшему запрос. Если сервер пересылок не может выполнить запрос, ведомый сервер сообщает хосту о неудаче.

Ведомые серверы не пытаются самостоятельно выполнить запрос, если сервер пересылок не смог на него ответить.

Процесс разрешения имен

Разрешение имен – это процесс преобразования имен в IP-адреса. Например, при подключении к Web-узлу VOLSU.RU, используется имя WWW.VOLSU.RU. DNS разрешает это имя в

соответствующий IP-адрес. Привязки «IP-адрес/имя» хранятся в распределенной БД DNS. Серверы имен DNS разрешают прямые и обратные запросы на поиск имени. Прямой запрос разрешает имя в IP-адрес. Обратный запрос разрешает IP-адрес в имя. Сервер имен может разрешать запросы лишь для той зоны, в которой он обладает полномочиями. Если сервер не может разрешить запрос, он передает его другому серверу имен, который сможет это сделать. Для снижения DNS-трафика в сети сервер имен кэширует результаты запроса.

Рекурсивные и итеративные запросы.

Выдавая рекурсивный запрос, DNS-клиент требует, чтобы DNS-сервер вернул ему либо запрашиваемую запись ресурса, либо сообщение об ошибке, указывающее на отсутствие записи или доменного имени. DNS-сервер не может отослать DNS-клиент к другому DNS-серверу.

Таким образом, если DNS-сервер, получивший рекурсивный запрос, не располагает запрашиваемой информацией, он обращается к другим серверам до тех пор, пока не получит эту информацию или пока не выяснит, что запрос невыполним.

Рекурсивные запросы обычно выдаются DNS-клиентами или DNS-серверами, настроенными на передачу запросов другим серверам, то есть серверами пересылок.

Выдавая итеративный запрос, DNS-клиент дает возможность серверу вернуть наиболее полезный ответ, который тот может подготовить на основе имеющейся в его зоне или кэше информации. Если запрашиваемый DNS-сервер не имеет точной информации о запрашиваемом имени, наиболее полезная информация, которую он способен вернуть, – ссылка (referral) (то есть указатель на DNS-сервер, полномочный для более низкого уровня пространства доменных имен). DNS-клиент может затем запросить DNS-сервер, на который он получил ссылку. Клиент продолжает этот процесс, пока не найдет полномочный для запрашиваемого имени DNS-сервер, пока не возникнет ошибка или пока не истечет время ожидания.

Этот процесс иногда называют «просмотром дерева», и такой тип запросов обычно используется DNS-сервером, пытающимся выполнить рекурсивный запрос DNS-клиента. Примеры итеративного и

рекурсивного запросов показаны на рисунке 48. Предполагается, что ни один из серверов не имеет запрашиваемой информации в своем кэше.

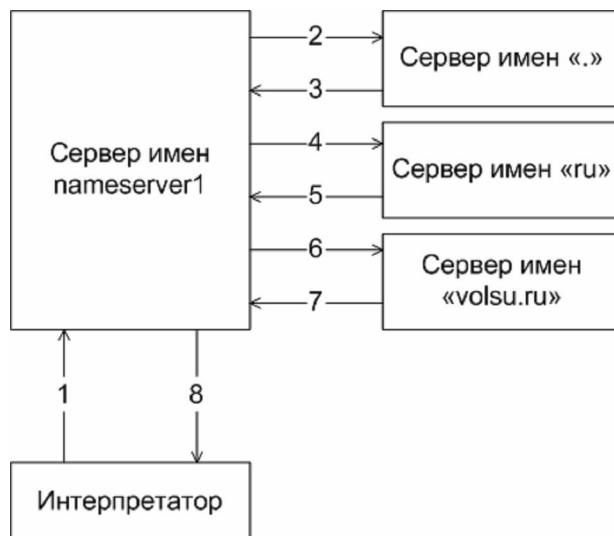


Рис. 48. Пример итеративных и рекурсивных запросов

1. Клиенту, который находится где-то в Интернете, требуется IP-адрес **imit.volsu.ru**. В этом случае происходят следующие события.

2. Клиент связывается с сервером **nameserver1**, делая рекурсивный запрос имени **imit.volsu.ru**. Сервер должен вернуть или ответ, или сообщение об ошибке.

3. **Nameserver1** проверяет свой кэш и зоны, пытаясь найти ответ, но не находит его. Тогда он связывается с полномочным сервером Интернета (корневым сервером), выдавая итеративный запрос имени **imit.volsu.ru**.

4. Корневому серверу Интернета ответ не известен, и он отвечает ссылкой на полномочный сервер домена **.ru**.

5. **Nameserver1** связывается с полномочным сервером домена **.ru**, выдавая итеративный запрос имени **imit.volsu.ru**.

6. Полномочный сервер домена **.ru** тоже не знает ответа и отвечает ссылкой на полномочный сервер домена **volsu.ru**.

7. **Nameserver1** связывается с полномочным сервером домена **volsu.ru**, выдавая итеративный запрос имени **imit.volsu.ru**.

8. Полномочный сервер домена **volsu.ru** знает ответ и сообщает запрошенный IP-адрес.

9. **Nameserver1** отвечает на клиентский запрос, возвращая IP-адрес **imit.volsu.ru**.

Кэширование и TTL

При обработке рекурсивного запроса серверу часто требуется посылать несколько запросов, чтобы получить окончательный ответ. Сервер кэширует всю принимаемую в ходе этого процесса информацию на период, указываемый в возвращаемых данных. Этот период называется временем жизни (Time-To-Live, TTL) и выражается в секундах. Значение TTL определяется администратором сервера основной зоны, содержащей искомые данные. Меньшие значения TTL прибавляют уверенности в том, что информация о домене соответствует действительности при ее частом изменении, но повышают нагрузку на сервер имен и увеличивают трафик в Интернете. Поскольку данные кэшируются, изменения в записях ресурсов не могут быть доступны сразу во всем Интернете. Как только данные помещены в кэш DNS-сервера, последний хранит их в течение заданного TTL. Включая в ответ данные из кэша, DNS-сервер сообщает оставшееся время жизни этих данных. В свою очередь, интерпретатор может кэшировать эти данные, используя TTL, указанное сервером.

Отрицательное кэширование

Кроме выполненных запросов, интерпретаторы и серверы могут кэшировать и отрицательные ответы, то есть информацию о том, что определенные наборы записей ресурсов или имена DNS-доменов не существуют. Отрицательное кэширование уменьшает время получения отрицательных ответов. Это также уменьшает сетевой трафик, сокращая число сообщений, пересылаемых как между интерпретаторами и серверами имен, так и между самими серверами имен. Отрицательное кэширование описывается в RFC 1034 и 2308.

Зоны прямого просмотра

Зона прямого просмотра позволяет генерировать прямые запросы поиска имени. Для работы службы DNS на сервере имен необходимо сконфигурировать не менее одной зоны прямого просмотра.

Зоны прямого просмотра содержат информацию, необходимую для разрешения имен в DNS-домене. Они должны включать записи SOA и NS, а также могут включать записи любых других типов, кроме PTR.

Возможно создание зоны одного из трех типов:

1. Active Directory-integrated (Интегрированная в Active Directory). Главная копия новой зоны, использует для хранения и репликации файлов зоны службу Active Directory. Зоны такого типа обеспечивают безопасное обновление и интегрированное хранение. Стандартные зонные передачи не осуществляются – файл БД зоны реплицируются одновременно с хранилищем Active Directory.

2. Standard primary (Основная). Главная копия новой зоны, хранится как обычный текстовый файл. Администрирование и поддержка основной зоны осуществляется на том компьютере, где была создана. Зоны такого типа упрощают обмен DNS-данными с другими серверами DNS, хранящими данные в виде текста.

3. Standard secondary (Дополнительная). Реплика существующей зоны хранится в обычных текстовых файлах и доступна только для чтения. Для создания дополнительной зоны надо сначала создать основную. При создании следует указать основной DNS-сервер, который передает информацию о зоне на сервер имен, содержащий дополнительную зону. Дополнительные зоны создаются для обеспечения избыточности и уменьшения нагрузки на сервер имен, содержащий основной файл БД зоны. Обычно зоне присваивается имя наивысшего домена в иерархии, охватываемой зоной, то есть имя корневого домена зоны. Например, зоне, включающей домены VOLSU.RU и IMIT.VOLSU.RU, будет присвоено имя VOLSU.RU.

Имя файла зоны – это имя файла БД, по умолчанию состоящее из имени зоны с расширением .DNS. Например, если имя зоны – VOLSU.RU, то файл БД по умолчанию будет называться VOLSU.RU.DNS.

Зоны обратного просмотра

В большинстве запросов клиент указывает имя и запрашивает IP-адрес, соответствующий этому имени. Запросы такого типа называют прямым просмотром (forward lookup). В случае, если у клиента уже есть IP-адрес компьютера и он хочет определить его DNS-имя, используется стандарт DNS, который называется обратным просмотром (reverse lookup).

Зоны обратного просмотра позволяют генерировать обратные запросы на поиск имени. Эти зоны не обязательны, однако они нужны для работы утилит устранения неполадок, таких как NSLOOKUP.

Зоны обратного просмотра содержат информацию, необходимую для обратного просмотра, и обычно включают записи SOA, NS, PTR и CNAME.

Типы зон обратного просмотра соответствуют типам зон прямого просмотра: интегрированная в Active Directory, основная и дополнительная.

Поскольку распределенная БД DNS индексируется по имени, а не по IP-адресу, при обработке обратного запроса должен производиться полный перебор всех доменных имен. Для решения этой проблемы создан специальный домен второго уровня IN-ADDR.ARPA. Этот домен придерживается той же иерархичной системы именования, но основывается не на доменных именах, а на IP-адресах:

- поддоменам присваиваются имена, соответствующие IP-адресам (4 октета, разделенные точками);
- порядок октетов IP-адреса меняется на противоположный;
- организации администрируют поддомены домена IN-ADDR.ARPA, основываясь на назначенных им IP-адресах и маске подсети.

Например, организация, которой выделен диапазон IP-адресов от 169.254.16.0 до 169.254.16.255 с маской подсети 255.255.255.0, обладает полномочиями в отношении домена 16.254.169.IN-ADDR.ARPA.

Для создания зоны следует указать идентификатор сети (network ID) или имя зоны обратного просмотра. Если в идентификаторе сети указан 0, он появится в имени зоны. Например, для идентификатора сети 169 будет создана зона 169.in-addr.arpa, а для идентификатора сети 169.0 – зона 0.169.in-addr.arpa.

Имя файла зоны по умолчанию определяется идентификатором сети и маской подсети. DNS обращает порядок октетов IP-адреса и добавляет суффикс IN-ADDR.ARPA. Например, имя файла зоны обратного просмотра для сети 169.254 будет 254.169.IN-ADDR.ARPA.DNS.

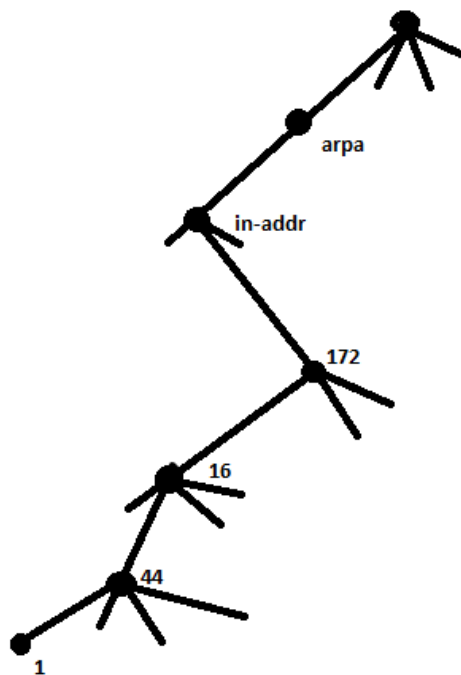


Рис. 49. Пространство имен **in-addr.arpa**

В дереве доменов **in-addr.arpa** для обратного сопоставления IP-адресов с уникальными доменными именами используются записи ресурсов PTR. Например, если клиенту нужно найти доменное имя, сопоставленное с IP-адресом **172.16.44.1**, он запрашивает запись ресурса PTR для доменного имени **1.44.16.172.in-addr.arpa**.

Записи ресурсов и зоны

Записи ресурсов (resource records) – совокупность информации в базе данных DNS, которую можно использовать для обработки клиентских запросов. Каждый DNS-сервер содержит записи ресурсов, необходимые для ответов на запросы по той части пространства имен, для которой он является полномочным сервером. DNS-сервер считается полномочным для непрерывной области пространства имен DNS, если он содержит информацию об этой области.

При разрешении имен серверы просматривают свои зоны, которые физически представляют собой файлы базы данных DNS. Зоны содержат записи ресурсов, которые и составляют информацию, связанную с DNS-доменом. Например, одни записи ресурсов связывают DNS-имена с IP-адресами, а другие – IP-адреса с DNS-именами.

Формат записей ресурсов

Записи ресурсов имеют следующий синтаксис.

Название	Описание
Owner (Владелец)	Имя хоста или DNS-домена, которому принадлежит эта запись ресурсов.
TTL (Время жизни)	32-битное целое число, представляющее интервал времени (в секундах), в течение которого DNS-сервер или интерпретатор должен кэшировать эту запись до того, как ее отбросить. Это поле необязательно, и, если оно не определено, клиент использует минимальное TTL, указанное в SOA.
Class (Класс)	Определяет используемое семейство протоколов. Для Интернета практически всегда содержит IN. Другое значение, определенное в RFC 1034, – это CH, для системы Chaos, использовавшейся для экспериментаторов в Массачусеттском технологическом институте.
Type (Тип)	Определяет тип записи ресурсов.
RDATA (Данные)	Данные записи ресурсов. Переменная содержит информацию, связанную с записью данного типа. Например, в записи типа A эта переменная представляет собой 32-разрядный IP-адрес, определяющий соответствующий хост.

В IP-пакетах записи ресурсов представляются в двоичной форме, а в файле базы данных DNS – как текст (обычно в виде одной строки). Если этот текст не уместается на одной строке, данные заключаются в скобки. В большинстве реализаций DNS многострочными могут быть лишь записи SOA (Start of Authority). Кроме того, в зонные файлы часто вставляют пустые строки и комментарии, которые игнорируются DNS-сервером. Комментарии всегда начинаются с точки с запятой (;) и заканчиваются символом возврата каретки.

Типы записей ресурсов

Для предоставления DNS-информации в TCP/IP-сетях используются записи различных типов, в частности:

- SOA;
- NS;
- PTR;
- CNAME;
- MX;
- SRV.

Запись ресурса SOA

В начале каждой зоны содержится запись SOA (Start of Authority), которая включает следующие поля:

- Owner (Владелец);
- TTL (Время жизни);
- Class (Класс);
- Type (Тип);
- Authoritative server (Полномочный сервер) – задает основной полномочный DNS-сервер зоны;
- Responsible person (Ответственное лицо) – указывает адрес электронной почты администратора, ответственного за зону. Вместо символа @ в адресе ставится **точка**;
- Serial number (Порядковый номер) – показывает, сколько раз проводилось обновление зоны. Дополнительный сервер зоны определяет, нужно ли начинать номером основного сервера. Если порядковый номер основного сервера выше, дополнительный иницирует зонную передачу;
- Refresh (Период обновления) – сообщает, как часто дополнительный сервер зоны проверяет наличие изменений в ней;
- Retry (Повторная попытка) – задает, сколько времени дополнительный сервер, послав запрос на зонную передачу, будет ждать ответа от основного сервера, прежде чем предпринять следующую попытку запроса;
- Expire (Время устаревания) – показывает, как долго дополнительный сервер будет отвечать на запросы после предыдущей зонной передачи, прежде чем сбросит свою зону как устаревшую;
- Minimum TTL (Минимальное время жизни) – относится ко всем записям ресурсов зоны, в которых не установлены

значения TTL. Всякий раз, когда интерпретатор запрашивает сервер, последний возвращает записи ресурсов и минимального TTL. Отрицательные ответы кэшируются в течение минимального TTL записи ресурса SOA полномочной зоны.

Пример записи ресурса SOA:

```
imit.volsu.ru IN SOA (  
    imitdc1.imit.volsu.ru      ; полномочный сервер зоны  
    administrator.imit.volsu.ru ; адрес администратора зоны  
    ; (ответственного лица)  
    5099                      ; порядковый номер  
    3600                      ; период обновления (1 час)  
    600                      ; повторные попытки (10 мин.)  
    86400                    ; время устаревания (1 сутки)  
    60 )                     ; минимальный TTL (1 мин.)
```

Запись ресурса NS

Записи ресурса серверов имен NS (name server) определяют полномочные серверы зоны. Они указывают основные и дополнительные серверы, определенные в записях ресурсов SOA, а также серверы для всех делегированных зон. В корне каждой зоны должна быть минимум одна запись ресурса NS. Например, когда администратор **volsu.ru** делегирует полномочия для поддомена **imit.volsu.ru** серверу **imitdc1.imit.volsu.ru**, в зоны **volsu.ru** и **imit.volsu.ru** добавляются следующие записи: **imit.volsu.ru IN NS imitdc1.imit.volsu.ru**

Запись ресурса A

Запись ресурса адреса (**address, A**) сопоставляет полное доменное имя с IP-адресом, благодаря чему интерпретаторы могут получать IP-адреса по именам. Например, приведенная ниже запись ресурса адреса, расположенная в зоне **imit.volsu.ru**, сопоставляет FQDN сервера с его IP-адресом: **imitdc1 IN A 172.16.48.1**

Запись ресурса PTR

Запись ресурса указателя (pointer, PTR), в отличие от записи ресурса адреса A, сопоставляет IP-адрес с полным доменным именем. Например, следующая запись ресурса PTR сопоставляет IP-адрес сервера `imitc1.imit.volsu.ru` с его именем:

`1.48.16.172.in-addr.arpa IN PTR imitdc1.imit.volsu.ru`

Запись ресурса CNAME

Запись ресурса канонического имени (canonical name, CNAME) создает псевдоним (имя-синоним) полного доменного имени. Такие записи позволяют скрывать детали реализации сети от подключенных к ней клиентов.

Например, необходимо установить FTP-сервер с именем `ftp1.imit.volsu.ru` в поддомене `imit.volsu.ru`, но при этом известно, что через некоторое время потребуется перенести его на компьютер с именем `ftp2.imit.volsu.ru` и необходимо, чтобы пользователи не заметили это изменение. Можно создать псевдоним с именем `ftp.imit.volsu.ru`, указывающий на `ftp1.imit.volsu.ru`, а затем, переместив FTP-сервер на другой компьютер, просто изменить запись ресурса CNAME так, чтобы она указывала на сервер `ftp2.imit.volsu.ru`. Запись ресурса CNAME, создающая псевдоним для `ftp1.noam.reskit.com`, будет выглядеть так:

`ftp.imit.volsu.ru IN CNAME ftp1.imit.volsu.ru`

Когда DNS-клиент запрашивает запись ресурса A для `ftp.imit.volsu.ru`, DNS-сервер находит запись ресурса CNAME, обращается к `ftp1.imit.volsu.ru` и возвращает клиенту записи ресурсов A и CNAME.

Согласно RFC 2181, каждый псевдоним должен быть связан только с одним каноническим именем.

Запись ресурса MX

Запись ресурса почтового сервера (mail exchange, MX) определяет почтовый сервер для владельца доменного DNS-имени. Почтовый сервер – это хост, который либо обрабатывает, либо пересылает почту. Под обработкой почты подразумевается ее доставка адресату или передача другому почтовому транспорту.

Под пересылкой почты понимается ее отправка на конечный сервер-получатель, передача по протоколу SMTP на ближайший к получателю почтовый сервер или размещение в очереди на определенное время.

Записи ресурсов MX используются только почтовыми серверами. Если необходимо развернуть несколько почтовых серверов в одном DNS-домене, для этого домена используются несколько записей ресурсов MX.

Пример записей ресурсов MX для домена **imit.volsu.ru**:

*.imit.volsu.ru . IN MX 0 mailserver1.imit.volsu.ru.

*.imit.volsu.ru. IN MX 10 mailserver2.imit.volsu.ru.

Первые три поля в этих записях – стандартные поля владельца, класса и типа записи. Четвертое определяет приоритет почтового сервера. Оно сообщает, насколько предпочтительно использование той или иной записи ресурса MX по сравнению с другими. Предпочтение отдается записям с наименьшими значениями в поле приоритета. Таким образом, программа доставки почты, которой необходимо отправить почту в определенный DNS-домен, связывается с DNS-сервером этого домена и просматривает все записи почтовых серверов. Она выбирает почтовый сервер, для которого задано наименьшее значение в поле приоритета.

Например, пользователь посылает почтовое сообщение по адресу vasya@imit.volsu.ru в тот день, когда сервер mailserver1 неисправен, а сервер mailserver2 работает. Его почтовая программа попытается доставить почту серверу mailserver1, поскольку он более предпочтителен, но не сможет этого сделать, так как сервер не работает. Тогда почтовая программа, выбрав сервер mailserver2, благополучно доставит сообщение на этот сервер.

Запись ресурса SRV

Записи ресурсов служб (service, SRV) дают возможность определять адреса серверов конкретных служб, протоколов и DNS-доменов.

Например, если в домене имеется два Web-сервера, можно создать записи ресурсов SRV, определяющие, какие хосты

обслуживают Web-серверы, и интерпретаторы будут считывать эти записи для Web-серверов.

Формат записи ресурса SRV выглядит так:

`_Служба._Протокол.Имя TTL Класс SRV Приоритет Вес Порт Цель,`

где:

- служба – имя службы, например **http** или **telnet**. Некоторые службы определены стандартами, а некоторые могут определяться в соответствии с конкретными потребностями;

- протокол – задает протокол, например TCP или UDP;

- имя – указывает доменное имя, на которое ссылается запись;

- поля TTL и класса идентичны описанным ранее;

- приоритет – определяет приоритет хоста. Клиенты пытаются связываться с тем хостом, для которого в этом поле указано наименьшее значение;

- вес – используется для балансировки нагрузки. Когда в домене несколько записей имеет одинаковое значение приоритета, клиенты чаще пытаются использовать записи с более высоким весовым значением – если только клиенты не поддерживают какой-то другой механизм балансировки нагрузки;

- порт – указывает порт службы на данном хосте;

- цель – сообщает полное доменное имя хоста, поддерживающего службу.

Пример записей ресурсов SRV для Web-серверов:

`_http._tcp.volsu.ru IN SRV 0 0 80 webserver1.imit.volsu.ru`

`_http._tcp.volsu.ru IN SRV 10 0 80 webserver2.imit.volsu.ru`

В примере не указано значение TTL. Поэтому интерпретатор будет использовать минимальное TTL, определенное в записи ресурса SOA.

Если компьютеру нужно найти Web-сервер в DNS-домене volsu.ru, интерпретатор посылает такой запрос

`_http._tcp.www.volsu.ru.`

В ответе DNS-сервера будут содержаться записи SRV, показанные выше. Далее интерпретатор должен выбрать сервер WebServer1 или WebServer2, исходя из их приоритетов. Поскольку в поле приоритета для WebServer1 задано наименьшее значение, будет выбран WebServer1.

При одинаковых приоритетах, но разных весовых значениях клиент выбирает Web-сервер случайным образом, хотя вероятность выбора сервера с наивысшим весовым значением больше.

Затем интерпретатор запрашивает запись ресурса А для сервера webserver1.imit.volsu.ru, и DNS-сервер передает эту запись. Наконец, клиент устанавливает связь с Web-сервером.

Делегирующие и связывающие записи

Делегирующая (delegation record) и связывающая (glue record) записи – это записи, добавляемые в зону для делегирования поддомена в отдельную зону. Первая представляет собой запись NS в родительской зоне, в которой указан сервер имен, полномочный для делегированной зоны, а вторая – запись А для того же сервера имен.

Пример:

сервер имен для DNS-домена volsu.ru должен делегировать полномочия для зоны imit.volsu.ru серверу имен imitdc1.imit.volsu.ru. Тогда в зону volsu.ru добавляются следующие записи:

imit.volsu.ru IN NS imitdc1.imit.volsu.ru

imitdc1.imit.volsu.ru IN A 172.16.54.1

Делегирующие записи необходимы для разрешения имен. Связывающие записи нужны, если сервер имен, полномочный для делегированной зоны, также является членом домена, для которого ему делегируются полномочия.

В приведенном выше примере связывающая запись обязательна, поскольку imitdc1.imit.volsu.ru – член делегированного домена imit.volsu.ru. Однако если бы он был членом другого домена, интерпретатор мог бы выполнить разрешение имен, просто найдя IP-адрес уполномоченного сервера имен по его имени.

Когда интерпретатор выдает запрос на имя, содержащееся в дочерней зоне, серверу имен, полномочному для родительской зоны, этот сервер проверяет свою зону. Делегирующая запись указывает ему, какой сервер имен полномочен для дочерней зоны. После этого сервер, полномочный для родительской зоны, может вернуть интерпретатору требуемую ссылку.

Хранение зон

Стандарты DNS не определяют внутреннюю структуру записей ресурсов, и поэтому она варьируется в разных реализациях DNS. Как правило, зоны хранятся на серверах в виде обычного текста, но это не

является обязательным требованием. В Windows 2008 базу данных DNS можно интегрировать с базой данных Active Directory, и тогда она хранится в формате Active Directory.

Имя	Описание
db.домен	Зона прямого просмотра. Например, DNS-домена volsu.ru db.volsu.ru.
db.адрес	Зона обратного просмотра. Например, если имеется сеть класса С с сетевым адресом 172.16.32, этому файлу присваивается имя db.172.16.32.
db.cache	Этот файл, называемый также файлом корневых ссылок (root hints file), содержит имена и IP-адреса серверов имен, обслуживающих корневой DNS-домен. Данные файлы на серверах, использующих корневые DNS-серверы Интернета, практически одинаковы, но для серверов, использующих частные корневые DNS-серверы, они должны быть изменены. (Корневой DNS-сервер – это сервер, полномочный для корня соответствующего пространства имен).
db.127.0.0.1	Применяется при разрешении запросов на адрес типа обратной петли. Одинаков на всех серверах имен.

Имена файлов баз данных произвольны и определяются конфигурацией DNS-сервера. По умолчанию DNS-серверы в Windows 2008 используют не имена файлов, типичные для BIND, а файлы **имя_зоны.dns**.

Однако при перенесении файлов базы данных DNS с другого DNS-сервера, DNS-сервер Windows 2008 можно настроить на использование имен файлов BIND.

Устранение неполадок DNS

Для устранения неполадок DNS используется утилита командной строки **nslookup** и функции мониторинга и регистрации событий, доступные в оснастке DNS.

Основная диагностическая утилита службы DNS **nslookup** устанавливается вместе с TCP/IP и позволяет просматривать записи ресурсов и пересылать запросы любому серверу имен. У **nslookup** имеются два режима работы:

1. Интерактивный. Используется для получения нескольких блоков информации. Для запуска интерактивного режима следует запустить **nslookup** из командной строки без параметров. Для выхода из интерактивного режима следует набрать **exit**.

2. Неинтерактивный. **nslookup** запускается из командной строки с дополнительными параметрами:

```
nslookup [-ПАРАМЕТР ...] [ИСКОМЫЙ_КОМПЬЮТЕР | -[СЕРВЕР]]
```

где:

- ПАРАМЕТР – указывает один или несколько параметров. Для получения их полного перечня следует в интерактивном режиме набрать знак «?»;

- ИСКОМЫЙ_КОМПЬЮТЕР – если искомый компьютер представлен IP-адресом, nslookup вернет имя узла, а если именем – IP-адрес. Если компьютер представлен именем домена без завершающей точки, к имени добавляется имя домена DNS по умолчанию;

- СЕРВЕР – указывает сервер имен DNS. Если имя сервера опущено, используется сервер имен по умолчанию.

Практические задания

Постановка задачи

В данной лабораторной работе с использованием установленной в предыдущей работе виртуальной машины под управлением операционной системы Windows 2008 Server будет настроен DNS-сервер, будут созданы зоны прямого и обратного просмотра, а также ряд записей в этих зонах. Виртуальная машина под управлением операционной системы Windows XP Professional будет использоваться как DNS-клиент, и с ее помощью будет протестирована работоспособность развернутого DNS-сервера.

Задание NETBIOS-имени виртуальной машины под управлением Windows 2008 Server

На первом шаге настройки DNS-сервера необходимо задать NETBIOS-имя виртуальной машины под управлением Windows 2008 Server. Для этого в окне **Диспетчер сервера** необходимо выбрать ссылку **Изменить свойства системы** (рис. 50).

В открывшемся окне **Свойства системы** (рис. 51) следует нажать кнопку **Изменить**.

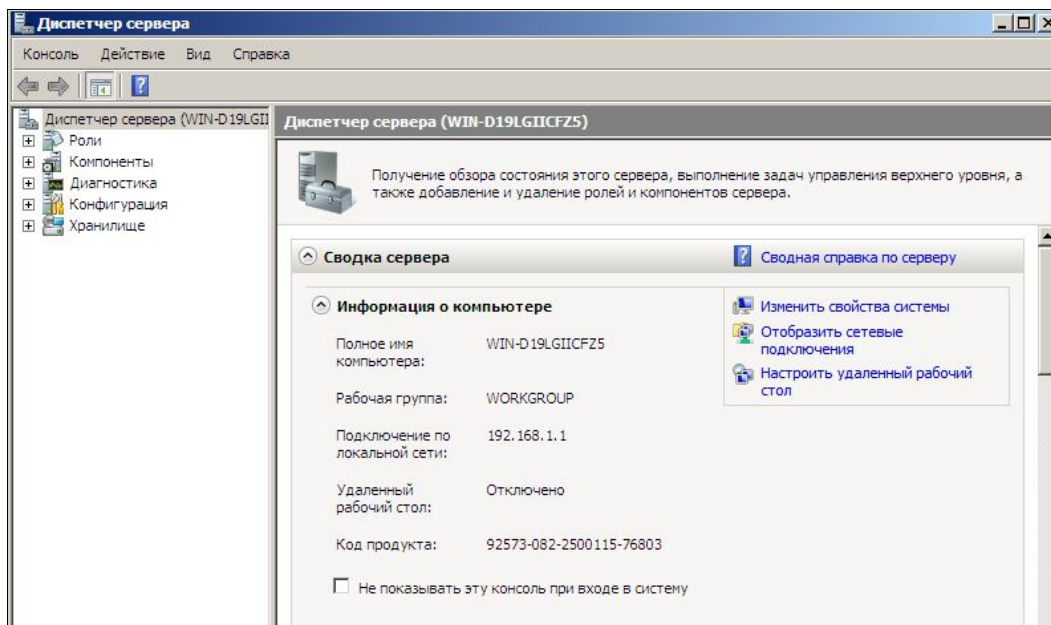


Рис. 50. Диалоговое окно Диспетчер сервера

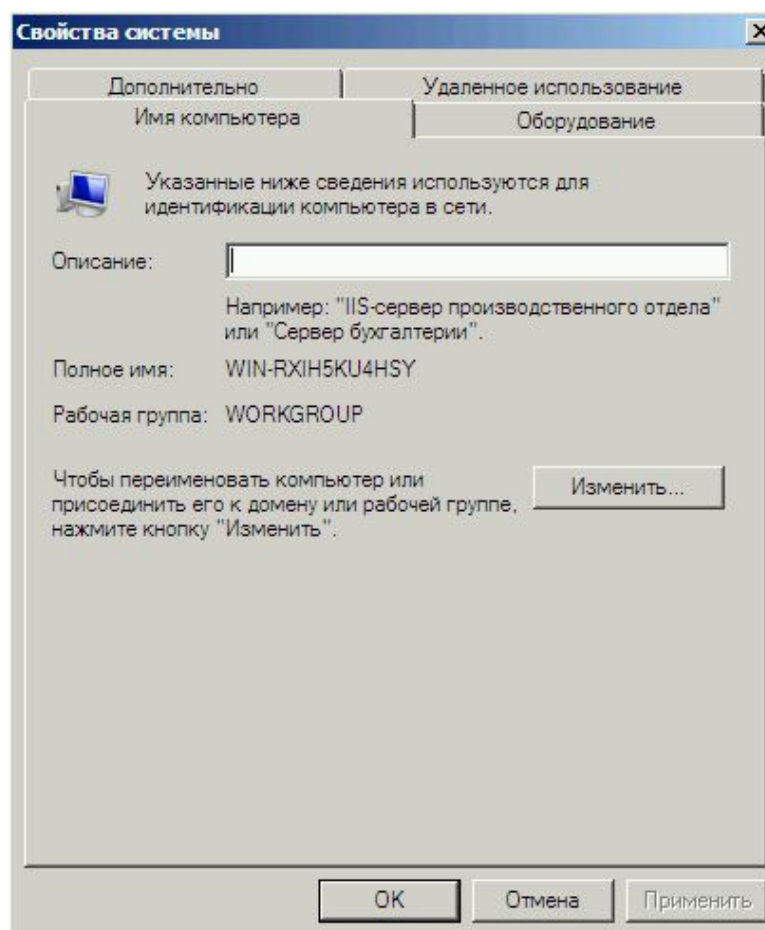


Рис. 51. Свойства системы

В поле **Имя компьютера** следует вписать **NETBIOS-имя компьютера**, например **Server**, указать, что данный компьютер входит в рабочую группу **WORKGROUP**, а затем нажать кнопку **ОК** (рис. 52).

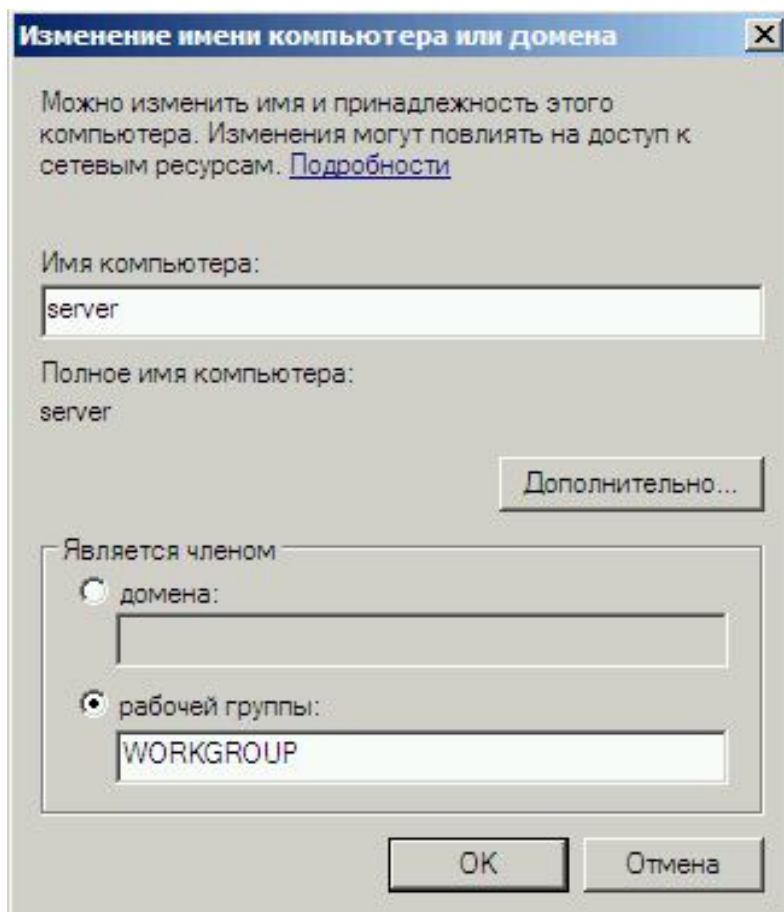


Рис. 52. Изменение имени компьютера

После изменения имени необходимо произвести перезагрузку виртуальной машины.

Добавление роли DNS-сервера на виртуальной машине под управлением Windows 2008 Server

Далее необходимо на виртуальной машине SERVER добавить роль **DNS-сервер**. Для установки службы DNS необходимо запустить **Диспетчер сервера**, выбрав значок рядом с кнопкой **Пуск** или в меню **Пуск**, верхняя строка (рис. 53).

На левой панели **Диспетчера сервера** необходимо выбрать пункт **Роли**, затем на правой панели нажать ссылку **Добавить роли** (рис. 54).

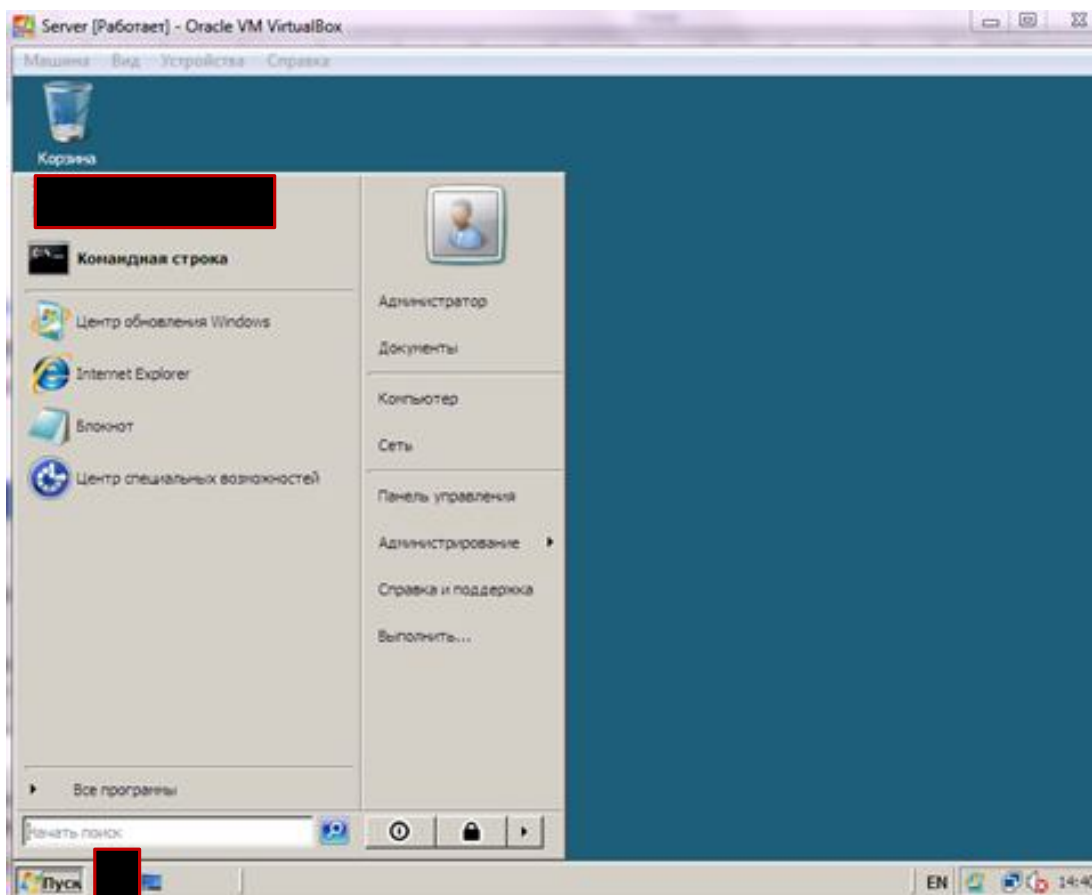


Рис. 53. Запуск Диспетчера сервера

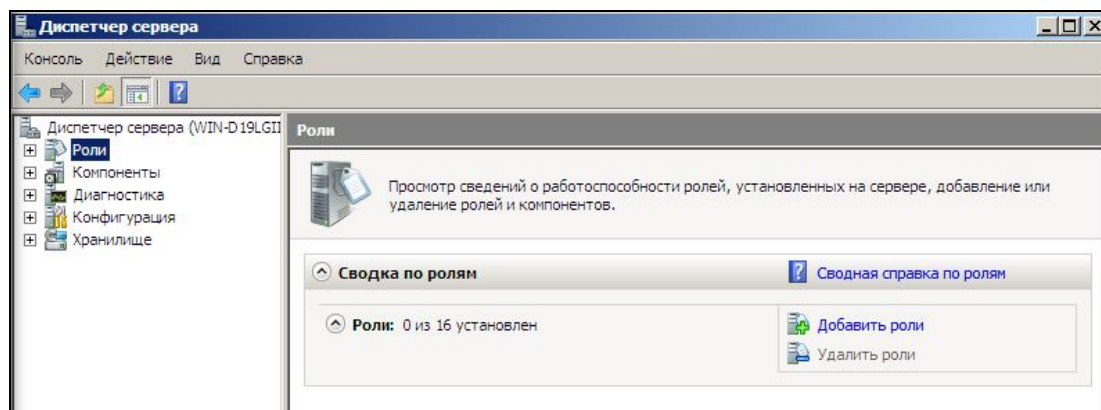


Рис. 54. Диалоговое окно Диспетчер сервера

Будет запущен **Мастер добавления ролей**, работа которого состоит из нескольких этапов. Первый этап – информационный, **Знакомство с DNS-сервером**. Следует ознакомиться с инструкциями и нажать кнопку **Далее** (рис. 55).

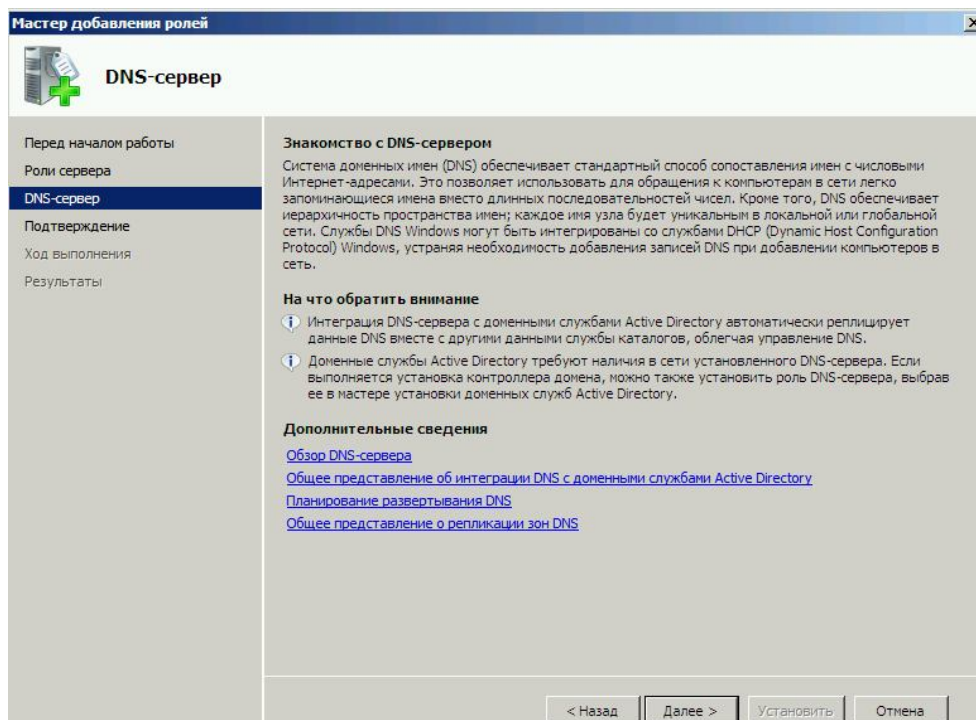


Рис. 55. Мастер добавления ролей – Знакомство с DNS-сервером

На следующем этапе необходимо ознакомиться со списком добавляемых ролей (DNS-сервер) и нажать кнопку **Установить** для начала установки этой роли (рис. 56).

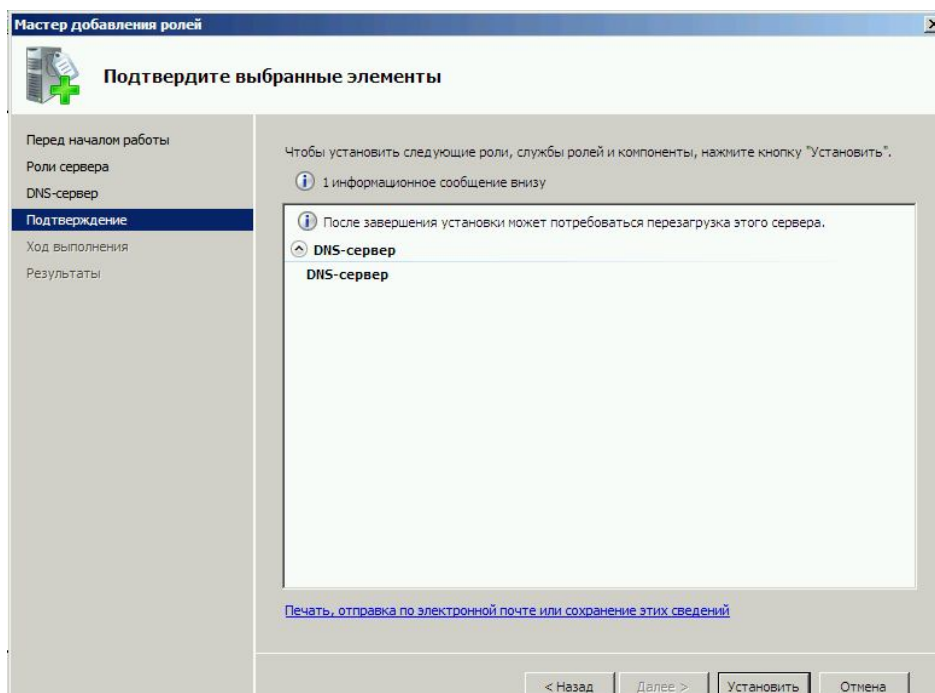


Рис. 56. Мастер добавления ролей – подтверждение установки выбранных элементов

Начнется процесс установки DNS-сервера (рис. 57).

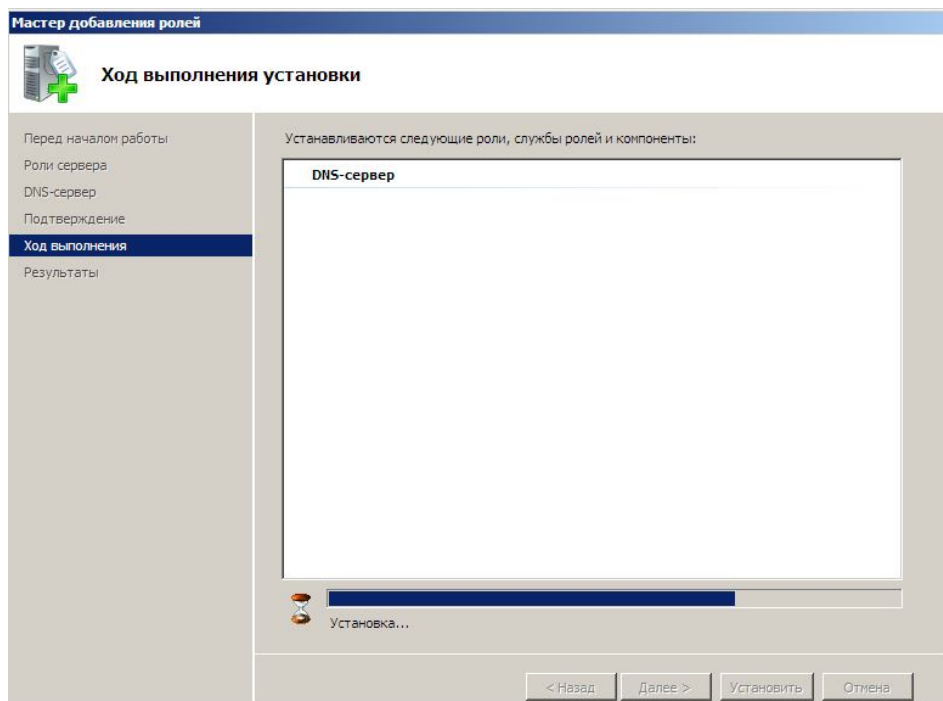


Рис. 57. Установка DNS-сервера

После завершения установки будет выдано предупреждение об отключенной службе автоматического обновления (рис. 58). После ознакомления с результатами установки следует нажать кнопку **Заккрыть**.

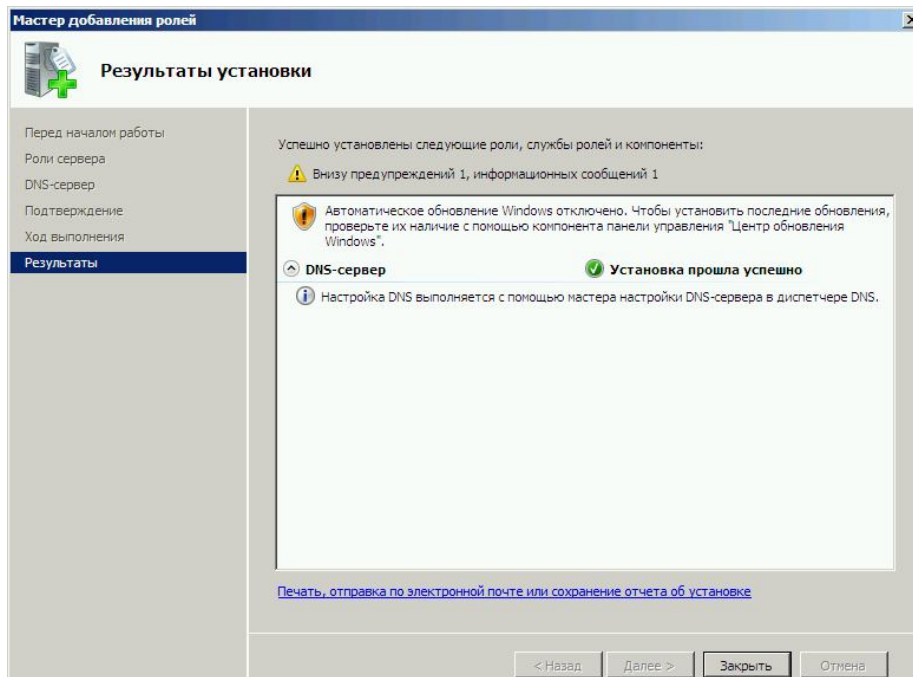


Рис. 58. Результаты установки

В диалоговом окне **Диспетчер сервера** появятся сведения о том, что в настоящее время сервер выполняет одну роль – DNS-сервер (рис. 59).

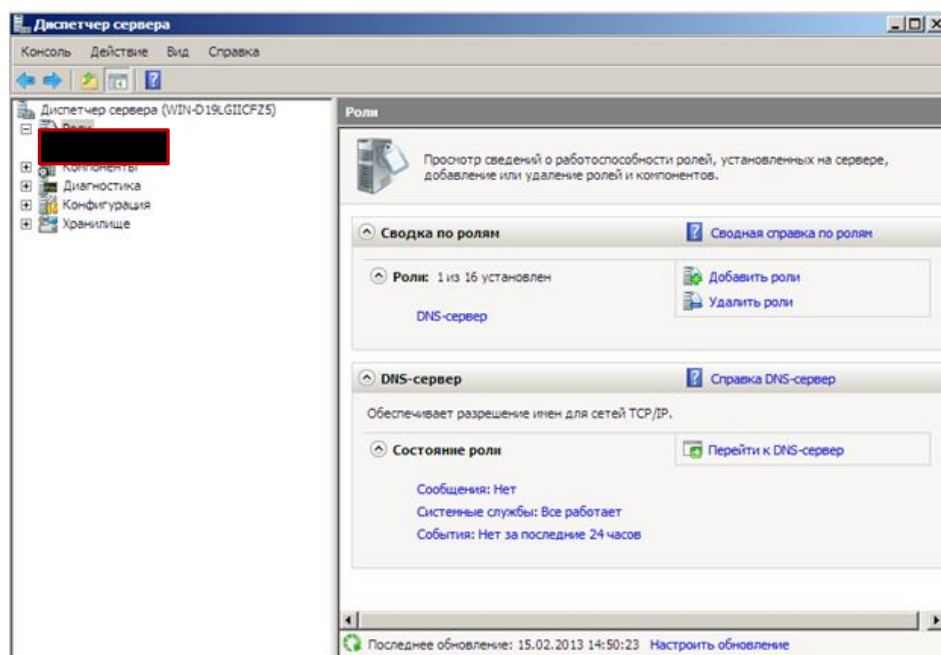


Рис. 59. Диспетчер сервера – Роли

Инструментарий администрирования DNS

Администрирование DNS-сервера осуществляется при помощи консоли **Диспетчер DNS**, доступной через меню **Пуск | Администрирование | DNS** (рис. 60).

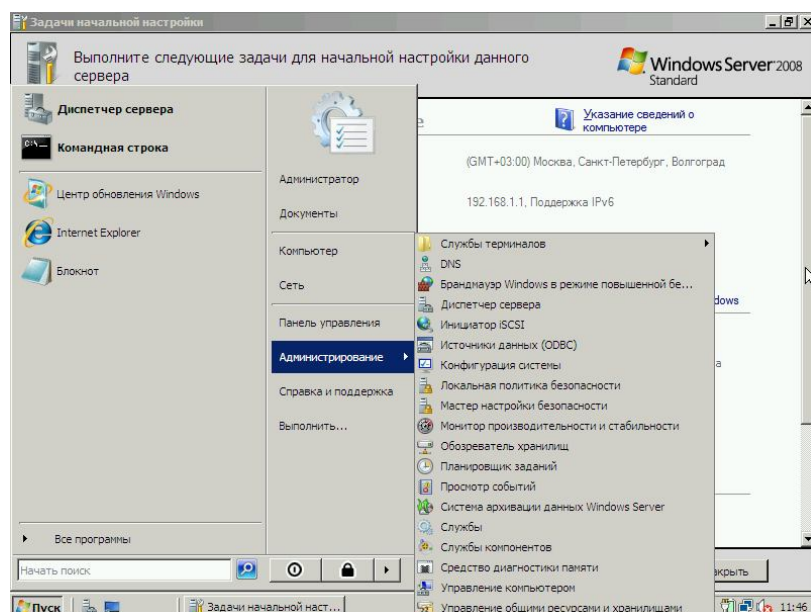


Рис. 60. Запуск Диспетчера DNS

Диспетчер DNS позволяет создавать зоны прямого и обратного просмотра, добавлять записи ресурсов в файл БД зоны и конфигурировать сервер DNS для использования технологии DDNS (Dynamic DNS), которая позволяет другим серверам и службам производить автоматические обновления зонных файлов (рис. 61).

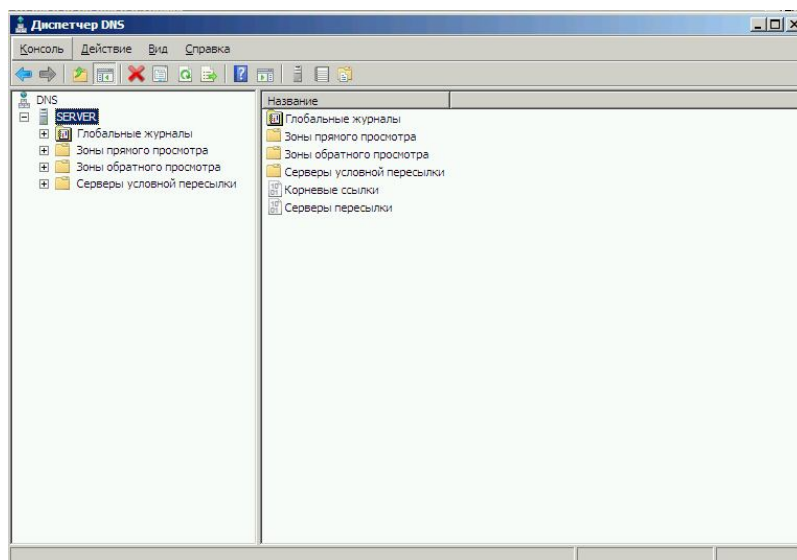


Рис. 61. Диспетчер DNS

Создание зоны прямого просмотра

Следующий этап настройки DNS-сервера – создание зоны прямого просмотра. Зона прямого просмотра позволяет обрабатывать запросы прямого просмотра. Для функционирования службы DNS необходима минимум одна зона прямого просмотра.

Для создания зоны прямого просмотра следует развернуть окно оснастки DNS, в дереве консоли раскрыть узел с именем установленного сервера (SERVER).

Далее необходимо выделить папку **Зоны прямого просмотра**, щелкнуть по ней правой кнопкой мыши и в контекстном меню выбрать пункт **Создать новую зону** (рис. 62).

Запустится Мастер создания новой зоны (рис. 63).

Мастер позволяет выбрать три параметра: тип зоны, имя зоны и файл зоны.

Возможно создание зон трех типов (рис. 64):

- основная зона;
- дополнительная зона;
- зона-заглушка.

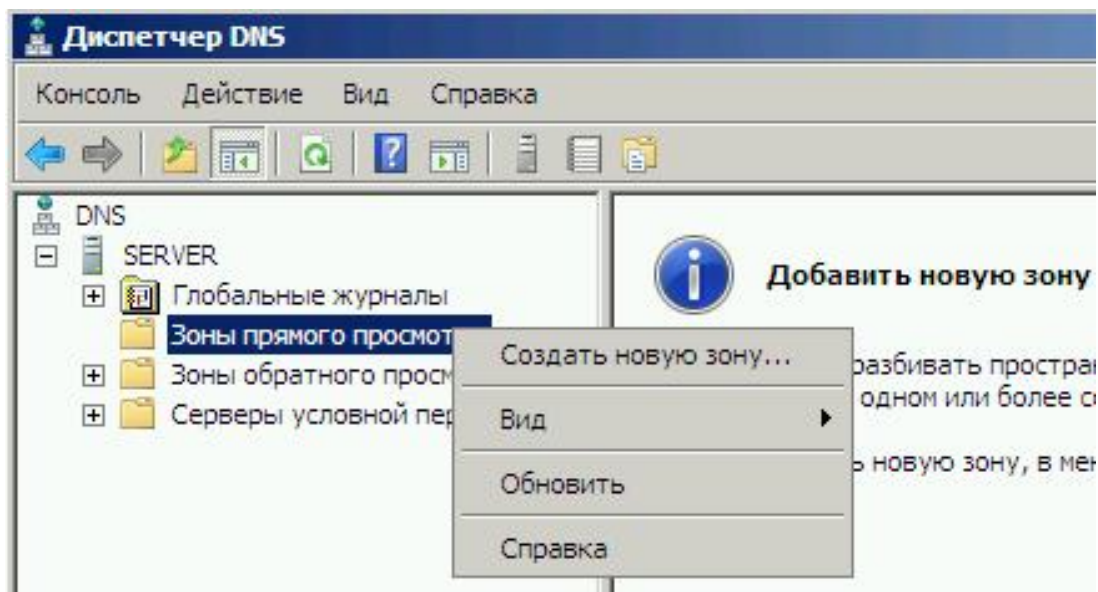


Рис. 62. Создание новой зоны прямого просмотра

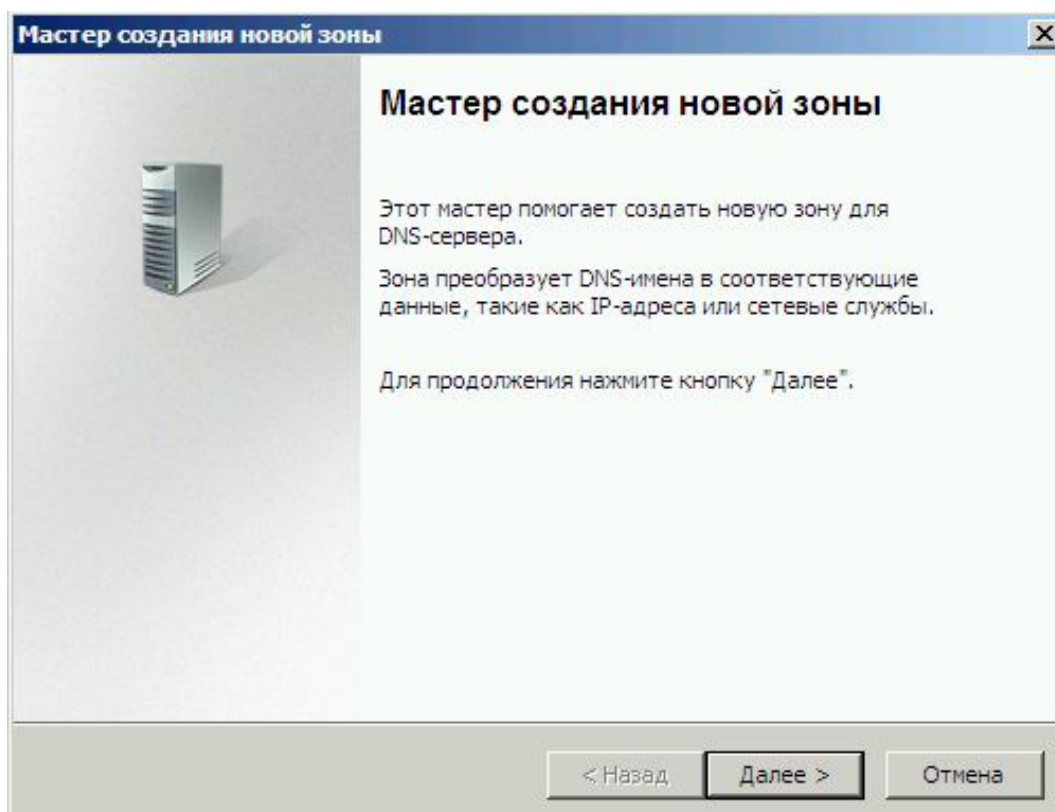


Рис. 63. Мастер создания новой зоны – начальный этап

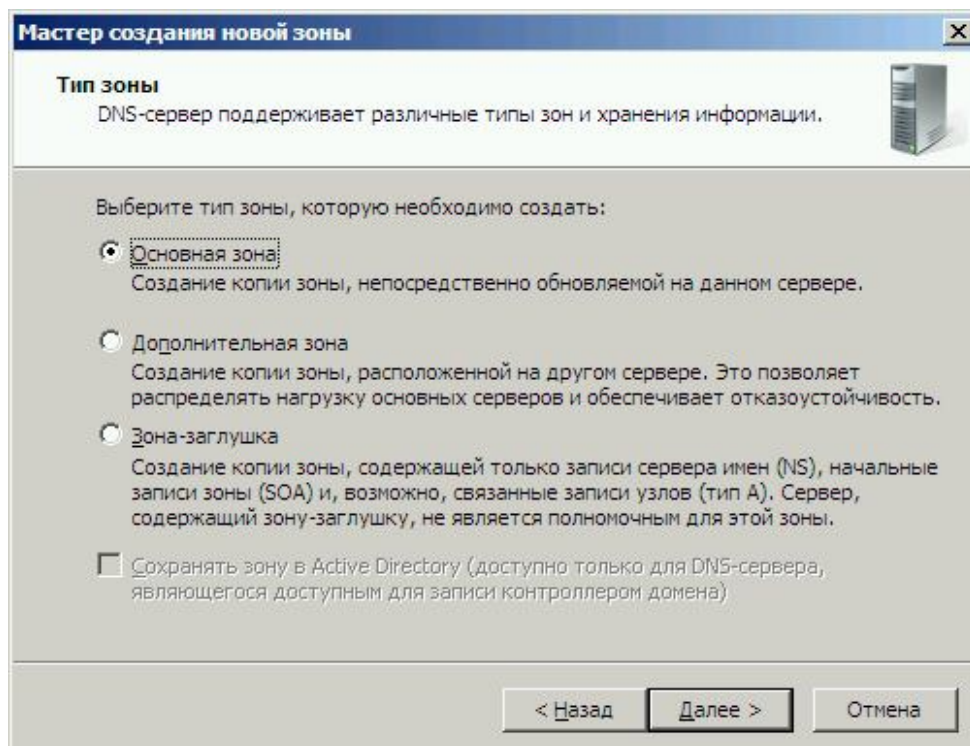


Рис. 64. Мастер создания новой зоны – Тип зоны

Следует выбрать переключатель **Основная** и щелкнуть кнопку **Далее**. Откроется окно выбора имени зоны (рис. 65).

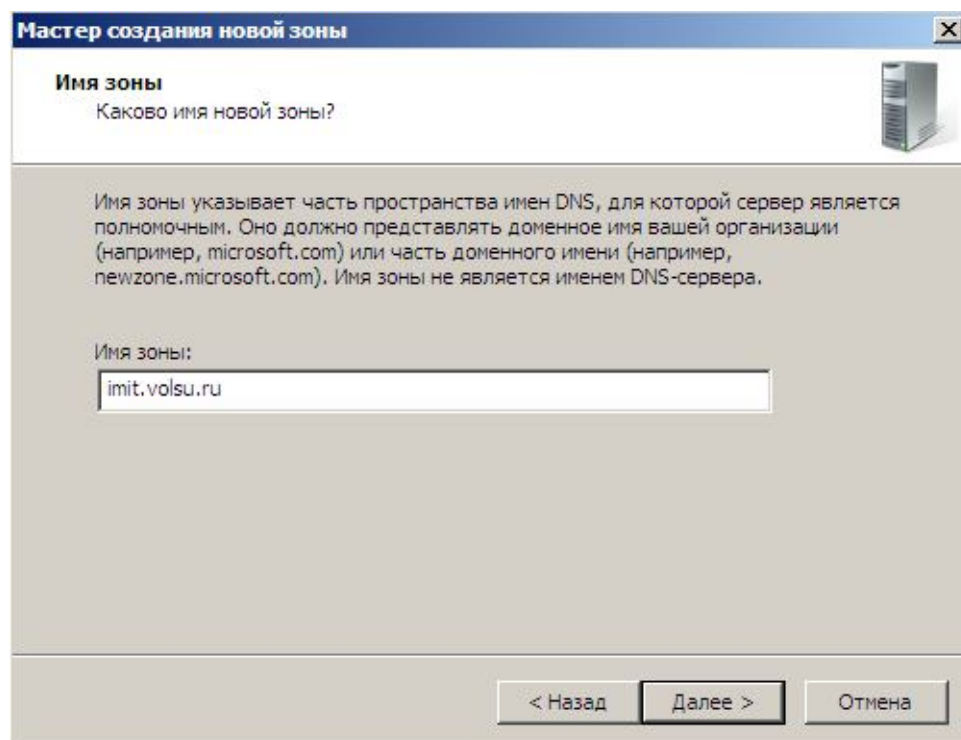


Рис. 65. Мастер создания новой зоны – Имя зоны

Необходимо указать имя новой зоны, например, `imit.volsu.ru`, щелкнуть кнопку **Далее**. Откроется окно выбора имени файла зоны (рис. 66). Необходимо выбрать переключатель **Создать новый файл** и оставить предлагаемое по умолчанию имя файла – `imit.volsu.ru/dns` – без изменений, после чего нажать кнопку **Далее**.

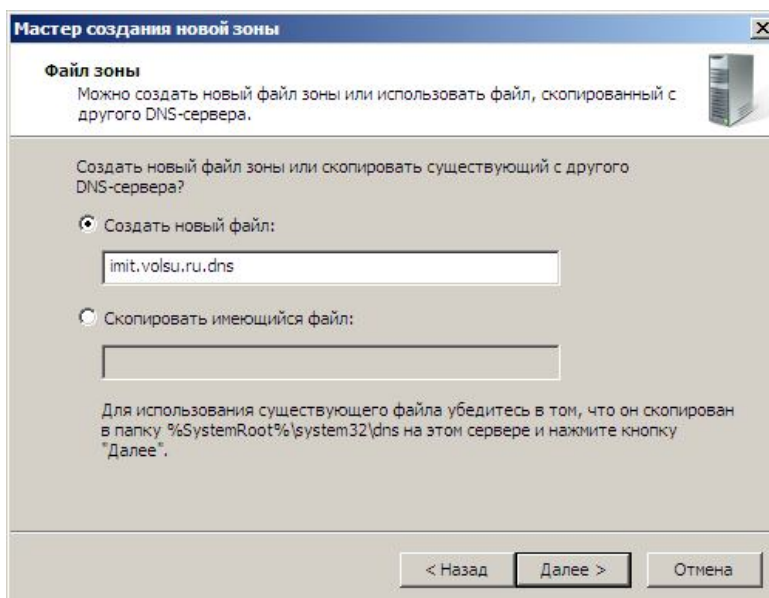


Рис. 66. Мастер создания новой зоны – Файл зоны

На следующем этапе работы **Мастера** производится настройка возможности проведения динамического обновления. В данном случае следует запретить динамические обновления (рис. 67).

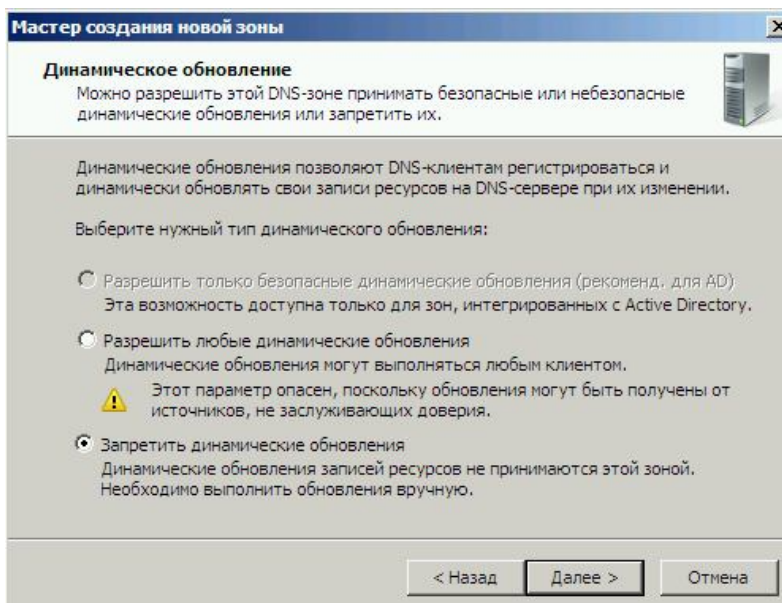


Рис. 67. Диалоговое окно Динамическое обновление

На последнем этапе работы **Мастера** отображаются выбранные настройки (рис. 68). Следует просмотреть сводку выбранных параметров и нажать кнопку **Готово**.

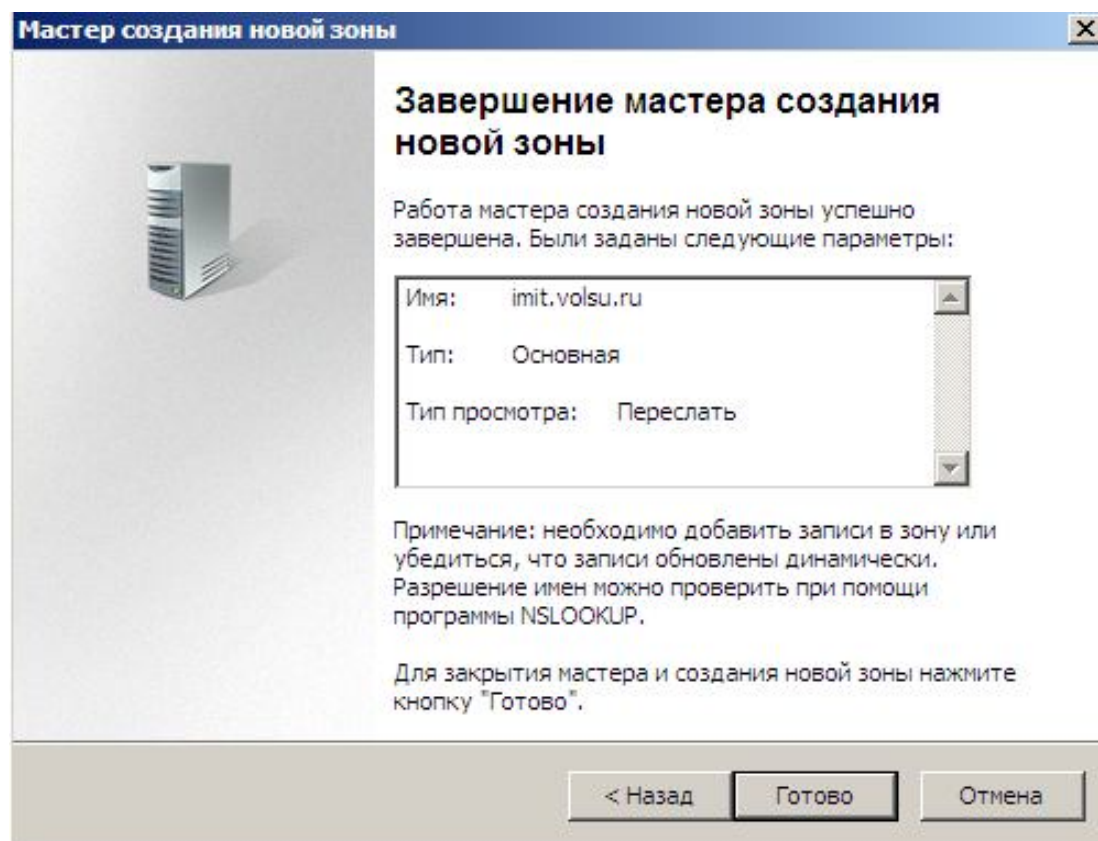


Рис. 68. Диалоговое окно **Завершение мастера создания новой зоны**

В результате будет создана зона прямого просмотра и сгенерированы записи **Начальная запись зоны (SOA)** и **Сервер имен (NS)** (рис. 69).

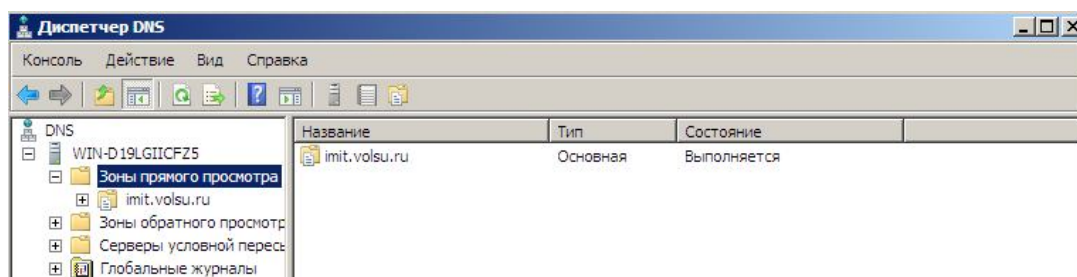


Рис. 69. Результаты создания новой зоны

Теперь сервер может разрешать имена хостов в IP-адреса, используя файл основной зоны просмотра.

Создание зоны обратного просмотра

Для создания зоны обратного просмотра в дереве консоли Диспетчер DNS необходимо выделить папку **Зоны обратного просмотра** и в контекстном меню выбрать пункт **Создать новую зону** (рис. 70).

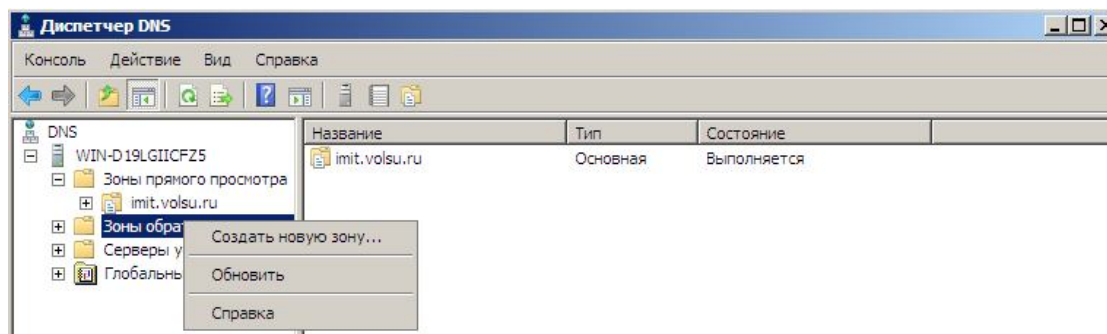


Рис. 70. Создание новой зоны обратного просмотра

Запустится **Мастер создания новой зоны** (рис. 71). Необходимо нажать кнопку **Далее**.

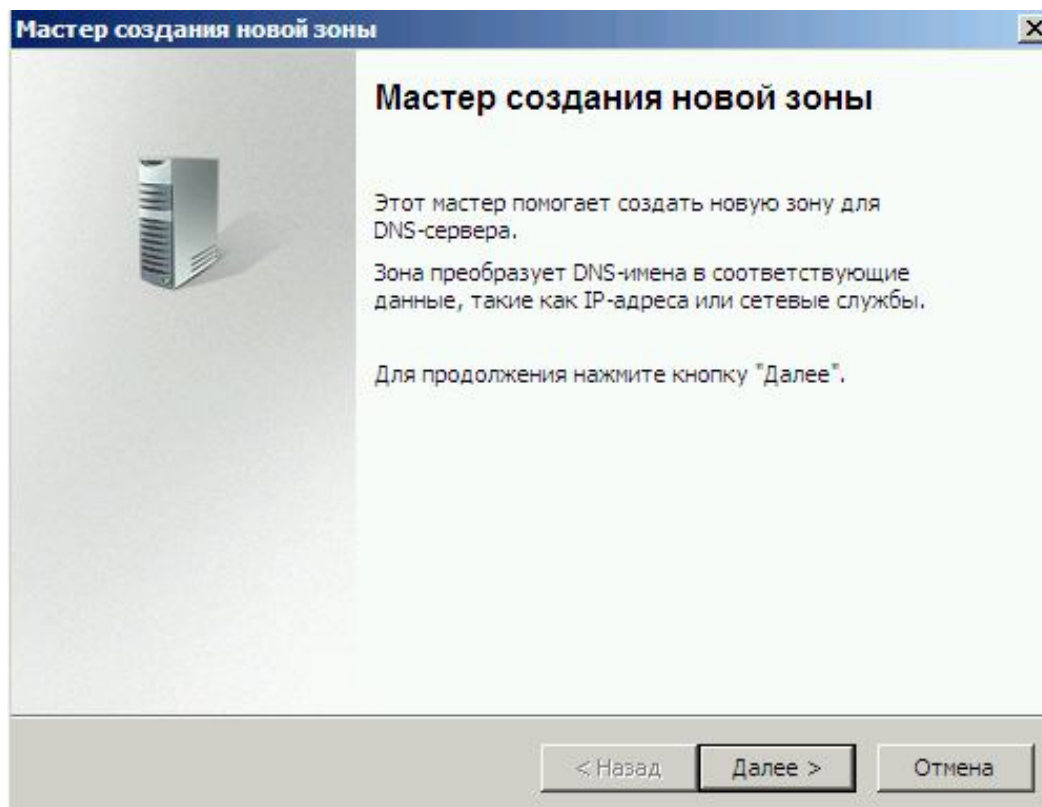


Рис. 71. Мастер создания новой зоны обратного просмотра

Затем следует выбрать тип создаваемой зоны. В данном случае необходимо выбрать вариант **Основная зона** и нажать кнопку **Далее** (рис. 72).

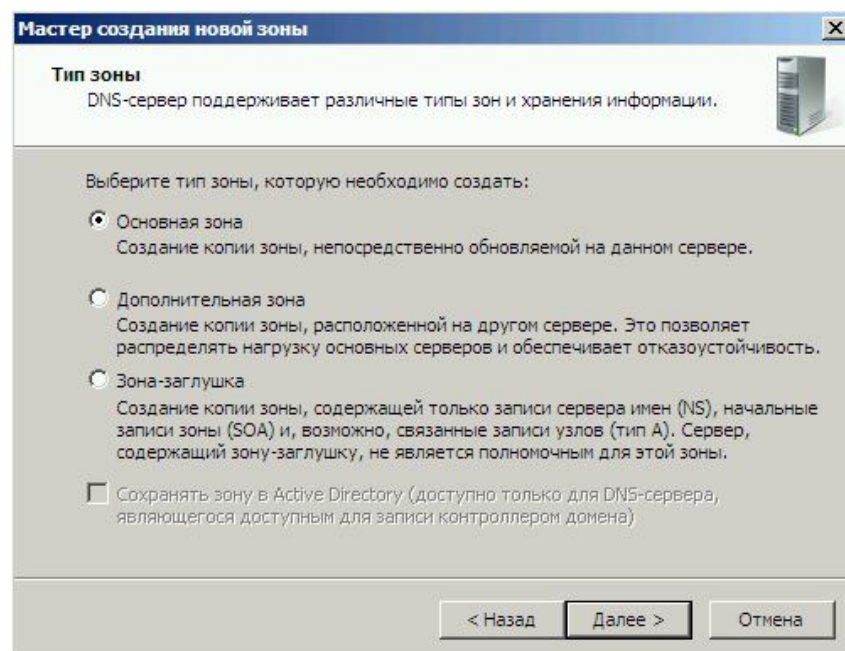


Рис. 72. Выбор типа новой обратной зоны

Далее необходимо выбрать версию протокола IP, для которой производится создание зоны обратного просмотра. В данном случае используется протокол IPv4 (рис. 73).

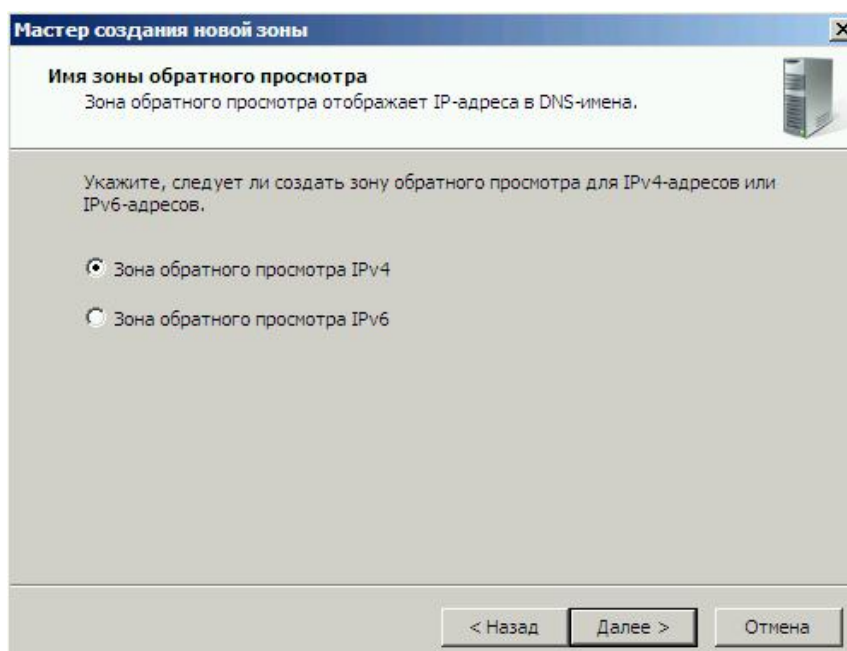


Рис. 73. Выбор протокола обратной зоны

На следующем этапе производится указание идентификатора сети или имени зоны обратного просмотра (рис. 74). Необходимо убедиться, что выбран переключатель **Идентификатор сети**. В поле под ним необходимо ввести адрес **192.168.1**. Поле **Имя зоны обратного просмотра** внизу окна должно выглядеть как **1.168.192.in-addr.arpa**.

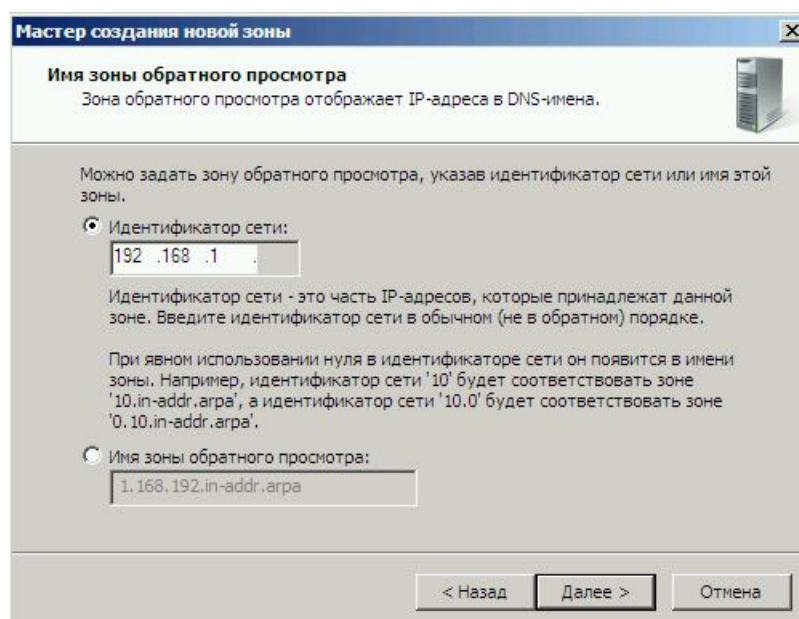


Рис. 74. Выбор имени зоны обратного просмотра

Далее необходимо указать имя файла, в котором будет храниться информация о зоне. Можно оставить предложенное по умолчанию имя файла (рис. 75) и нажать кнопку **Далее**.

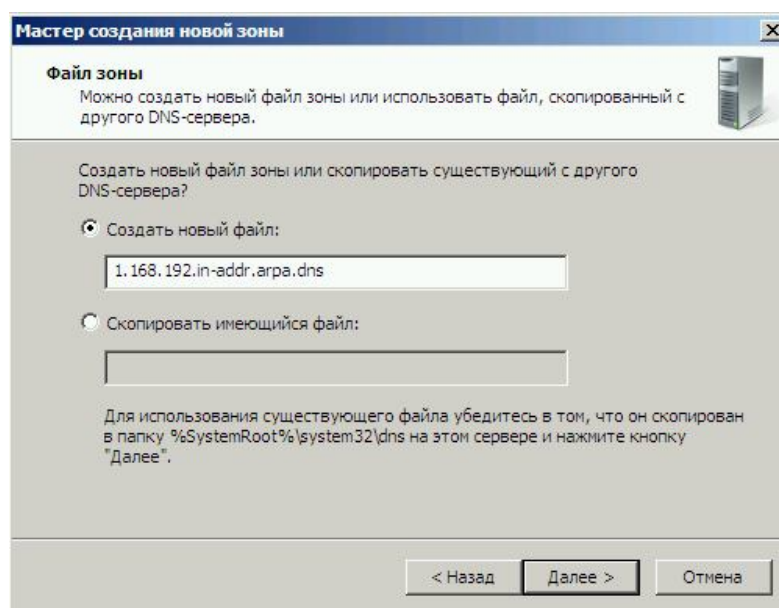


Рис. 75. Выбор имени файла обратной зоны

Следующий шаг работы **Мастера** служит для установки параметров динамического обновления. В данной лабораторной работе следует запретить динамические обновления (рис. 76).

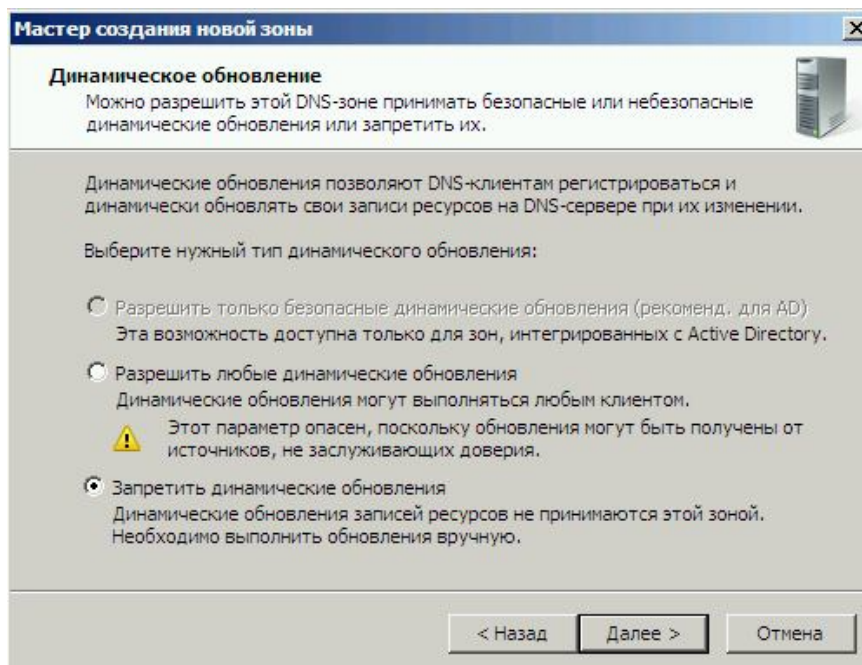


Рис. 76. Динамическое обновление

На последнем шаге работы **Мастера** можно проверить заданные параметры создаваемой зоны обратного просмотра (рис. 77) и нажать **Готово**.

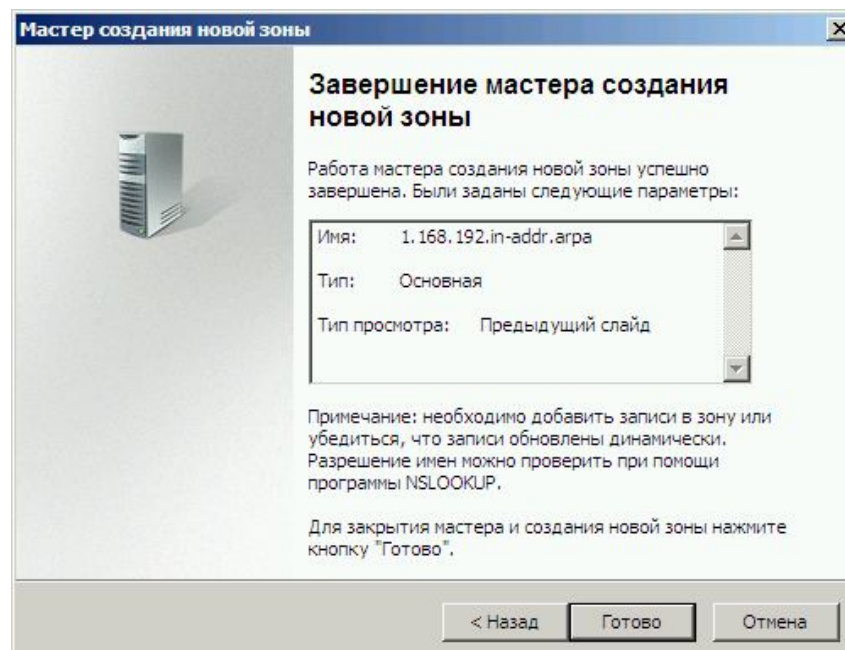


Рис. 77. Завершение мастера создания зоны обратного просмотра

В результате будет создана зона обратного просмотра с записями SOA и NS (рис. 78).

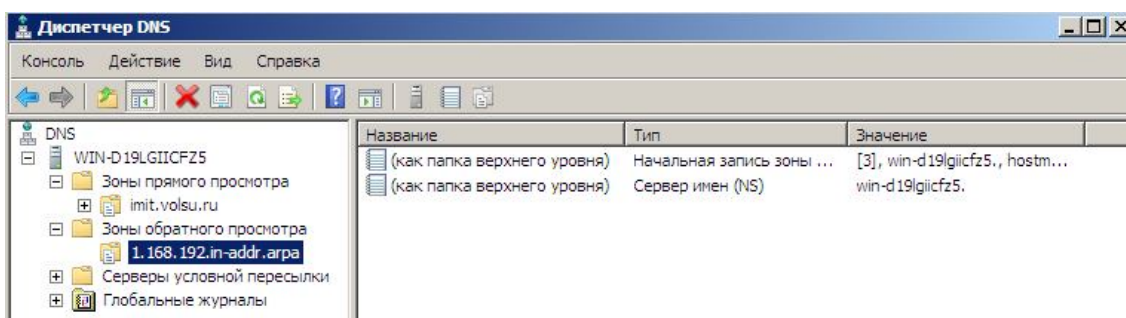


Рис. 78. Результаты создания зоны обратного просмотра

Создание записи узла в зоне прямого просмотра

Далее в новой зоне следует создать запись узла. Для этого необходимо щелкнуть правой кнопкой мыши по имени зоны прямого просмотра (**imit.volsu.ru**) и выбрать в контекстном меню пункт **Создать узел (A или AAAA)...** (рис. 79).

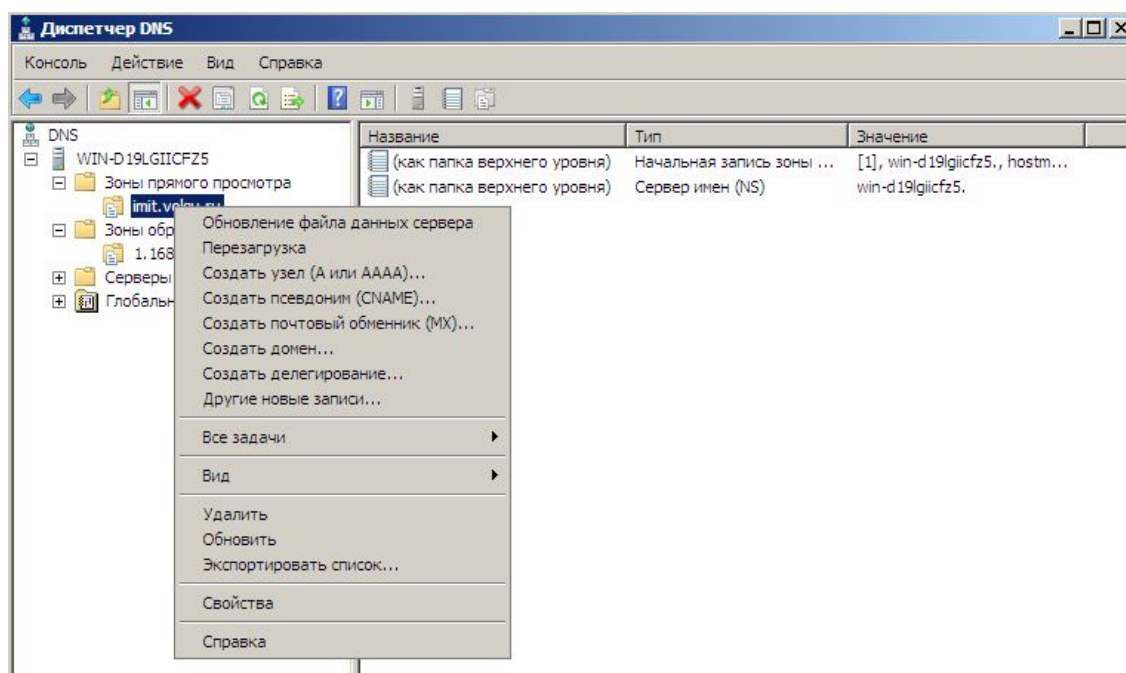


Рис. 79. Создание узла

Откроется диалоговое окно выбора параметров записи узла (рис. 80). В поле имени узла следует указать доменное имя создаваемой записи – имя хоста (SERVER) и IP-адрес сервера (192.168.1.1). Следует также установить флажок **Создать**

соответствующую PTR-запись для того, чтобы в зоне обратного просмотра была автоматически создана соответствующая PTR-запись.

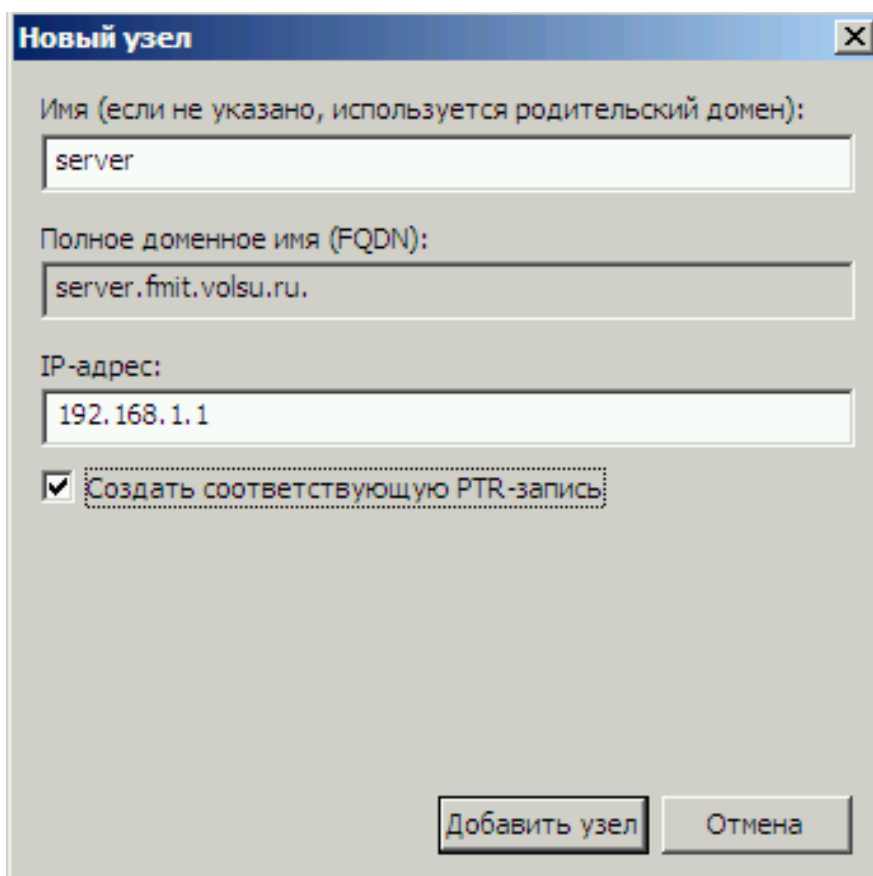


Рис. 80. Имя и адрес нового узла

После нажатия кнопки **Добавить узел** появится подтверждение об успешном создании записи узла (рис. 81).

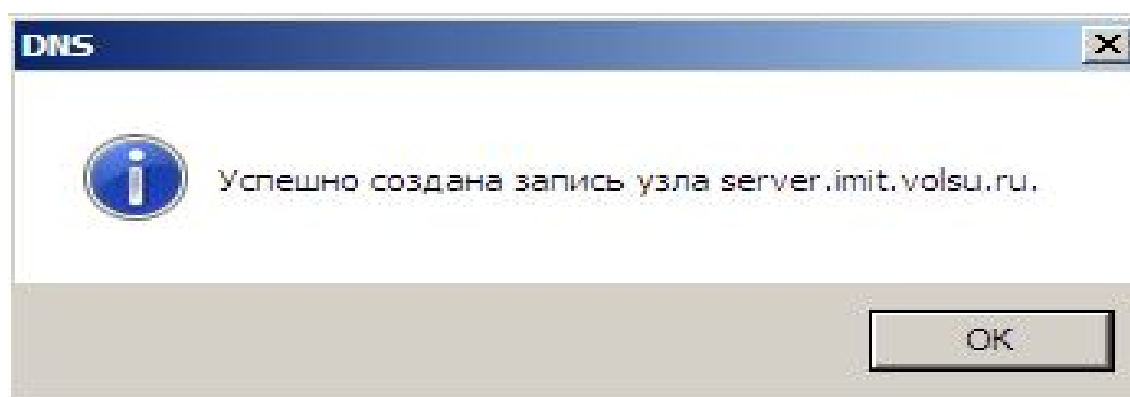


Рис. 81. Окно сообщения о добавлении узла

Аналогичным образом необходимо самостоятельно добавить запись узла CLIENT с IP-адресом 192.168.1.2 в зону прямого просмотра (рис. 82).

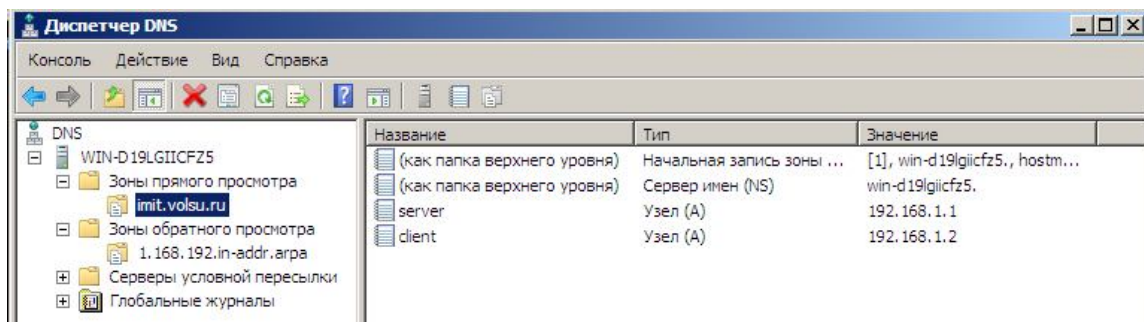


Рис. 82. Результат добавления двух узлов

Тестирование работоспособности службы DNS

Для проверки работоспособности зоны прямого просмотра в командной строке следует запустить утилиту **nslookup** в интерактивном режиме и произвести запрос на разрешение имени компьютера в IP-адрес, указав его FQDN (рис. 83). Аналогично тестируется способность сервера к обратному разрешению имен: указывается IP-адрес компьютера.

```

Администратор: Командная строка - nslookup
Microsoft Windows [Версия 6.0.6001]
(C) Корпорация Майкрософт, 2006. Все права защищены.

C:\Users\Администратор>nslookup
Default Server: server.imit.volsu.ru
Address: 192.168.1.1

> server.imit.volsu.ru
Server: server.imit.volsu.ru
Address: 192.168.1.1

Name: server.imit.volsu.ru
Address: 192.168.1.1

> client.imit.volsu.ru
Server: server.imit.volsu.ru
Address: 192.168.1.1

Name: client.imit.volsu.ru
Address: 192.168.1.2

> 192.168.1.2
Server: server.imit.volsu.ru
Address: 192.168.1.1

Name: client.imit.volsu.ru
Address: 192.168.1.2

> 192.168.1.1
Server: server.imit.volsu.ru
Address: 192.168.1.1

Name: server.imit.volsu.ru
Address: 192.168.1.1

>

```

Рис. 83. Разрешение имени в IP-адрес и IP-адреса в имя

Контрольные вопросы

1. Что представляет собой служба DNS? Для каких целей используется DNS? В чем преимущество использования доменных имен перед IP-адресами?
2. Дайте определение понятию «пространство доменных имен». Что включает в себя структура пространства доменных имен? Дайте определение понятию «DNS-домен». Для чего осуществляется деление на подмножества пространства имен DNS?
3. Опишите принцип формирования доменных имен и существующие ограничения.
4. Дайте определения понятию «зона». Чем зона отличается от домена? В чём различие между основной и дополнительной зонами?
5. Дайте определения понятиям DNS-сервера и DNS-интерпретатора. Опишите действия DNS-сервера при получении DNS-запроса. Чем основной сервер зоны отличается от дополнительного? Какие преимущества дает использование дополнительных DNS-серверов?
6. Для каких целей используются серверы кэширования и серверы пересылок? Какие DNS-серверы называются ведомыми? В каких режимах может использоваться сервер пересылок? В чем отличие между ними?
7. Что представляет собой процесс разрешения имен? В чем состоит отличие между прямым и обратным запросом на разрешение имен? Чем рекурсивный запрос на разрешение имени отличается от итеративного?
8. Что такое TTL в DNS? Какие типы зон прямого просмотра могут быть созданы?
9. Какая информация содержится в зоне прямого просмотра? Какие существуют типы зон прямого и обратного просмотра? Какое имя обычно присваивается зоне?
10. Какая информация содержится в зоне обратного просмотра? Для каких целей используется домен IN-ADDR.ARPA? Как формируется имя зоны обратного просмотра?
11. Дайте определение понятию «запись ресурса». Опишите формат записи ресурса.

12. Для каких целей используется запись ресурса SOA? Какая информация в ней содержится?

13. Для каких целей используются записи ресурсов NS, A и PTR?

14. Для каких целей используются записи ресурсов CNAME и MX?

15. Для каких целей используется запись ресурса SRV? Опишите ее формат. Приведите пример.

16. Для каких целей используются делегирующие и связывающие записи? Приведите пример.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
Лабораторная работа № 1. СТЕК ПРОТОКОЛОВ TCP/IP.....	4
ЦЕЛЬ РАБОТЫ.....	4
ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ.....	4
Архитектура TCP/IP	4
IP-адресация	7
Классы IP-адресов	7
Подсети и маски подсетей	8
Определение идентификатора сети	10
Создание надсетей и метод CIDR.....	10
Общие и частные адреса	12
Команда тестирования настроек сетевого интерфейса.....	14
Команда тестирования работоспособности сетевого соединения между двумя узлами ping	16
ПРАКТИЧЕСКИЕ ЗАДАНИЯ.....	18
Постановка задачи.....	18
Создание новой виртуальной машины.....	18
Установка Windows 2008 Server	23
Установка Windows XP Professional.....	29
Настройка статического IP-адреса сетевого интерфейса виртуальной машины под управлением Windows XP Professional.....	32
Отключение брандмауэра в Windows XP	34
Выбор типа сетевого подключения виртуальной машины.....	35
Тестирование параметров сетевого интерфейса виртуальной машины под управлением Windows XP	36
Настройка статического IP-адреса сетевого интерфейса виртуальной машины под управлением Windows 2008 Server	37
Отключение брандмауэра в Windows 2008 Server	40
Тестирование параметров сетевого интерфейса виртуальной машины под управлением Windows 2008 Server.....	41
Проверка работоспособности сетевого соединения между виртуальными машинами	42
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	43
Лабораторная работа № 2. СЛУЖБА DNS.....	44
ЦЕЛИ РАБОТЫ	44
ТЕОРЕТИЧЕСКОЕ ВВЕДЕНИЕ.....	44
Назначение службы DNS. Преимущества доменных имен над IP-адресами.....	44

Пространство доменных имен	45
Правила именования доменов	47
Зоны	47
Серверы имен DNS.....	49
Серверы кэширования.....	51
Серверы пересылок и ведомые серверы.....	51
Процесс разрешения имен	52
Рекурсивные и итеративные запросы.....	53
Кэширование и TTL	55
Отрицательное кэширование.....	55
Зоны прямого просмотра	55
Зоны обратного просмотра	56
Записи ресурсов и зоны.....	58
Формат записей ресурсов.....	59
Типы записей ресурсов	59
Запись ресурса SOA	60
Запись ресурса NS	61
Запись ресурса A	61
Запись ресурса PTR.....	62
Запись ресурса CNAME	62
Запись ресурса MX.....	62
Запись ресурса SRV	63
Делегирующие и связывающие записи	65
Хранение зон	65
Устранение неполадок DNS.....	66
ПРАКТИЧЕСКИЕ ЗАДАНИЯ.....	67
Постановка задачи.....	67
Задание NETBIOS-имени виртуальной машины под управлением Windows 2008 Server.....	67
Добавление роли DNS-сервера на виртуальной машине под управлением Windows 2008 Server.....	69
Инструментарий администрирования DNS	73
Создание зоны прямого просмотра	74
Создание зоны обратного просмотра	79
Создание записи узла в зоне прямого просмотра	83
Тестирование работоспособности службы DNS.....	85
КОНТРОЛЬНЫЕ ВОПРОСЫ.....	86

Учебное издание

Полубоярова Наталья Михайловна, **Полубояров** Валерий Викторович

ОПЕРАЦИОННЫЕ СИСТЕМЫ

*Учебно-методическое пособие к лабораторному практикуму
для бакалавров, обучающихся по направлениям подготовки
230100.62 Информатика и вычислительная техника,
230400.62 Информационные системы и технологии,
231000.62 Программная инженерия, 230700.62 Прикладная информатика,
010500.62 Математическое обеспечение
и администрирование информационных систем*

Главный редактор *А.В. Шестакова*
Техническое редактирование *О.С. Кашиук*
Оформление обложки *Н.Н. Захаровой*

Печатается в авторской редакции.

Подписано в печать 27.05 2013 г. Формат 60×84/16.
Бумага офсетная. Гарнитура Таймс. Усл. печ. л. 5,35. Уч.-изд. л. 5,75.
Тираж 100 экз. (1-й завод 1–50 экз.). Заказ . «С» 64.

Издательство Волгоградского государственного университета.
400062 Волгоград, просп. Университетский, 100.
E-mail: izvolgu@volsu.ru