

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ КРАСНОЯРСКОГО КРАЯ**  
**КРАЕВОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**  
**ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**  
**«КРАСНОЯРСКИЙ КОЛЛЕДЖ ОТРАСЛЕВЫХ ТЕХНОЛОГИЙ**  
**И ПРЕДПРИНИМАТЕЛЬСТВА»**

**РАССМОТРЕНО**

методической комиссией  
протокол № 6 от «20» июня 2024 г.

**УТВЕРЖДЕНО**

Директор КГБПОУ «Красноярский колледж  
отраслевых технологий и  
предпринимательства»

\_\_\_\_\_/Н. В. Журова/  
Приказ № 01-60-2П от «01» июля 2024 г.

**АДАПТИРОВАННАЯ ПРОГРАММА ПОДГОТОВКИ**  
**СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА**

09.02.07 Информационные системы и программирование

*на базе среднего общего образования*

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**  
**К ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ И ЛАБОРАТОРНЫХ РАБОТ**

**ПМ.07 Соадминистрирование и автоматизация баз данных и серверов**

Зам. директора по УР \_\_\_\_\_ / Е.В. Миля /  
подпись

Красноярск 2024

Организация-разработчик: краевое государственное бюджетное профессиональное образовательное учреждение «Красноярский колледж отраслевых технологий и предпринимательства»

Разработчик:

Швецова Наталья Ярославовна, преподаватель КГБПОУ «Красноярский колледж отраслевых технологий и предпринимательства»

## СОДЕРЖАНИЕ

### СОДЕРЖАНИЕ 3

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА 4

2 ОБЩИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ И ОФОРМЛЕНИЮ  
ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ 5

3 МЕТОДИКА ПРОВЕДЕНИЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ И ЛАБОРАТОРНЫХ  
РАБОТ 6

4 СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ 7

## 1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические указания к проведению практических занятий и лабораторных работ по МДК 07.01 Управление и автоматизация баз данных и МДК 07.02 Сертификация информационных систем предназначены для обучающихся СПО по специальности **09.02.07 «Информационные системы и программирование»**

Уровень профессиональной подготовки по специальности **09.02.07 «Информационные системы и программирование»**, определяемый ФГОС СПО, предусматривает владение практическими навыками выбора материалов для профессиональной деятельности.

Формируемые компетенции, реализуемые в процессе выполнения практических занятий и лабораторных работ:

|                                |                                                                                                                                                                                                                                                                                                                             |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Иметь практический опыт</b> | В участии в соадминистрировании серверов; разработке политики безопасности SQL сервера, базы данных и отдельных объектов базы данных; применении законодательства Российской Федерации в области сертификации программных средств информационных технологий                                                                 |
| <b>уметь</b>                   | проектировать и создавать базы данных; выполнять запросы по обработке данных на языке SQL; осуществлять основные функции по администрированию баз данных; разрабатывать политику безопасности SQL сервера, базы данных и отдельных объектов базы данных; владеть технологиями проведения сертификации программного средства |
| <b>знать</b>                   | модели данных, основные операции и ограничения; технологию установки и настройки сервера баз данных; требования к безопасности сервера базы данных; государственные стандарты и требования к обслуживанию баз данных                                                                                                        |

Выполнение лабораторных работ способствует формированию и развитию общих и профессиональных компетенций:

| <b>Код</b> | <b>Наименование общих компетенций</b>                                                                                                                                                 |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ОК 1       | Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам                                                                                    |
| ОК 2       | Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности                                                                 |
| ОК 3       | Планировать и реализовывать собственное профессиональное и личностное развитие                                                                                                        |
| ОК 4       | Планировать и реализовывать собственное профессиональное и личностное развитие                                                                                                        |
| ОК 5       | Планировать и реализовывать собственное профессиональное и личностное развитие                                                                                                        |
| ОК 6       | Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей                                                   |
| ОК 7       | Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях                                                                         |
| ОК 8       | Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности |
| ОК 9       | Использовать информационные технологии в профессиональной деятельности                                                                                                                |
| ОК 10      | Пользоваться профессиональной документацией на государственном и иностранном языке                                                                                                    |
| ОК 11      | Планировать предпринимательскую деятельность в профессиональной сфере                                                                                                                 |

| <b>Код</b> | <b>Наименование профессиональных компетенций</b>                                                                                           |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| ВД 7       | <i><b>Соадминистрирование баз данных и серверов</b></i>                                                                                    |
| ПК 7.1     | Выявлять технические проблемы, возникающие в процессе эксплуатации баз данных и серверов                                                   |
| ПК 7.2     | Осуществлять администрирование отдельных компонент серверов                                                                                |
| ПК 7.3     | Формировать требования к конфигурации локальных компьютерных сетей и серверного оборудования, необходимые для работы баз данных и серверов |
| ПК 7.4     | Осуществлять администрирование баз данных в рамках своей компетенции                                                                       |
| ПК 7.5     | Проводить аудит систем безопасности баз данных и серверов, с использованием регламентов по защите информации.                              |

Формируемые личностные результаты в ходе освоения общеобразовательной дисциплины: ЛР 04, ЛР 07, ЛР 10, ЛР 15, ЛР 16, ЛР 17.

## **2 ОБЩИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ И ОФОРМЛЕНИЮ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ**

Лабораторные (практические) работы выполняются обучающимися по графику, составленному в соответствии с рабочей программой МДК 07.01 Управление и автоматизация баз данных и МДК 07.02 Сертификация информационных систем.

МДК 07.01 Управление и автоматизация баз данных и МДК 07.02 Сертификация информационных систем зависит от содержания лабораторных (практических) работ, которые соответствуют более глубокому освоению дисциплины, закреплению теоретических знаний и прививают обучающимся практические навыки самостоятельной работы.

Задача лабораторных работ (практических занятий) – закрепить теоретические знания обучающихся.

Согласно учебного плана по специальности и программы учебной дисциплины на лабораторные (практические) занятия обучающихся выделено 54 академических часа, из них:

| <b>№ и наименование разделов (тем)</b>                                            | <b>Количество часов</b> |
|-----------------------------------------------------------------------------------|-------------------------|
| <i><b>Раздел 1. Технологии администрирования серверов и баз данных</b></i>        | 36                      |
| <i><b>Раздел 2. Обеспечение качества и сертификация информационных систем</b></i> | 18                      |
| <b>ИТОГО:</b>                                                                     | 54                      |

### 3 МЕТОДИКА ПРОВЕДЕНИЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ И ЛАБОРАТОРНЫХ РАБОТ

Целью практических занятий является приобретение практических навыков работы с базами данных, а именно принципов построения и администрирования баз данных, организация серверов баз данных, защиты и сохранности информации баз данных, сертификации информационных систем.

Целью лабораторных работ является отработка обучающимися практических навыков по построению и администрированию баз данных, организации работы серверов баз данных, проведению сертификации информационных систем.

Исходя, из поставленных целей в работе будут решаться следующие задачи:

***Закрепление знаний*** по:

- ✓ моделям данных,
- ✓ основным операциям и ограничениям;
- ✓ технологии установки и настройки сервера баз данных;
- ✓ требованиям к безопасности сервера базы данных;
- ✓ государственным стандартам и требованиям к обслуживанию баз данных

***Ознакомиться:***

- ✓ с проектированием и созданием базы данных;
- ✓ с выполнением запросов по обработке данных на языке SQL;
- ✓ с осуществлением основных функции по администрированию баз данных;
- ✓ с разработкой политики безопасности SQL сервера, базы данных и отдельных объектов

базы данных;

- ✓ с технологиями проведения сертификации программного средства;

***При выполнении лабораторной работы формируются навыки:***

- ✓ участия в соадминистрировании серверов;
- ✓ в разработке политики безопасности SQL сервера, базы данных и отдельных объектов базы

данных;

- ✓ применения законодательства Российской Федерации в области сертификации

программных средств информационных технологий.

***Научиться пользоваться:***

- ✓ языком SQL;
- ✓ законодательством Российской Федерации в области сертификации программных средств информационных технологий.

## 4 СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ

### Раздел 1. Технологии администрирования серверов и баз данных

#### МДК. 07.01 Управление и автоматизация баз данных

#### Тема 1.1. Принципы построения и администрирования баз данных

##### Лабораторное занятие № 1

Тема: «Построение схемы базы данных» (3 часа)

Цель:

1. изучение общих принципов проектирования реляционных моделей данных, знакомство с основами реляционного исчисления.

#### ИНСТРУКЦИОННАЯ КАРТА

##### Материально - техническое оснащение

Оборудование: ПК, Windows, MS Access

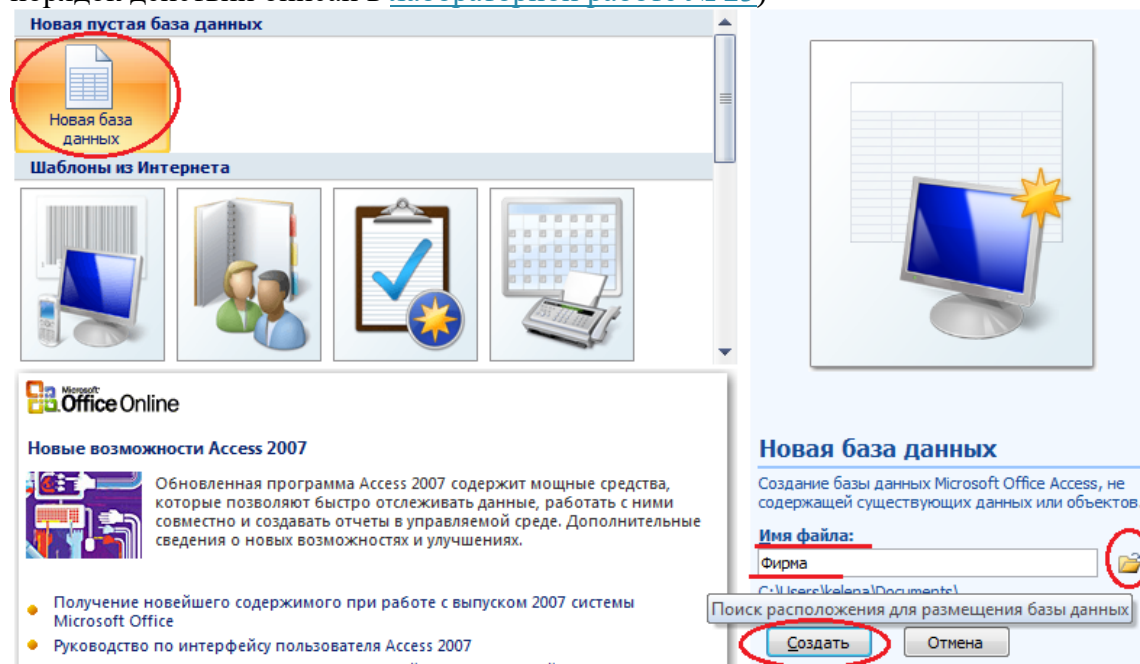
Характер выполнения работы: обучающиеся выполняют работу индивидуально.

##### Практические задания

1. Запустите Microsoft Access.

2. Создайте базу данных **Фирма**. Сотрудники данной организации работают с клиентами и выполняют их заказы.

Для этого запустите **Access**, создайте **новую базу данных**, задайте **имя** базы данных **Фирма**, выберите **место сохранения** Вашей базы данных (диск X:) и нажмите кнопку **Создать**. (подробный порядок действий описан в [лабораторной работе № 23](#))



3. Создайте в режиме Конструктор 3 таблицы: **Сотрудники**, **Клиенты** и **Заказы**. Если все сведения поместить в одной таблице, то она станет очень неудобной для работы. В ней начнутся повторы данных. Всякий раз, когда сотрудник Иванов будет работать с какой-либо фирмой, придется прописывать данные о сотруднике и клиенте заново, в результате чего можно допустить множество ошибок. Чтобы уменьшить число ошибок, можно исходную таблицу разбить на несколько таблиц и установить связи между ними. Это будет более рационально.

**Таблица Сотрудники**

| Работа с таблицами   |                   |                  |
|----------------------|-------------------|------------------|
| Главная              | Создание          | Внешние данные   |
| Режим                | Ключевое поле     | Построитель      |
| Режимы               | Сервис            | Проверка условий |
| Вставить строки      |                   | Удалить строки   |
| Страница свойств     |                   | Индексы          |
| Показать или скрыть  |                   |                  |
| Все таблицы          | сотрудники        |                  |
| сотрудники           | Имя поля          |                  |
| сотрудники : таблица | Тип данных        |                  |
| клиенты              |                   |                  |
| клиенты : таблица    |                   |                  |
| заказы               |                   |                  |
| заказы : таблица     |                   |                  |
|                      | Код сотрудника    | Счетчик          |
|                      | Фамилия           | Текстовый        |
|                      | Имя               | Текстовый        |
|                      | Отчество          | Текстовый        |
|                      | Должность         | Текстовый        |
|                      | Телефон           | Текстовый        |
|                      | Адрес             | Текстовый        |
|                      | Дата рождения     | Дата/время       |
|                      | Зарботная плата   | Денежный         |
|                      | Фото              | Поле объекта OLE |
|                      | Электронная почта | Гиперссылка      |

Таблица Клиенты

| Работа с таблицами   |                   |                  |
|----------------------|-------------------|------------------|
| Главная              | Создание          | Внешние данные   |
| Режим                | Ключевое поле     | Построитель      |
| Режимы               | Сервис            | Проверка условий |
| Вставить строки      |                   | Удалить строки   |
| Страница свойств     |                   | Индексы          |
| Показать или скрыть  |                   |                  |
| Все таблицы          | клиенты           |                  |
| сотрудники           | Имя поля          |                  |
| сотрудники : таблица | Тип данных        |                  |
| клиенты              |                   |                  |
| клиенты : таблица    |                   |                  |
| заказы               |                   |                  |
| заказы : таблица     |                   |                  |
|                      | Код клиента       | Счетчик          |
|                      | Название компании | Текстовый        |
|                      | Адрес             | Текстовый        |
|                      | Номер телефона    | Текстовый        |
|                      | Факс              | Числовой         |
|                      | Электронная почта | Гиперссылка      |

Таблица Заказы

| Работа с таблицами   |                      |                  |
|----------------------|----------------------|------------------|
| Главная              | Создание             | Внешние данные   |
| Режим                | Ключевое поле        | Построитель      |
| Режимы               | Сервис               | Проверка условий |
| Вставить строки      |                      | Удалить строки   |
| Страница свойств     |                      | Индексы          |
| Показать или скрыть  |                      |                  |
| Все таблицы          | заказы               |                  |
| сотрудники           | Имя поля             |                  |
| сотрудники : таблица | Тип данных           |                  |
| клиенты              |                      |                  |
| клиенты : таблица    |                      |                  |
| заказы               |                      |                  |
| заказы : таблица     |                      |                  |
|                      | Код заказа           | Счетчик          |
|                      | Код клиента          | Числовой         |
|                      | Код сотрудника       | Числовой         |
|                      | Дата размещения      | Дата/время       |
|                      | Дата исполнения      | Дата/время       |
|                      | Сумма                | Денежный         |
|                      | Отметка о выполнении | Логический       |

#### 4. Установите ключевые поля.

Отдельные таблицы, содержащие информацию по определенной теме, необходимо связать в единую структуру базы данных. Для связывания таблиц следует задать **ключевые поля**.

**Ключ** состоит из одного или нескольких полей, значения которых однозначно определяют каждую запись в таблице. Наиболее подходящим в качестве ключевого поля является *Счетчик*, так как значения в данном поле являются уникальными (т. е. исключают повторы).

При создании таблиц в режиме конструктора ключевое поле устанавливается автоматически. Откройте созданные Вами таблицы в режиме Конструктор и проверьте установленные ключевые поля:

1) в таблице *Сотрудники* ключевое поле *Код сотрудника*

| сотрудники                                                                                       |                  |
|--------------------------------------------------------------------------------------------------|------------------|
| Имя поля                                                                                         | Тип данных       |
|  Код сотрудника | Счетчик          |
| Фамилия                                                                                          | Текстовый        |
| Имя                                                                                              | Текстовый        |
| Отчество                                                                                         | Текстовый        |
| Должность                                                                                        | Текстовый        |
| Телефон                                                                                          | Текстовый        |
| Адрес                                                                                            | Текстовый        |
| Дата рождения                                                                                    | Дата/время       |
| Зарботная плата                                                                                  | Денежный         |
| Фото                                                                                             | Поле объекта OLE |
| Электронная почта                                                                                | Гиперссылка      |

2) в таблице *Клиенты* ключевое поле *Код клиента*

| клиенты                                                                                        |             |
|------------------------------------------------------------------------------------------------|-------------|
| Имя поля                                                                                       | Тип данных  |
|  Код клиента | Счетчик     |
| Название компании                                                                              | Текстовый   |
| Адрес                                                                                          | Текстовый   |
| Номер телефона                                                                                 | Текстовый   |
| Факс                                                                                           | Числовой    |
| Электронная почта                                                                              | Гиперссылка |

3) в таблице *Заказы* ключевое поле *Код заказа*

| заказы                                                                                         |            |
|------------------------------------------------------------------------------------------------|------------|
| Имя поля                                                                                       | Тип данных |
|  Код заказа | Счетчик    |
| Код клиента                                                                                    | Числовой   |
| Код сотрудника                                                                                 | Числовой   |
| Дата размещения                                                                                | Дата/время |
| Дата исполнения                                                                                | Дата/время |
| Сумма                                                                                          | Денежный   |
| Отметка о выполнении                                                                           | Логический |

Если значение *Ключевых полей* не задалось автоматически, то задайте их вручную. Для этого откройте таблицу **Сотрудники** в режиме **Конструктора**. Нажмите правой кнопкой мыши на поле **Код сотрудника** и в появившемся контекстном меню выберите команду **Ключевое поле**. Если в таблице необходимо установить несколько ключевых полей, то выделить их можно, удерживая клавишу **Ctrl**. Для таблицы **Клиенты** установите ключевое поле **Код клиента**, а для таблицы **Заказы** - **Код заказа**.

##### 5. Создайте раскрывающиеся списки с помощью Мастера подстановок.

Таблица **Заказы** содержит поля **Код сотрудника** и **Код клиента**. При их заполнении могут возникнуть некоторые трудности, так как не всегда удастся запомнить все предприятия, с которыми работает фирма, и всех сотрудников с номером кода. Для удобства можно создать раскрывающиеся списки с помощью **Мастера подстановок**.

Откройте таблицу **Заказы** в режиме **Конструктора**. Для поля **Код клиента** выберите тип данных **Мастер подстановок**.

| заказы               |                    |
|----------------------|--------------------|
| Имя поля             | Тип данных         |
| Код заказа           | Счетчик            |
| Код клиента          | Числовой           |
| Код сотрудника       | Текстовый          |
| Дата размещения      | Поле МЕМО          |
| Дата исполнения      | Числовой           |
| Сумма                | Дата/время         |
| Отметка о выполнении | Денежный           |
|                      | Счетчик            |
|                      | Логический         |
|                      | Поле объекта OLE   |
|                      | Гиперссылка        |
|                      | Вложение           |
|                      | Мастер подстановок |

В появившемся окне выберите команду **Объект "столбец подстановки"** будет использовать значения из таблицы или запроса и щелкните на кнопке **Далее**.

Создание подстановки

Мастер создает столбец подстановки, в котором отображается список значений для выбора. Каким способом столбец подстановки будет получать эти значения?

☒ Объект "столбец подстановки" будет использовать значения из таблицы или запроса.

☐ Будет введен фиксированный набор значений.

Отмена < Назад **Далее >** Готово

В списке таблиц выберите **таблицу Клиенты** и щелкните на кнопке **Далее**.

Создание подстановки

Выберите таблицу или запрос со значениями, которые будет содержать столбец подстановки.

Таблица: клиенты

Таблица: сотрудники

Показать ☒ Таблицы ☐ Запросы ☐ Таблицы и запросы

Отмена < Назад **Далее >** Готово

В списке **Доступные поля** выберите поле **Код клиента** и щелкните на кнопке со стрелкой >>, чтобы ввести поле в список **Выбранные поля**. Таким же образом добавьте поле **Название компании** и щелкните на кнопке **Далее**.

**Создание подстановки**

Какие поля содержат значения, которые следует включить в столбец подстановки? Отобранные поля станут столбцами в объекте "столбец подстановки".

Доступные поля: Адрес, Номер телефона, Факс, Электронная почта

Выбранные поля: Код клиента, Название компании

Оформление: [>], [>>], [<], [<<]

Отмена < Назад **Далее >** Готово

Выберите порядок сортировки списка по полю **Название компании** и нажмите кнопку **Далее**.

**Создание подстановки**

Выберите порядок сортировки элементов списка.

Допускается сортировка записей по возрастанию или по убыванию, включающая до 4 полей.

1. [ ] по возрастанию

2. (Отсутствует) по возрастанию

3. Код клиента по возрастанию

4. **Название компании** по возрастанию

Отмена < Назад **Далее >** Готово

В следующем диалоговом окне задайте необходимую ширину столбцов раскрывающегося списка, установите флажок **Скрыть ключевой столбец** и нажмите кнопку **Далее**.

**Создание подстановки**

Задайте ширину столбцов, которые содержит столбец подстановки.

Перетащите правую границу заголовка столбца на нужную ширину или дважды щелкните ее для автоматического подбора ширины.

☒ **Скрыть ключевой столбец (рекомендуется)**

Название компании

Отмена < Назад **Далее >** Готово

На последнем шаге **Мастера подстановок** замените при необходимости надпись для поля подстановки и щелкните на кнопке **Готово**.

Создание подстановки

Задайте подпись, которую содержит столбец подстановки.

Код клиента

Сохранить несколько значений для этой подстановки?

☐ Разрешить несколько значений

Указаны все сведения, необходимые мастеру, чтобы создать столбец подстановки.

Отмена < Назад Далее > **Готово**

Сохраните полученный результат.

заказы

| Имя поля             | Тип данных |
|----------------------|------------|
| Код заказа           | Счетчик    |
| Код клиента          | Числовой   |
| Код сотрудника       | Числовой   |
| Дата размещения      | Дата/время |
| Дата исполнения      | Дата/время |
| Сумма                | Денежный   |
| Отметка о выполнении | Логический |

Создание подстановки

Перед созданием связи необходимо сохранить таблицу. Выполнить это сейчас?

**Да** Нет

6. Аналогичным образом создайте раскрывающийся список для поля **Код сотрудника**.

Теперь в списке таблиц выберите таблицу **Сотрудники**

Создание подстановки

Выберите таблицу или запрос со значениями, которые будет содержать столбец подстановки.

Таблица: клиенты  
**Таблица: сотрудники**

Показать ☒ Таблицы ☐ Запросы ☐ Таблицы и запросы

Отмена < Назад **Далее >** Готово

В списке **Доступные поля** выберите поля **Код сотрудника, Фамилия, Имя**.

Создание подстановки

Какие поля содержат значения, которые следует включить в столбец подстановки? Отобранные поля станут столбцами в объекте "столбец подстановки".

Доступные поля:

- Отчество
- Должность
- Телефон
- Адрес
- Дата рождения
- Зарботная плата
- Электронная почта

Выбранные поля:

- Код сотрудника
- Фамилия
- Имя

Оформление: [Оформление]

Отмена < Назад **Далее >** Готово

Порядок сортировки списка выберите по полю **Фамилия**.

Создание подстановки

Выберите порядок сортировки элементов списка.

Допускается сортировка записей по возрастанию или по убыванию, включающая до 4 полей.

1 [ ] по возрастанию

2 (Отсутствует) по возрастанию

3 Код сотрудника по возрастанию

4 Фамилия по возрастанию

Имя по возрастанию

Отмена < Назад **Далее >** Готово

Все остальные действия проводятся аналогично пункту 6.

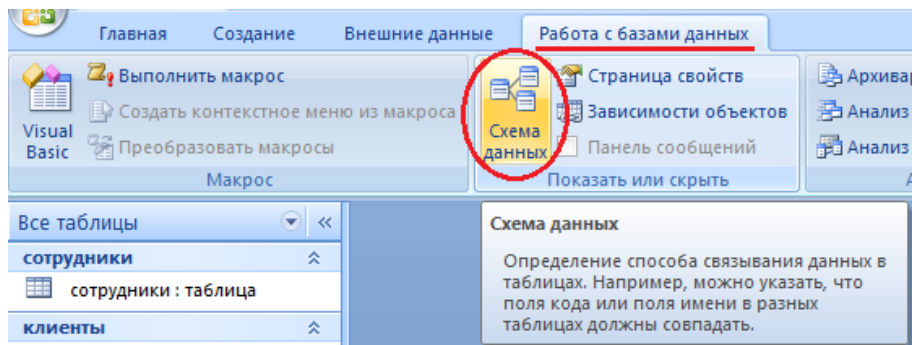
## 7. Создайте связей между таблицами.

Существует несколько типов отношений между таблицами:

- *при отношении «один-к-одному»* каждой записи ключевого поля в первой таблице соответствует только одна запись в связанном поле другой таблицы, и наоборот. Отношения такого типа используются не очень часто. Иногда их можно использовать для разделения таблиц, содержащих много полей, для отделения части таблицы по соображениям безопасности;
- *при отношении «один-к-многим»* каждой записи в первой таблице соответствует несколько записей во второй, но запись во второй таблице не может иметь более одной связанной записи в первой таблице;
- *при отношении «многие-к-многим»* одной записи в первой таблице могут соответствовать несколько записей во второй таблице, а одной записи во второй таблице могут соответствовать несколько записей в первой.

**Закройте все открытые таблицы, так как создавать или изменять связи между открытыми таблицами нельзя.**

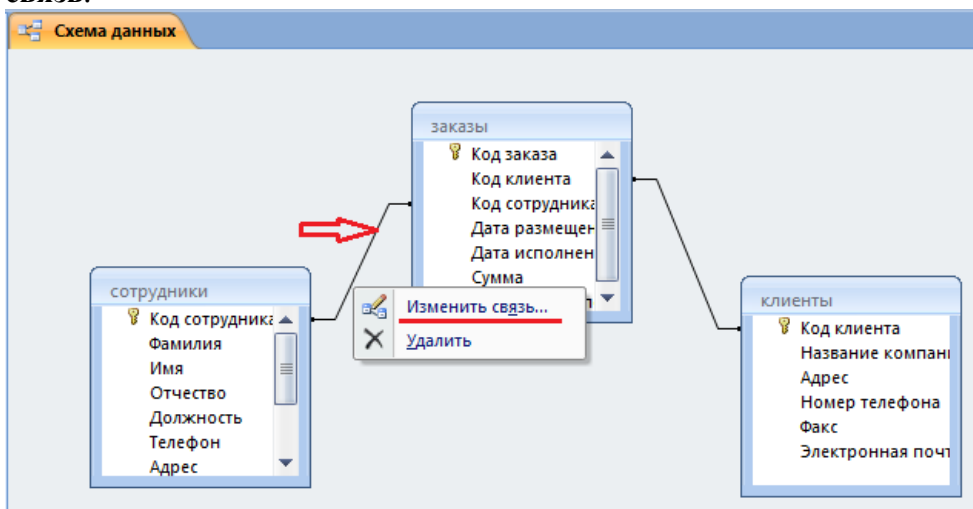
**Выполните команду вкладки Лента Работа с базами данных кнопка Схема данных**



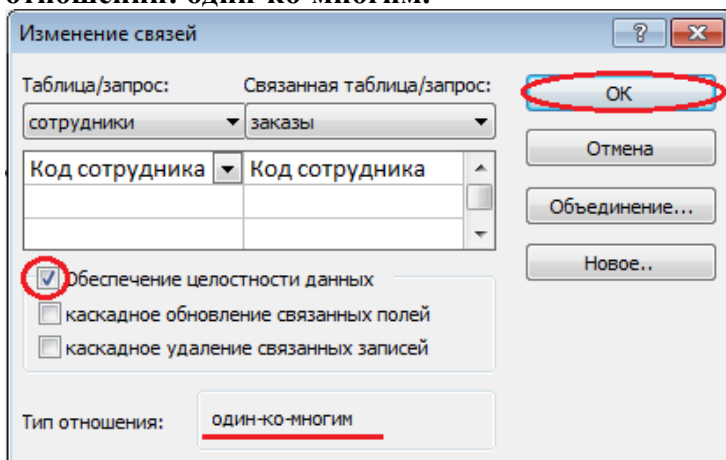
Если ранее никаких связей между таблицами базы не было, то при открытии окна **Схема данных** одновременно открывается окно **Добавление таблицы**, в котором выбираются нужные таблицы. Для добавления в схему данных новой таблицы необходимо щелкнуть правой кнопкой мыши на схеме данных и в контекстном меню выбрать пункт **Добавить таблицу**.

Если связи между таблицами уже были заданы, то откроется окно **Схема данных**, на котором будут отображены таблицы и связи между ними.

Отредактируйте связь между таблицами Сотрудники и Заказы, для этого щелкните правой кнопкой мыши (ПКМ) на линию связи и в открывшемся контекстном меню выберите команду **Изменить связь**.



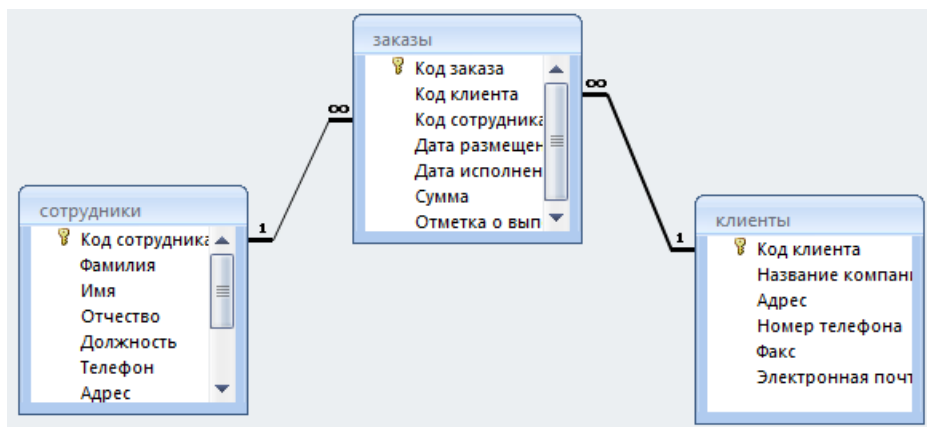
Откроется диалоговое окно **Изменение связей**, в котором включите флажок Обеспечение целостности данных. Это позволит предотвратить случаи удаления записей из одной таблицы, при которых связанные с ними данные других таблиц останутся без связи. Обратите внимание на **тип отношений: один-ко-многим**.



Флажки **Каскадное обновление связанных полей** и **Каскадное удаление связанных записей** обеспечивают одновременное обновление или удаление данных во всех подчиненных таблицах при их изменении в главной таблице. Параметры связи можно изменить, нажав на кнопку **Объединение**. После установления всех необходимых параметров нажмите **кнопку ОК**.

Аналогично измените связь между таблицами Клиенты и Заказы.

В результате должна получиться схема данных, представленная на рисунке.



На схеме данных связи отображаются в виде соединительных линий со специальными значками около таблиц. Связь «один-к-многим» помечается «1» вблизи главной таблицы (имеющей первичный ключ) и «∞» вблизи подчиненной таблицы (имеющей внешний ключ). Связь «один-к-одному» помечается двумя «1» (оба поля таблиц имеют первичные ключи). Неопределенная связь не имеет никаких знаков. Если установлено объединение, то его направление отмечается стрелкой на конце соединительной линии (ни одно из объединенных полей не является ключевым и не имеет уникального индекса).

### 7. В таблицу Сотрудники внесите данные о 7 работниках.

| Код сотр | Фамилия  | Имя     | Отчество       | Должность     | Телефон     | Адрес           | Дата рожде | Заработная  | Фото | Электронная почта |
|----------|----------|---------|----------------|---------------|-------------|-----------------|------------|-------------|------|-------------------|
| 1        | Иванов   | Сергей  | Юрьевич        | Директор      | 89182121567 | ул.Батарейная,8 | 05.07.1987 | 57 000,00 Р |      | ivanov@mail.ru    |
| 2        | Орлова   | Юлия    | Константиновна | Зам директора | 89894938474 | ул.Красная,10   | 07.09.1985 | 50 000,00 Р |      | orlova@mail.ru    |
| 3        | Романов  | Вадим   | Романович      | Менеджер      | 89883467464 | ул.Матвиевко,46 | 13.05.1989 | 50 000,00 Р |      | romanov@mail.ru   |
| 4        | Суворов  | Максим  | Александрович  | Менеджер      | 89184857634 | ул.Минская,11   | 19.01.1983 | 45 000,00 Р |      | syvov@mail.ru     |
| 5        | Марченко | Андрей  | Евгеньевич     | Бухгалтер     | 89887765265 | ул.Мамаева,1    | 22.04.1989 | 42 500,00 Р |      | Marchenko@mail.ru |
| 6        | Вдовенко | Николай | Андреевич      | Упаковщик     | 89894289642 | ул.Рабочая,19   | 27.08.1990 | 40 000,00 Р |      | Vdovenko@mail.ru  |
| 7        | Афонин   | Олег    | Павлович       | Грузчик       | 89641414100 | ул.Свободы,27   | 21.05.1984 | 38 000,00 Р |      | Afonia@mail.ru    |

### 8. В таблицу Клиенты внесите данные о 7 предприятиях, с которыми работает данная фирма.

| Код клиент | Название компании | Адрес              | Номер теле  | Факс      | Электронная почта |
|------------|-------------------|--------------------|-------------|-----------|-------------------|
| 1          | NaVi              | ул. Советов,47     | 89188754416 | 861276543 | navi@mail.ru      |
| 2          | Virtys            | ул. Мира,31        | 89897464633 | 861209864 | virtys@gmail.com  |
| 3          | Godsent           | ул. Рубина,11      | 89886741641 | 861273733 | godsent12@bk.com  |
| 4          | NiP               | ул. Куникова,89    | 89647814711 | 861247474 | n1pi@mail.ru      |
| 5          | Qbfire            | ул. Ленина,13      | 89881614614 | 861274644 | qbfi13@gmail.com  |
| 6          | Gambit            | ул. Планеристов,99 | 89894164141 | 861284174 | gambity12@mail.ru |
| 7          | Vega              | ул. Весенняя,76    | 89181571751 | 861267464 | v3gaa2133@mail.ru |

### 9. В таблице Заказы оформите 5 заявок, поступивших на фирму.

| Код заказа | Код клиент | Код сотрудника | Дата размещения | Дата исполнения | Сумма       | Отметка о выполнении                |
|------------|------------|----------------|-----------------|-----------------|-------------|-------------------------------------|
| 1          | Godsent    | Романов        | 03.03.2018      | 05.03.2018      | 30 000,00 Р | <input checked="" type="checkbox"/> |
| 2          | Gambit     | Суворов        | 04.03.2018      | 09.03.2018      | 20 000,00 Р | <input type="checkbox"/>            |
| 3          | NaVi       | Суворов        | 01.03.2018      | 04.03.2018      | 54 000,00 Р | <input checked="" type="checkbox"/> |
| 4          | Vega       | Вдовенко       | 02.03.2018      | 07.03.2018      | 15 000,00 Р | <input checked="" type="checkbox"/> |
| 5          | NiP        | Вдовенко       | 13.03.2018      |                 | 27 000,00 Р | <input checked="" type="checkbox"/> |

### 10. Покажите работу преподавателю.

### 11. Ответьте на контрольные вопросы.

#### Контрольные вопросы:

- 1 С помощью чего можно создавать таблицы?
- 2 Что такое ключевое поле?
- 3 Как установить несколько ключевых полей?
- 4 Как установить связи между таблицами?
- 5 Какие существуют отношения между таблицами?
- 6 Что означают на схеме данных «1» и «∞»?
- 7 Зачем нужен Мастер подстановок?
- 8 Для чего нужен механизм запросов?

#### Отчет

Сохранить БД в своей папке под названием БД\_Фамилия.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №2

**Тема:** Составление словаря данных (3 часа)

**Цель:**

1. Изучение средства создания словаря данных Access

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование** ПК, Windows, MS Access

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

Практические задания

#### Теоретическая часть

Определив отдельные элементы данных, составляющие таблицы базы данных, и установив отношения между ними, необходимо подготовить описание базы данных, называемое словарем данных. Словари данных имеют огромное значение для баз данных. Недокументированной базой данных тяжело управлять и ее трудно обслуживать. Ошибки и промахи, допущенные на этапе ее разработки, во многих случаях можно обнаружить при подготовке предварительного словаря данных.

Завершив разработку структуры базы данных и убедившись в ее правильности, требуется довести до конца разработку подробного словаря данных. По мере добавления в приложение новых или модификации существующих форм и отчетов, необходимо соответствующим образом обновлять и словарь. Даже при создании базы данных для личного использования упрощенная версия словаря данных позволит получить большую пользу при малых затратах времени.

Стандартный словарь данных

Словарь данных содержит информацию о базе данных в целом, обо всех таблицах, полях, включенных в таблицы, о первичных и внешних ключах, а также включает толкование используемых кодов. В словаре также приводится назначение и описание каждого приложения, использующего базу.

Поскольку словари данных имеют иерархическую структуру, они могут храниться в традиционном формате, поддерживаемом текстовыми редакторами Windows. Ниже приводится структура стандартного словаря данных, в котором заданы традиционные заголовки:

1. **БАЗА ДАННЫХ**—Полное название базы и имя файла.

Описание назначения и общего содержания базы данных, а также лиц, которые могут ею пользоваться. Список приложений, работающих с базой, и информация о других базах данных, использующих данные из данной базы. Если имеются, то сюда же включаются диаграммы базы данных.

**А. ОБЛАСТЬ ДАННЫХ** — Название группы, к которой принадлежат таблицы.

Если таблицы классифицируются по группам, например, финансовая группа, то включается описание каждой группы.

1. ТАБЛИЦА — Таблицы, входящие в область данных.

а) ДОСТУП — Права пользователей на доступ к таблице.

б) ЗАПИСЬ — Общее определение элементов данных.

(1) ПЕРВИЧНЫЙ КЛЮЧ - Поле (поля) первичного ключа.

(а) ИНДЕКС — Описание индекса первичного ключа.

(2) ВНЕШНИЕ КЛЮЧИ - Внешние ключевые поля.

(а) ИНДЕКС - Индексы . внешних ключей.

(3) ПОЛЯ — Неключевые поля.

(а) ПЕРЕЧИСЛИМЫЕ МНОЖЕСТВА — Допустимые коды для полей.

За каждым заголовком идет текст, описывающий назначение элемента, базы данных, к которому относится заголовок. Последующие разделы словаря включают описания объектов, которые используют таблицы базы данных, с подзаголовками для запросов, форм и отчетов. Изображения, снятые с экрана, и копии отчетов также добавляются в словарь данных. Распечатки кода программ обычно приводятся в приложениях словаря. Подробные словари данных необходимы для обслуживания базы данных. Кроме того, описание словаря можно представить в табличной форме.

Интегрированный словарь данных

Надстройка Архивариус (Documentor), которая впервые появилась в Access 2.0., позволяет создать отчет, где подробно описываются объекты и значения их свойств для текущей базы данных.

Средство Access 97 "Публикация в MS Word", запускаемое с помощью кнопки на панели инструментов, позволяет сохранить отчет в формате .RTF. Затем созданный файл можно импортировать в Microsoft Word или любой другой текстовый процессор, который обрабатывает файлы в формате RTF, например, WordPad. Кроме того, можно экспортировать отчет в формате BIFF, нажав на панели инструментов кнопку "Анализ в MS Excel".

Во многих случаях Архивариус (Documentor) сообщит вам больше, чем вы хотите знать о вашей базе данных; например, полный отчет обо всех объектах базы данных Борея (Northwind) составляет около 400 страниц. Однако чаще требуется поместить в словарь данных только информацию о таблицах и, возможно, запросах.

Для создания словаря данных с помощью Архивариуса (Documentor):

1. Откройте требуемую базу данных и выберите команду "Сервис, Анализ, Архивариус" (Tools, Analyze, Documentor).

2. Выберите вкладку с требуемым типом объектов, которые необходимо описать. Если выбрать вкладку "Все типы объектов" (All Object Types) и нажать кнопку "Выбрать все" (Select All), то можно создать описание для всех объектов базы данных.

3. Нажмите кнопку "Параметры" (Options) для вывода диалогового окна "Печать описания таблицы" (Print Table Definition). По умолчанию печатается наиболее подробное описание таблиц и индексов. Если база данных не имеет системы защиты, то можно сбросить флажок "Разрешения для пользователей и групп" (Permissions By User and Group), чтобы не выводить в отчете данные о правах доступа. Нажмите кнопку ОК для возврата в первое диалоговое окно Архивариуса (Documentor).

4. Выберите требуемые таблицы, нажимая кнопку "Выделить" (Select). При этом будет установлен флажок напротив имени указанной таблицы. Кроме того, можно выделить объект, дважды щелкнув по его названию. Если необходимо документировать все таблицы базы данных, нажмите кнопку "Выбрать все" (Select All).

5. Выберите вкладку "Запросы" (Queries) для вывода запросов базы данных. Нажмите кнопку "Параметры" (Options) для вывода диалогового окна "Печать описания запроса" (Print Table Definition). Сбросьте флажок "Разрешения для пользователей и групп" (Permissions By Group and Group), чтобы не выводить в отчете данные о правах доступа, и задайте нужные параметры, чтобы не выводить данные, дублирующие информацию о полях таблиц. Нажмите кнопку "ОК".

6. Выберите запрос "Сведения о заказах", дважды щелкнув по его имени, и нажмите кнопку ОК для создания отчета длиной 13 страниц.

7. Вскоре появляется окно предварительного просмотра "Описание объекта" (Object Definition). Обратите внимание на то, что отчет о трех объектах занимает одну страницу.

8. Нажмите на панели инструментов кнопку "Печать" (Print) для печати отчета либо щелкните по кнопке "Связи с Office" (Office Links) для создания файла в формате .RTF или .XLS

9. Нажмите на панели инструментов кнопку "Закрыть" (Close) либо дважды щелкните по кнопке системного меню для закрытия окна предварительного просмотра.

Документирование других объектов базы данных выполняется аналогичным образом. Можно вывести данные о самой базе данных, выбрав вкладку "Текущая база данных" (Current Database), или распечатать данные из выбранных форм, отчетов, макросов и модулей.

## ОТЧЕТ

Сохраните работу в своей папке БД\_2\_Фамилия.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Тема 1.2. Серверы баз данных

### Лабораторное занятие №3

**Тема:** Разработка технических требований к серверу баз данных (3 часа)

#### Цель:

1. Познакомиться с принципами разработки технических требований к серверу баз данных

**ИНСТРУКЦИОННАЯ КАРТА**

#### Материально - техническое оснащение

**Оборудование:** ПК, Windows, MS Access

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

В поле Имя базы данных введите имя нашей будущей базы данных, например –University. Поле Владелец-задан по умолчанию, в зависимости от настройки сервера. Папка с базой данных будет создана по умолчанию на диске C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS2\MSSQL\DATA \. Прежде чем нажать кнопку Добавить, просмотрите Параметры Файловые группы для создаваемой базы данных. После нажатия на кнопку [OK] программа "SQL Server Management Studio " создаст базу данных, имя которой вы увидите в обозревателе объектов, а также сгенерирует необходимый SQL-код для создания базы данных с теми свойствами, которые указаны в этом диалоговом окне и передаст его серверу СУБД для выполнения. Пример этих операторов приведен на рис. 4. (нажмите на имени базы данных University правой клавишей и из контекстного меню выберите Создать скрипт как.. CREATE). Если параметры введены правильно, база данных будет создана.

Содержащиеся в сценарии операторы отделяются друг от друга символом ";". Сценарий может содержать поясняющие комментарии двух видов: многострочный комментарий (начинается символами "/\*" и заканчивается символами "\*/") и однострочный комментарий, который начинается символами "--" и продолжается до конца строки. При создании базы данных возможны следующие типичные ошибки: 1. На целевом компьютере не запущен или не установлен сервер СУБД – т.е. выполнять команду создания базы данных просто некому. 2. На целевом компьютере нет каталога, в котором предполагается создать базу данных. 3. Файл, в котором должна будет находиться база

данных на сервере, уже существует. После создания базы данных вся введенная о базе данных информация запоминается программой SQL Server Management Studio и в окно редактора в дерево на вкладке "Проводник" добавляется узел с зарегистрированной базой данных (рис. 5).

Второй способ создания БД. Выполнить в программе SQL Server Management Studio команду "Создать запрос" на панели инструментов, затем ввести команду, создающую базу данных в окне "Script Execute" (рис. 3) и нажать кнопку . Команда CREATE DATABASE -Создание базы данных MS SQL Server Базы данных создаются командой CREATE DATABASE. Создание баз данных разрешается любому пользователю с ролью системного администратора или всем, кому системный администратор предоставил такое право. Команда CREATE DATABASE имеет следующий синтаксис: Если при создании базы не указан первичный файл данных и/или файл журнала, то отсутствующий файл (или файлы) создается с именем по умолчанию. Физические файлы будут находиться в стандартном каталоге. Первичному файлу присваивается имя имя\_базы.mdf, а файлу журнала — имя\_базы\_log.ldf. Если размер файлов не задан, то при создании размер первичного файла совпадает с размером первичного устройства базы model, а размер файла журнала и вторичных файлов данных равен 1 Мбайт. Он может быть и больше, если размер первичного файла базы данных model превышает 1 Мбайт. Хотя имена и размеры файлов указывать не обязательно, на практике это всегда следует делать. SQL Server создает базу данных за два этапа. На первом этапе база model копируется в новую базу данных, а на втором этапе инициализируется все неиспользуемое пространство. Команда CREATE DATABASE имеет следующие параметры: • PRIMARY—файл определяется как первичное устройство.

NAME —логическое имя; по умолчанию совпадает с именем файла. • FILENAME—полное имя файла на диске. • SIZE—исходный размер файла. Минимальный размер файла журнала равен 512 Кбайт. • MAXSIZE—максимальный размер файла. • UNLIMITED—размер файла не ограничивается. • FILEGROWTH—приращение размера в мегабайтах (MB), килобайтах (KB) или процентах (%). По умолчанию приращение равно 10%. • FOR LOAD—обеспечивает обратную совместимость со сценариями SQL, написанными для предыдущих версий SQL Server. • FOR ATTACH—указывает, что файлы базы данных уже существуют. Пользователь, создавший базу данных, является ее владельцем. Все параметры конфигурации базы копируются из базы model, если только при создании базы не был указан параметр FOR ATTACH. В этом случае параметры конфигурации читаются из существующей базы данных. Рассмотрим некоторые примеры команды CREATE DATABASE: /\* База данных со стандартным размером и именами файлов \*/ Задача 1. Создайте sql-скрипт создания новой базы данных под именем Educator на "D:\Базы данных\Группа\ФИО\_студента\Название\_БД.mdf, с первичным устройством, с исходным размером файла в 10 Мбайт и запустите на выполнение скрипт (кнопка на панели инструментов). Выполните в окне обозревателя объектов Обновление. Сохраните созданный скрипт в текущую папку под именем 1.sql. После успешного выполнения и обновления проводника у вас должна появиться новая база данных. Рис. 6. Окно проводника после выполнения сценария создания базы данных

#### Отчет

Сохранить работу

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №4

Тема: Разработка требований к корпоративной сети (3 часа)

Цель:

1. Составить и проанализировать требования к корпоративной сети

## ИНСТРУКЦИОННАЯ КАРТА

### Материально - техническое оснащение

Оборудование: компьютер, Windows, MS Access

Характер выполнения работы: обучающиеся выполняют работу индивидуально.

### Практические задания

#### Разработка требований

Разработка требований — это процесс, включающий мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Различают четыре основных этапа процесса разработки требований:

1. анализ технической осуществимости создания системы,
  2. формирование и анализ требований,
  3. специфицирование требований и создание соответствующей документации, 4. аттестация этих требований.
- На рис. 2 показаны взаимосвязи между этими этапами и результаты, сопровождающие каждый этап процесса разработки системных требований.

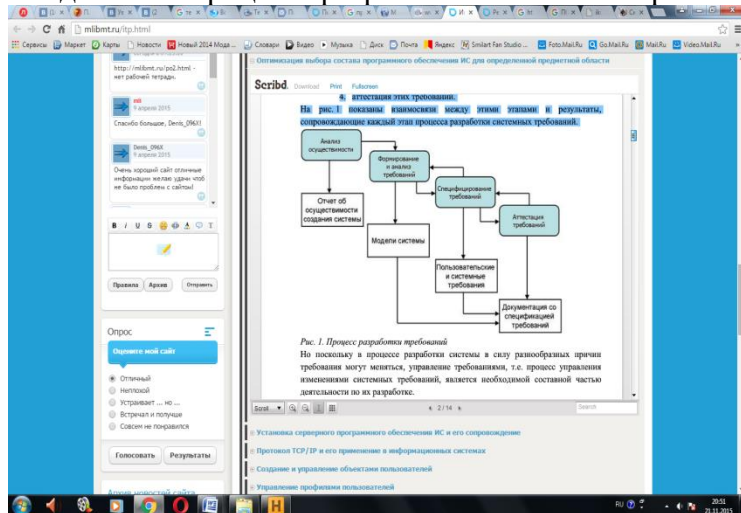
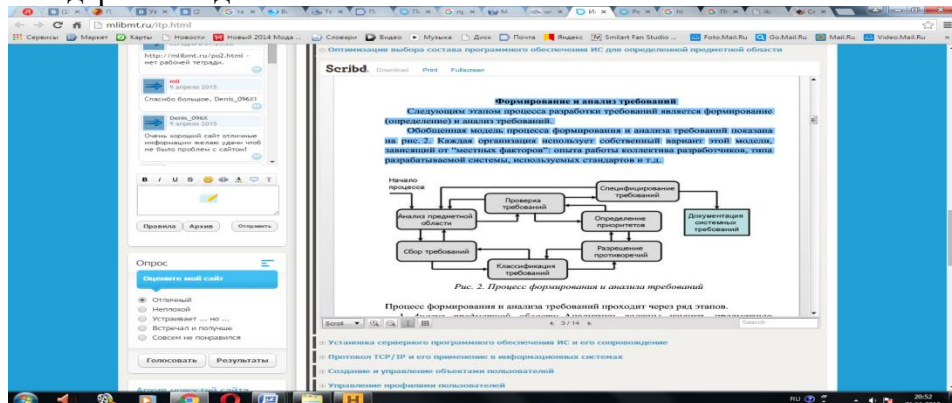


Рис. 2. Процесс разработки требований

Но поскольку в процессе разработки системы в силу разнообразных причин требования могут меняться, управление требованиями, т.е. процесс управления изменениями системных требований, является необходимой составной частью деятельности по их разработке.

#### Формирование и анализ требований

Следующим этапом процесса разработки требований является формирование (определение) и анализ требований. Обобщенная модель процесса формирования и анализа требований показана на рис. 3. Каждая организация использует собственный вариант этой модели, зависящий от —местных факторов!: опыта работы коллектива разработчиков, типа разрабатываемой системы, используемых стандартов и т.д.



### Рис. 3. Процесс формирования и анализа требований

Процесс формирования и анализа требований проходит через ряд этапов.

1. Анализ предметной области. Аналитики должны изучить предметную область, где будет эксплуатироваться система.
2. Сбор требований. Это процесс взаимодействия с лицами, формирующими требования. Во время этого процесса продолжается анализ предметной области.
3. Классификация требований. На этом этапе бесформенный набор требований преобразуется в логически связанные группы требований.
4. Разрешение противоречий. Без сомнения, требования многочисленных лиц, занятых в процессе формирования требований, будут противоречивыми. На этом этапе определяются и разрешаются противоречия различного рода.
5. Назначение приоритетов. В любом наборе требований одни из них будут более важны, чем другие. На этом этапе совместно с лицами, формирующими требования, определяются наиболее важные требования.
6. Проверка требований. На этом этапе определяется их полнота, последовательность и непротиворечивость.

Процесс формирования и анализа требований циклический, с обратной связью от одного этапа к другому. Цикл начинается с анализа предметной области и заканчивается проверкой требований. Понимание требований предметной области увеличивается в каждом цикле процесса формирования требований.

Рассмотрим три основных подхода к формированию требований: метод, основанный на множестве опорных точек зрения, сценарии и этнографический метод.

#### Опорные точки зрения

Подход с использованием различных опорных точек зрения к разработке требований признает различные (опорные) точки зрения на проблему и использует их в качестве основы построения и организации как процесса формирования требований, так и непосредственно самих требований.

Различные методы предлагают разные трактовки выражения "точка зрения". Точки зрения можно трактовать следующим образом.

1. Как источник информации о системных данных. В этом случае на основе опорных точек зрения строится модель создания и использования данных в системе. В процессе формирования требований отбираются все такие точки зрения (и на их основе определяются данные), которые будут созданы или использованы при работе системы, а также способы обработки этих данных.
2. Как структура представлений. В этом случае точки зрения рассматриваются как особая часть модели системы. Например, на основе различных точек зрения могут разрабатываться модели "сущность-связь", модели конечного автомата и т.д.
3. Как получатели системных сервисов. В этом случае точки зрения являются внешними (относительно системы) получателями системных сервисов. Точки зрения помогают определить данные, необходимые для выполнения системных сервисов или их управления.

Наиболее эффективным подходом к анализу таких систем является использование внешних опорных точек зрения. На основе этого подхода разработан метод VORD (Viewpoint-Oriented Requirements Definition — определение требований на основе точек зрения) для формирования и анализа требований.

Основные этапы метода VORD показаны на рис. 4.:

1. Идентификация точек зрения, получающих системные сервисы, и идентификация сервисов, соответствующих каждой точке зрения.
2. Структурирование точек зрения — создание иерархии сгруппированных точек зрения. Общесистемные сервисы предоставляются более высоким уровням иерархии и наследуются точками зрения низшего уровня.
3. Документирование опорных точек зрения, которое заключается в точном описании идентифицированных точек зрения и сервисов.
4. Отображение системы точек зрения, которая показывает системные объекты, определенные на основе информации, заключенной в опорных точках зрения.



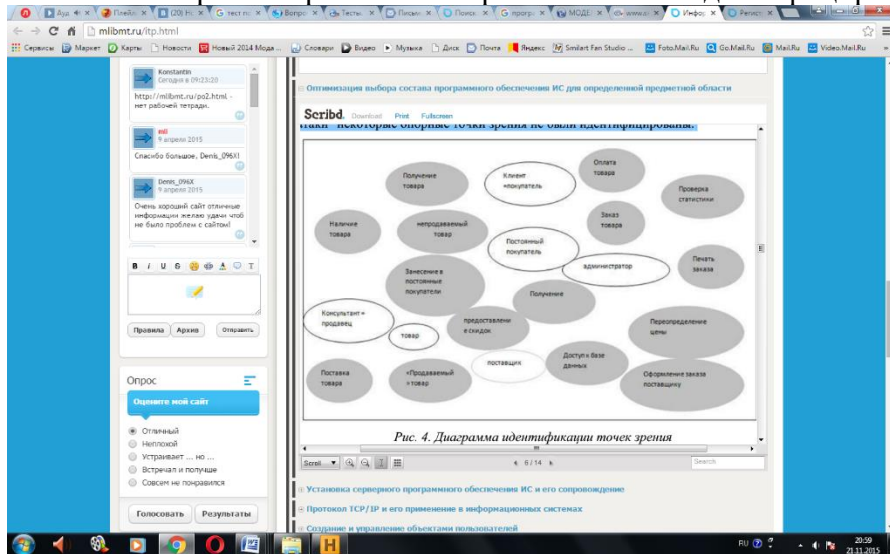
**Рис. 4. Метод VORD**

Пример. Рассмотрим использование метода VORD на первых трех шагах анализа требований для системы поддержки заказа и учета товаров в бакалейной лавке. В бакалейной лавке для каждого товара фиксируется место хранения (определенная полка), количество товара и его поставщик. Система поддержки заказа и учета товаров должна обеспечивать добавление информации о новом товаре, изменение или удаление информации об имеющемся товаре, хранение (добавление, изменение и удаление) информации о поставщиках, включающей в себя название фирмы, ее адрес и телефон. При помощи системы составляются заказы поставщикам. Каждый заказ может содержать несколько позиций, в каждой позиции указываются наименование товара и его количество в заказе. Система по требованию пользователя формирует и выдает на печать следующую справочную информацию:

- список всех товаров;
- список товаров, имеющих в наличии;
- список товаров, количество которых необходимо пополнить;
- список товаров, поставляемых данным поставщиком.

Первым шагом в формировании требований является идентификация опорных точек зрения. Во всех методах формирования требований, основанных на использовании точек зрения, начальная идентификация является наиболее трудной задачей. Один из подходов к идентификации точек зрения — метод "мозговой атаки", когда определяются потенциальные системные сервисы и организации, взаимодействующие с системой. Организуется встреча лиц, участвующих в формировании требований, которые предлагают свои точки зрения. Эти точки зрения представляются в виде диаграммы, состоящей из ряда круговых областей, отображающих возможные точки зрения (рис. 5). Во время "мозговой атаки" необходимо идентифицировать потенциальные опорные точки зрения, системные сервисы, входные данные, нефункциональные требования, управляющие события и исключительные ситуации.

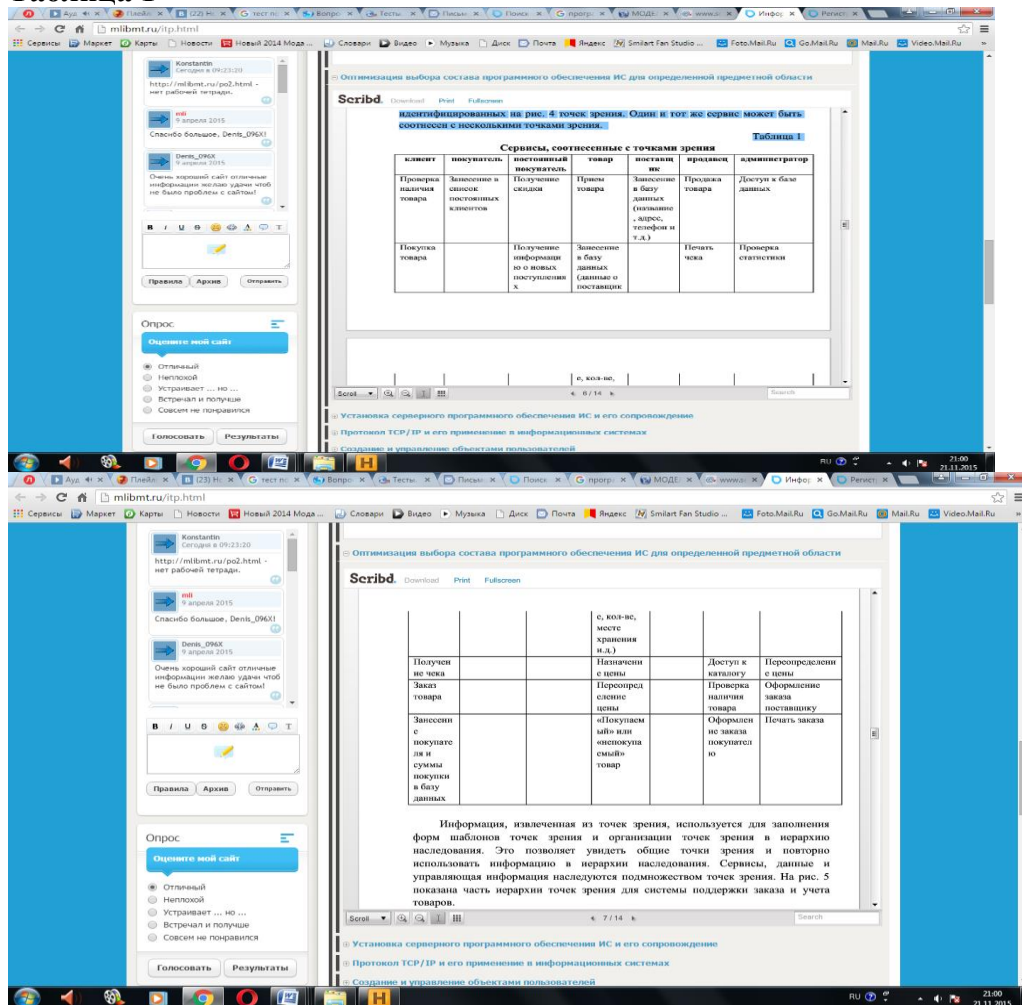
Следующей стадией процесса формирования требований будет идентификация опорных точек зрения (на рис. 5 показаны в виде темных круговых областей) и сервисов (показаны в виде затененных областей). Сервисы должны соответствовать опорным точкам зрения. Но могут быть сервисы, которые не поставлены им в соответствие. Это означает, что на начальном этапе "мозговой атаки" некоторые опорные точки зрения не были идентифицированы.



**Рис. 5. Диаграмма идентификации точек зрения**

В таблице 1 показано распределение сервисов для некоторых идентифицированных на рис. 5 точек зрения. Один и тот же сервис может быть соотнесен с несколькими точками зрения.

Таблица 1



Информация, извлеченная из точек зрения, используется для заполнения форм шаблонов точек зрения и организации точек зрения в иерархию наследования. Это позволяет увидеть общие точки зрения и повторно использовать информацию в иерархии наследования. Сервисы, данные и управляющая информация наследуются подмножеством точек зрения. На рис. 6 показана часть иерархии точек зрения для системы поддержки заказа и учета товаров.

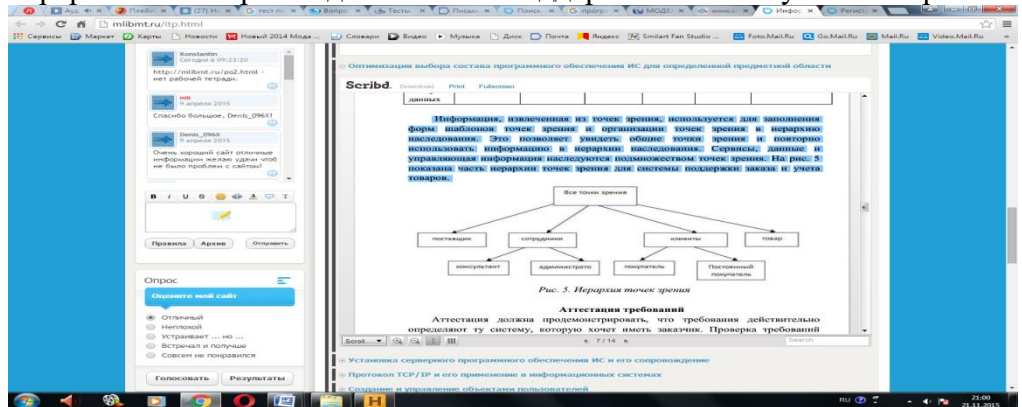


Рис. 6. Иерархия точек зрения  
Аттестация требований

Аттестация должна продемонстрировать, что требования действительно определяют ту систему, которую хочет иметь заказчик. Проверка требований важна, так как ошибки в спецификации требований могут привести к переделке системы и большим затратам, если будут обнаружены во время процесса разработки системы или после введения ее в эксплуатацию. Стоимость внесения в систему изменений, необходимых для устранения ошибок в требованиях, намного выше, чем исправление ошибок проектирования или кодирования. Причина в том, что изменение требований обычно влечет за собой значительные изменения в системе, после внесения которых она должна

пройти повторное тестирование. Во время процесса аттестации должны быть выполнены различные типы проверок требований.

1. Проверка правильности требований. Пользователь может считать, что система необходима для выполнения некоторых определенных функций. Однако дальнейшие размышления и анализ могут привести к необходимости введения дополнительных или новых функций. Системы предназначены для разных пользователей с различными потребностями, и поэтому набор требований будет представлять собой некоторый компромисс между требованиями пользователей системы.

2. Проверка на непротиворечивость. Спецификация требований не должна содержать противоречий. Это означает, что в требованиях не должно быть противоречащих друг другу ограничений или различных описаний одной и той же системной функции.

3. Проверка на полноту. Спецификация требований должна содержать требования, которые определяют все системные функции и ограничения, налагаемые на систему.

4. Проверка на выполнимость. На основе знания существующих технологий требования должны быть проверены на возможность их реального выполнения. Здесь также проверяются возможности финансирования и график разработки системы.

Существует ряд методов аттестации требований, которые можно использовать совместно или каждый в отдельности.

1. Обзор требований. Требования системно анализируются рецензентами.

2. Прототипирование. На этом этапе прототип системы демонстрируется конечным пользователям и заказчику. Они могут экспериментировать с этим прототипом, чтобы убедиться, что он отвечает их потребностям.

3. Генерация тестовых сценариев. В идеале требования должны быть такими, чтобы их реализацию можно было протестировать. Если тесты для требований разрабатываются как часть процесса аттестации, то часто это позволяет обнаружить проблемы в спецификации. Если такие тесты сложно или невозможно разработать, то обычно это означает, что требования трудно выполнить и поэтому необходимо их пересмотреть.

4. Автоматизированный анализ непротиворечивости. Если требования представлены в виде структурных или формальных системных моделей, можно использовать инструментальные CASE-средства для проверки непротиворечивости моделей. Для автоматизированной проверки непротиворечивости необходимо построить базу данных требований и затем проверить все требования в этой базе данных. Анализатор требований готовит отчет обо всех обнаруженных противоречиях.

### **Пользовательские и системные требования**

На основании полученных моделей строятся пользовательские требования, т.е. как было сказано в начале описание на естественном языке функции, выполняемых системой, и ограничений, накладываемых на неё. Пользовательские требования должны описывать внешнее поведение системы, основные функции и сервисы предоставляемые системой, её нефункциональные свойства. Необходимо выделить опорные точки зрения и сгруппировать требования в соответствии с ними. Пользовательские требования можно оформить как простым перечислением, так и используя нотацию вариантов использования.

Далее составляются системные требования. Они включают в себя:

1. Требования к архитектуре системы. Например, число и размещение хранилищ и серверов приложений.

2. Требования к параметрам оборудования. Например, частота процессоров серверов и клиентов, объём хранилищ, размер оперативной и видео памяти, пропускная способность канала и т.д.

3. Требования к параметрам системы. Например, время отклика на действие пользователя, максимальный размер передаваемого файла, максимальная скорость передачи данных, максимальное число одновременно работающих пользователей и т.д.

4. Требования к программному интерфейсу.

5. Требования к структуре системы. Например, масштабируемость, распределённость, модульность, открытость.

- масштабируемость – возможность распространения системы на большое количество машин, не приводящая к потере работоспособности и эффективности, при этом способность системы

наращивать свою мощность должна определяться только мощностью соответствующего аппаратного обеспечения.

- распределенность - система должна поддерживать распределённое хранение данных.
- модульность - система должна состоять из отдельных модулей, интегрированных между собой.
- открытость - наличие открытых интерфейсов для возможной доработки и интеграции с другими системами.

6. Требования по взаимодействию и интеграции с другими системами.

Например, использование общей базы данных, возможность получения данных из баз данных определённых систем и т.д.

## **ПРАКТИЧЕСКАЯ ЧАСТЬ**

### **Порядок выполнения работы**

1. Изучить предлагаемый теоретический материал.
2. Построить опорные точки зрения на основании метода VORD для формирования и анализа требований. Результатом должны явиться две диаграммы: диаграмма идентификации точек зрения и диаграмма иерархии точек зрения.
3. Составить информационную модель будущей системы, включающую в себя описание основных объектов системы и взаимодействия между ними. На основании полученной информационной модели и диаграмм идентификации точек зрения, диаграмма иерархии точек зрения сформировать требования пользователя и системные требования.
4. Провести аттестацию требований, указать какие типы проверок выбрали.
5. На основании описания системы (практическая работа №1), информационной модели, пользовательских и системных требований составить техническое задание на создание программного обеспечения (пример см. Приложение А). ТЗ должно содержать основные разделы, описанные в ГОСТ 34.602-89.
6. Построить отчёт, включающий все полученные уровни модели, описание функциональных блоков, потоков данных, хранилищ и внешних объектов.

### **Содержание отчета**

В отчете следует указать:

1. Цель работы
2. Введение
3. Программно-аппаратные средства, используемые при выполнении работы.
4. Основная часть (описание самой работы), выполненная согласно требованиям к результатам выполнения лабораторного практикума (п.2).
5. Заключение (выводы)

### **Отчет**

Сохранить работу

### **Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №5

Тема Конфигурирование сети (3 часа)

Цель:

1. Научиться настраивать конфигурацию ЛВС в Windows.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows , MS Access

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

##### Настраиваем виртуальный ПК для работы в сети

Запускаем обе, ранее созданные нами виртуальные машины командой **ВМ -Питание Power On**. Для работы в сети настроим сначала первую машину. Для этого в **Панели управления** найдем **Сетевые подключения-Подключение по локальной сети-Свойства**, затем находим свойства **Протокола Интернет (TCP/IP)** и пишем **IP-адрес** и **Маску подсети** как на рис. 1.

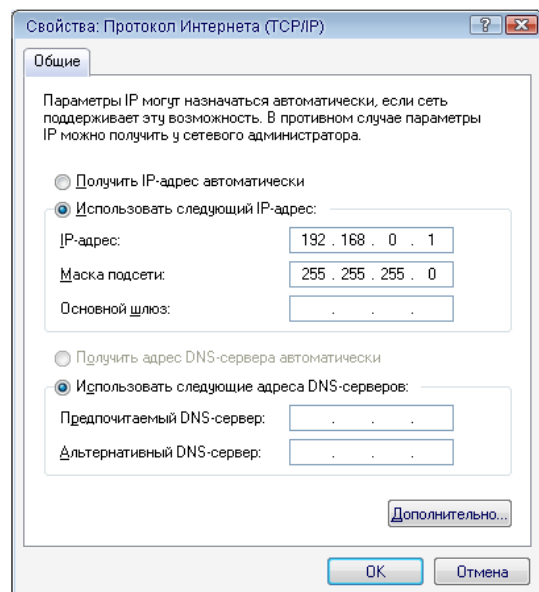
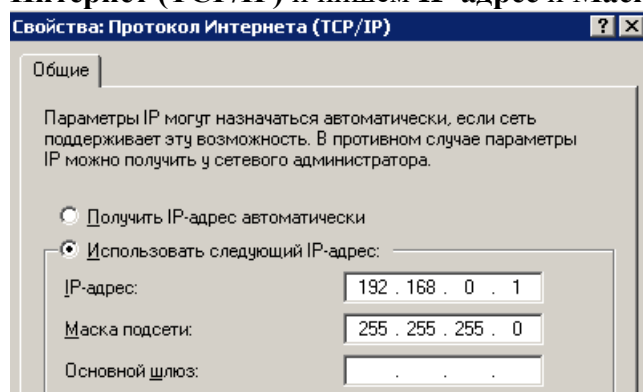


Рис.1. Настраиваем ВМ-1

#### Совет

Вам придется периодически переходить от окна физического ПК к окну виртуального ПК. Для этого нажимайте на сочетания клавиш **Ctrl+Alt**

Аналогично включим и настроим вторую машину ( рис.2).

☐ Получить IP-адрес автоматически  
☒ Использовать следующий IP-адрес:

IP-адрес:   
 Маска подсети:   
 Основной шлюз:

**Рис. 2. Настраиваем VM-2**

### Настраиваем виртуальную сеть

Для настройки сети выполним команду **Сетевое окружение-Установить домашнюю или малую сеть** (рис. 3 и 4).

**Мастер настройки сети**

**Выберите метод подключения.**

Выберите утверждение, наиболее точно описывающее этот компьютер.

☐ Этот компьютер имеет прямое подключение к Интернету. Другие компьютеры в сети подключаются к Интернету через этот компьютер.  
[Показать пример.](#)

☐ Этот компьютер подключен к Интернету через шлюз или через другой компьютер в сети.  
[Показать пример.](#)

☒ Другое

Подробнее о [настройке домашней или малой сети.](#)

**Рис. 3. Выбираем переключатель Другое**

**Мастер настройки сети**

**Другие способы подключения к Интернету...**

Выберите утверждение, наиболее точно описывающее этот компьютер.

☐ Этот компьютер подключен к Интернету напрямую или через сетевой концентратор. Другие компьютеры этой сети подключаются к Интернету таким же образом.  
[Показать пример.](#)

☐ Этот компьютер имеет прямое подключение к Интернету, сеть пока отсутствует.  
[Показать пример.](#)

☒ Этот компьютер принадлежит к сети, не имеющей подключения к Интернету.  
[Показать пример.](#)

Подробнее о [настройке домашней или малой сети.](#)

**Рис.4. Установим третий переключатель сверху**

Присвоим машине сетевое имя (рис. 5).

**Мастер настройки сети**

**Задайте имя и описание для этого компьютера.**

Описание: 
  
Примеры: "Компьютер в гостиной" или "Компьютер Игоря".

Имя компьютера: 
  
Примеры: GOSTINAYA или IGOR

Текущее имя компьютера 110-2.

Дополнительные сведения об [именах и описаниях компьютеров](#).

< Назад    Далее >    Отмена

**Рис. 5.** Даем машине имя и описание

На следующем шаге (нажав **Далее**) создадим рабочую группу 110 ( рис. 6).

**Мастер настройки сети**

**Задайте имя для вашей сети.**

Дайте имя вашей сети, указав имя рабочей группы. Все имена должны иметь одно и то же имя рабочей группы. Использование имени рабочей группы настоятельно не рекомендуется.

Рабочая группа: 
  
Примеры: HOME или OFFICE

**Рис. 6.** Задаем имя для локальной сети

Снова **Далее** и включим общий доступ ( рис. 7).

**Мастер настройки сети**

**Общий доступ к файлам и принтерам**

 Включение общего доступа к файлам и принтерам предоставляет доступ к папке "Общие документы" всем пользователям вашей сети. Если существует общий принтер, он также становится доступным для всех.

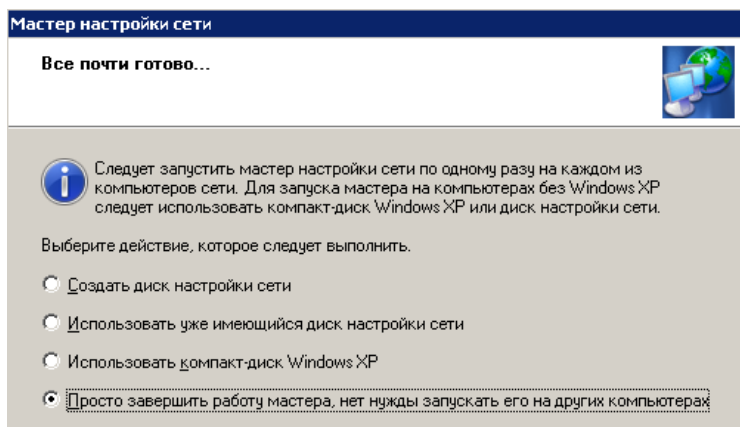
Выберите одну из следующих возможностей.

☒ Включить общий доступ к файлам и принтерам
  
Брандмауэр Windows будет настроен для осуществления общего доступа к файлам и принтерам в вашей сети.

☐ Отключить общий доступ к файлам и принтерам
  
Брандмауэр Windows будет блокировать общий доступ к файлам и принтерам в вашей сети. Если файлы и принтеры с общим доступом уже существуют, то доступ к ним будет прекращен.

**Рис.7.** Установим верхний переключатель

Работу мастера завершаем ( рис.8).

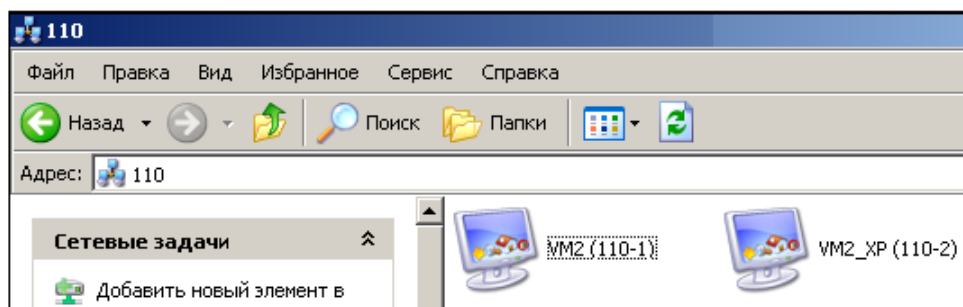


**Рис. 8.** Устанавливаем первый переключатель снизу

Эта машина настроена для работы в сети, перезагружаем ее и аналогично настроим другой виртуальный ПК, также включив его в рабочую группу 110. Перезагружаем. Сетевая настройка обеих машин завершена.

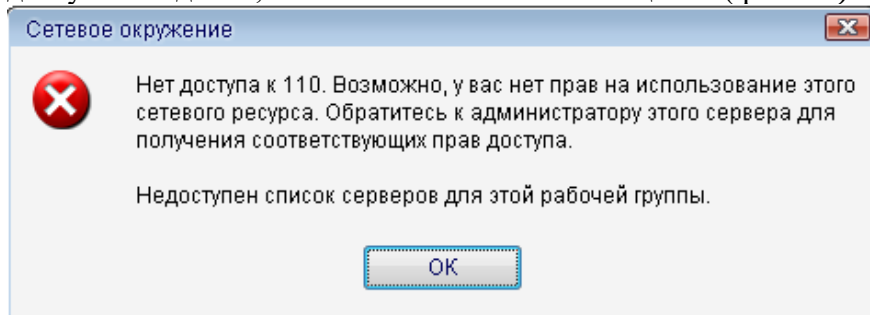
### Проверяем работу виртуальных машин в сети

Попробуем зайти с первой машины на вторую и наоборот. Для этого войдем в **Сетевое окружение** и выполним команду **Отобразить компьютеры рабочей группы**. Если все нормально, то в рабочей группе 110 мы увидим машину 1 и машину 2 (рис. 9).



**Рис. 15.9.** Локальная виртуальная сеть настроена

Далее мы можем работать с такой сетью, как с обычной. Например, создать папки с общим доступом. Однако, может выйти и такое сообщение (рис. 10).



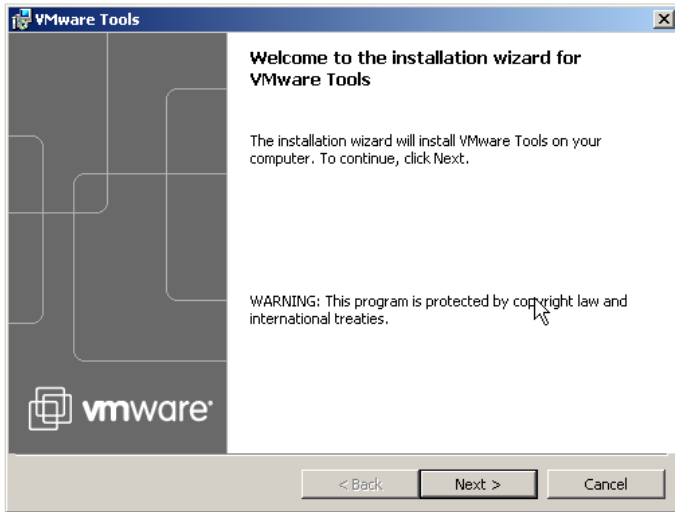
**Рис. 10.** Нет доступа к рабочей группе 110

### Примечание

Одной из причин конфликтов в локальной сети (отсутствие общего доступа между ресурсами) может быть установка разного времени на рабочих станциях, т.е. для обеих машин таймер времени должен быть синхронным. Еще причина – использование нелицензионной операционной системы. Другие причины мы изучим позднее.

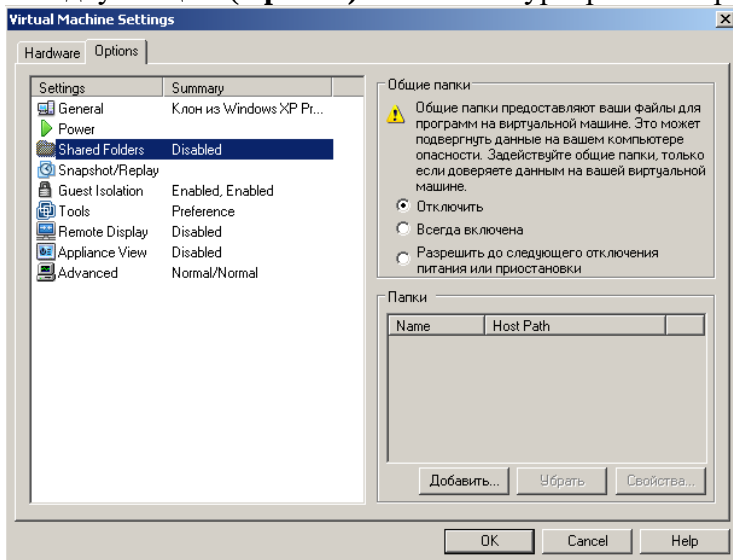
### Установка средств Wmware

Чтобы получить доступ из виртуальной машины к файлам на физическом ПК потребуется команда **ВМ-Установка средств Wmware** (рис. 11).



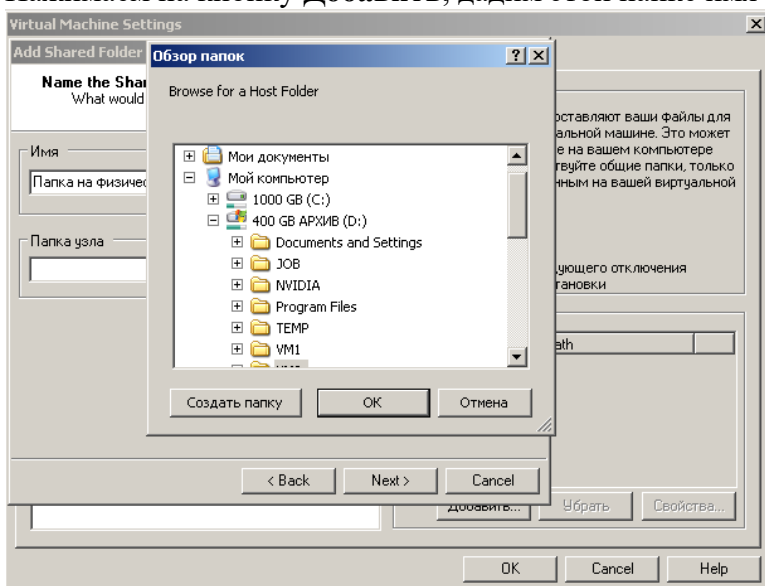
**Рис. 11.** Окно начала установки средств VMware

После инсталляции средств и перезагрузки виртуальной машины на рабочем столе Windows XP появится соответствующий значок . Далее выполним команду **ВМ-Настройки** и откроем вкладку **Опции (Options)** и встанем курсором на строчку папок с общим доступом ( рис. 12).



**Рис. 12.** Папки с общим доступом пока недоступны и пусты

Нажимаем на кнопку **Добавить**, дадим этой папке имя и укажем диск для нее ( рис. 13).



**Рис. 13.** Задаем параметры для папки с общим доступом

Далее активируем атрибуты папки ( рис. 14).

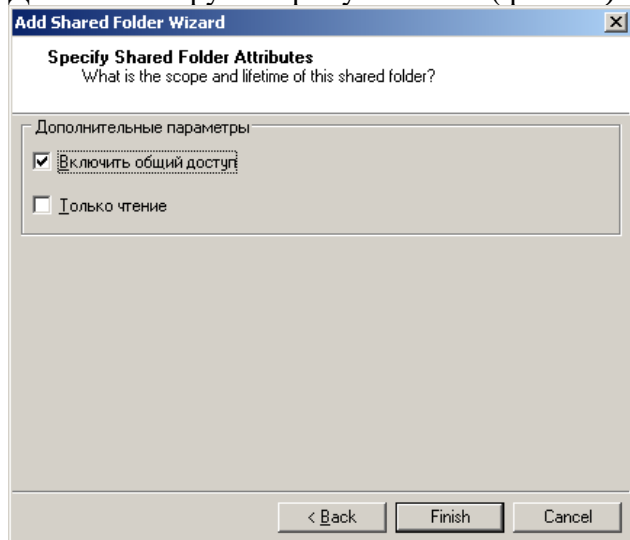


Рис. 14. В этом окне нам нужны оба флажка  
В следующем окне устанавливаем переключатель **Всегда включена** ( рис. 15).

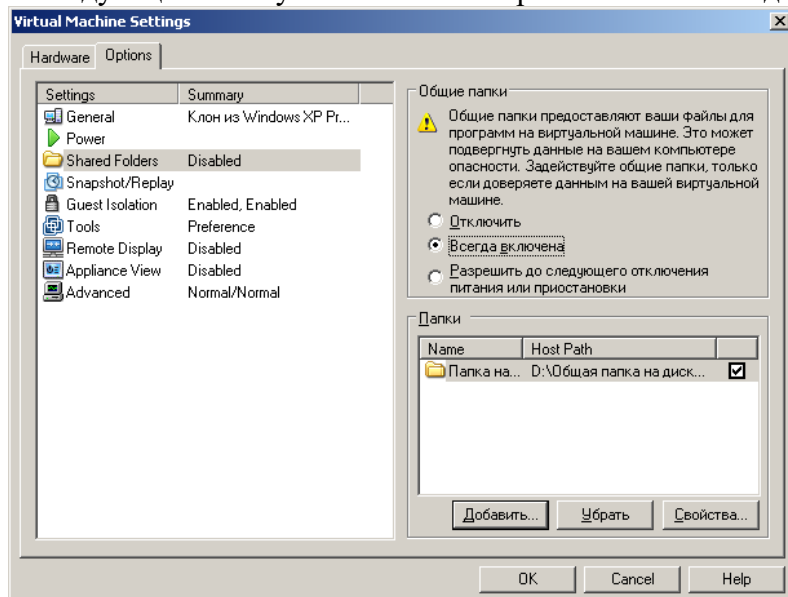
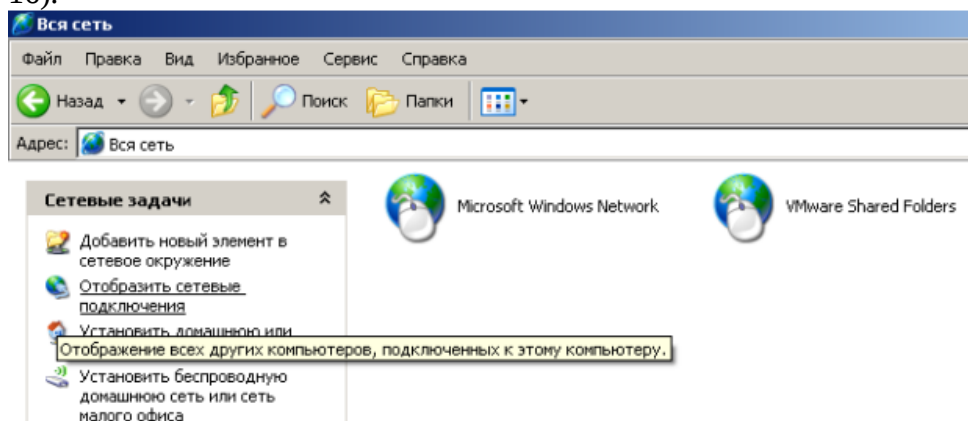
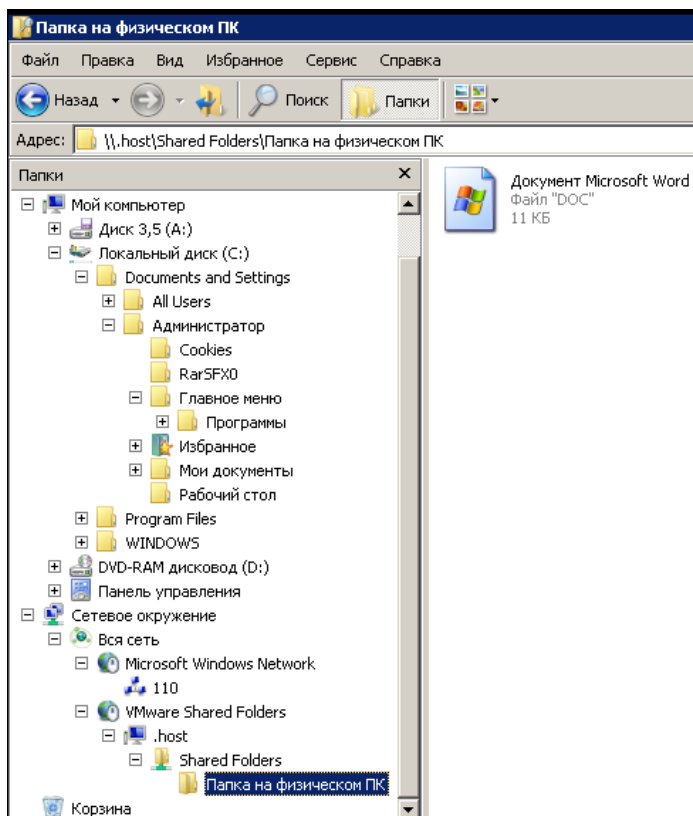


Рис. 15. Активируем этот переключатель

Теперь при просмотре всей сети мы увидим папку на нашем физическом ПК, т.е. через значок **VMware Shared Folders** у нас есть связь физического ПК с виртуальными машинами ( рис. 16).





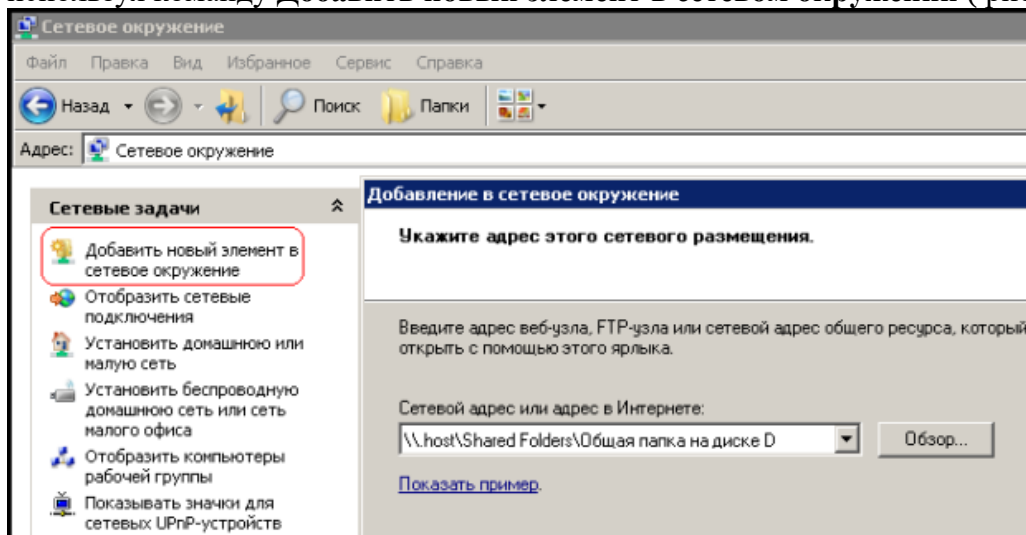
**Рис. 16.** Связь физической машины с виртуальной установлена  
Теперь на виртуальную машину мы можем устанавливать любой soft.

#### Примечание

Обратите внимание на то, что установка средств Wmware решает сразу и другие проблемы, например, настройку драйверов устройств на виртуальном ПК.

#### Совет

Если общая папка на физическом ПК не видна, то в **Сетевом окружении** вы ее можете добавить, используя команду **Добавить новый элемент в сетевом окружении** (рис. 17).



**Рис. 17.** Красным отмечена команда Добавить новый элемент в сетевом окружении

#### Контрольные вопросы:

1. Что такое компьютерные сети, каков их состав и назначение?
2. В чем заключаются преимущества объединения компьютеров в вычислительные сети?
3. Как вы понимаете принцип взаимодействия компьютеров в сети «клиент-сервер»? Каковы отличия компьютеров-серверов и компьютеров-клиентов?
4. Какие вы знаете виды сетей и способы передачи информации в них?
5. Каково назначение различных уровней модели сетевого взаимодействия?

6. Зачем при передаче файлов по сети нужны протоколы?

**Отчет**

Сохранить работу

**Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №6

**Тема:** Сравнение технических характеристик серверов (3 часа)

**Цель занятия:**

1. Научиться создавать DNS сервер, производить его настройку и конфигурирование

## ИНСТРУКЦИОННАЯ КАРТА

**Материально - техническое оснащение**

**Оборудование:** Компьютер, операционная система Windows,

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

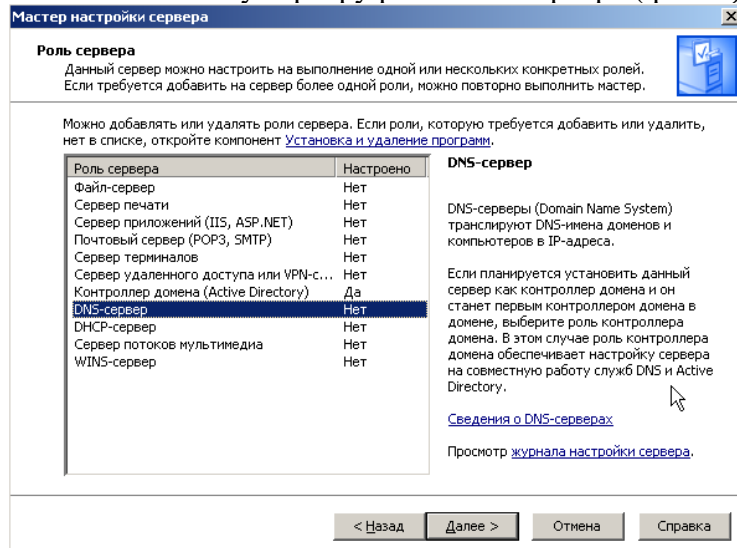
### Практические задания

**Задания для практического занятия:**

**Выбор для сервера роли DNS сервера**

**DNS (Domain Name System** — система доменных имён) — компьютерная система для получения IP-адреса по имени хоста и обратно.

Назначим нашему серверу роль DNS сервера ( рис. 1).



**Рис. 1.** Мастер настройки сервера-DNS сервер

Далее появиться предложение изменить динамический IP адрес на статический ( рис. 2).

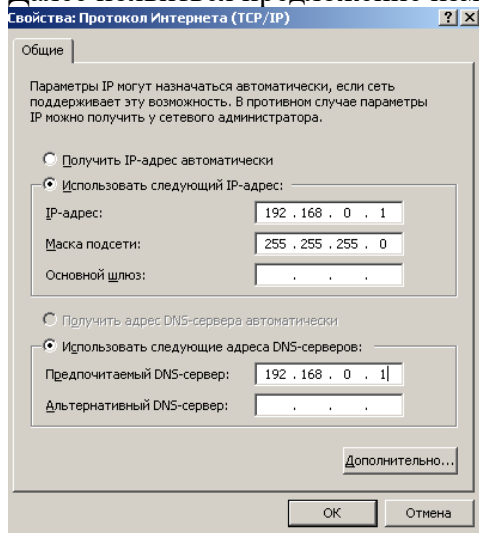


Рис. 2. Задаем серверу статический IP адрес

Далее появится Мастер настройки DNS сервера ( рис. 3).

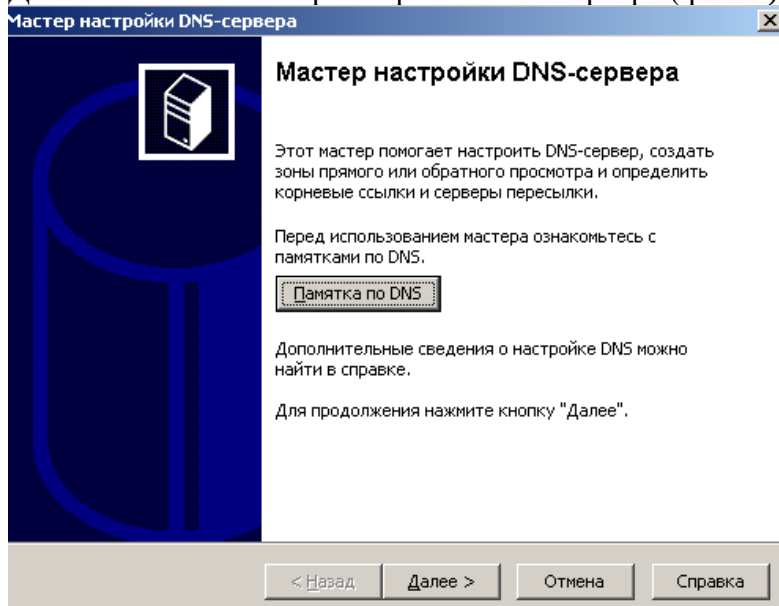


Рис.3. Окно Мастер настройки DNS сервера

Поскольку сеть у нас небольшая, то установим верхний переключатель ( рис. 4).

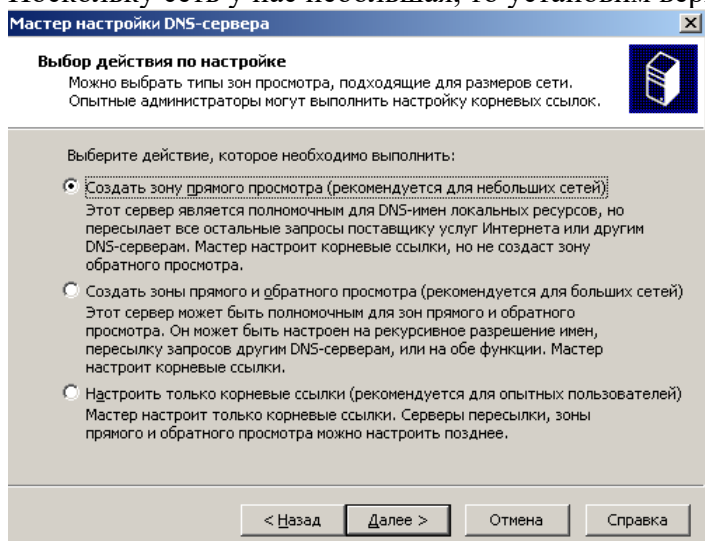
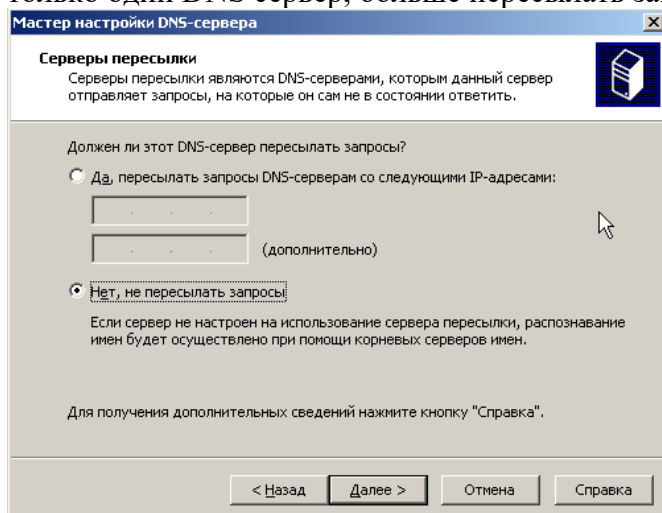
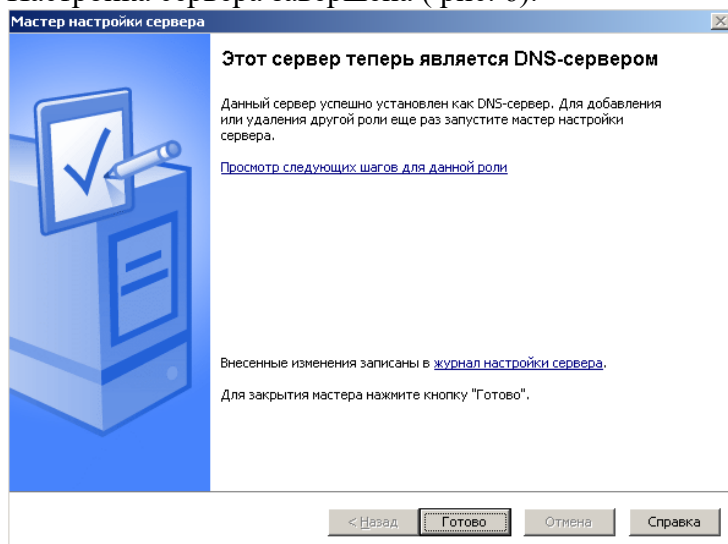


Рис. 4. Создание зоны прямого просмотра

Зону прямого просмотра назовем так же, как и домен – domain.110. Далее исходим из того, что у нас только один DNS сервер, больше пересылать запросы некому ( рис. 5).

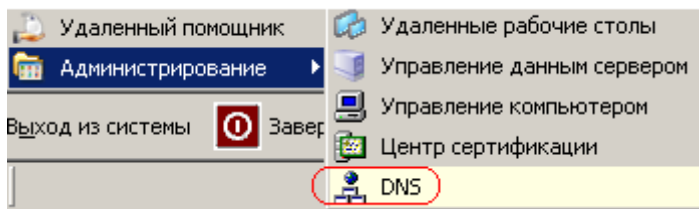


**Рис. 5.** Активируем нижний переключатель  
Настройка сервера завершена ( рис. 6).



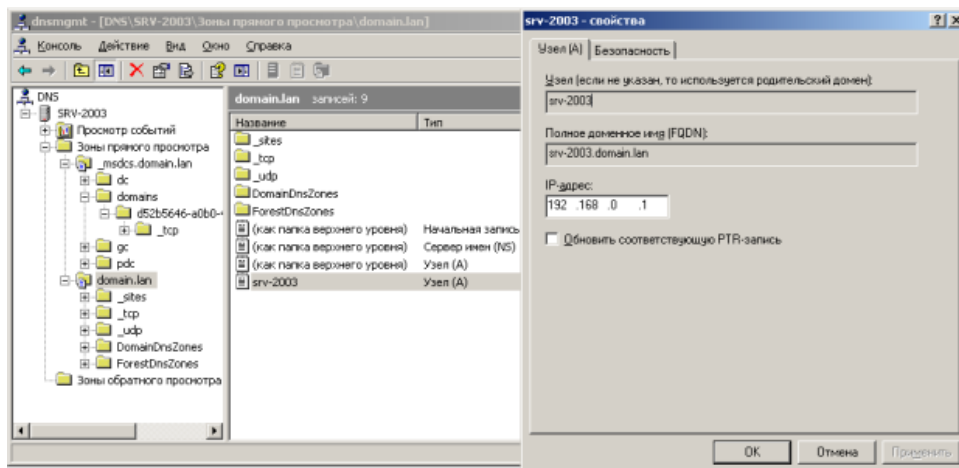
**Рис. 6.** Сервер получил роль DNS сервера

Выполнив команду, **Пуск-Все программы-Администрирование** мы увидим, что появилась новая оснастка ( рис. 7).



**Рис. 7.** На рисунке новая оснастка отмечена красным

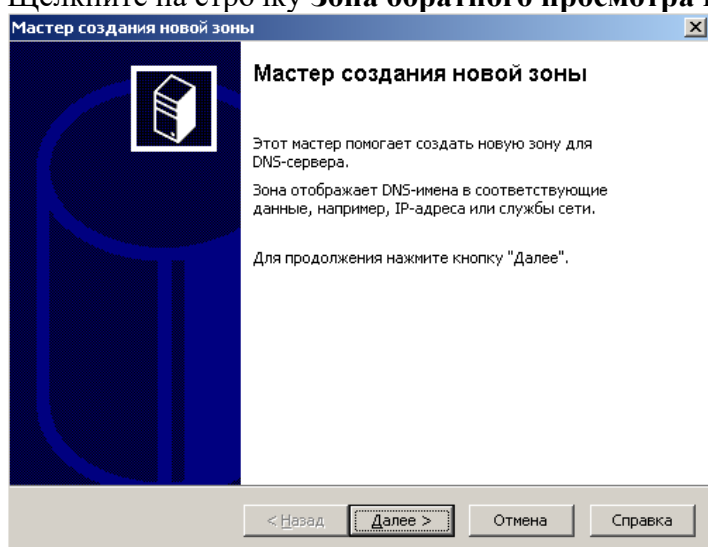
Откроем ее ( рис. 8). Здесь в зоне прямого просмотра вы можете увидеть соответствие имени сервера srv-2003 его IP адресу 192.168.0.1.



**Рис. 8.** На рисунке открыта зона прямого просмотра

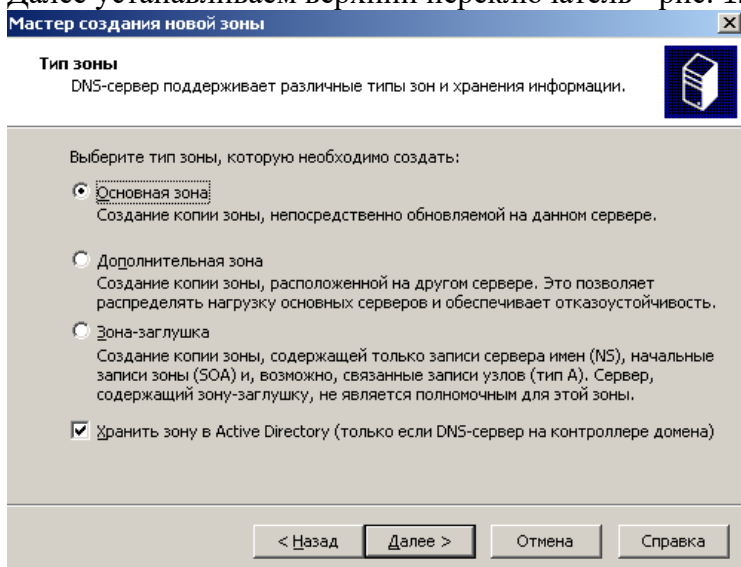
### Создание зоны обратного просмотра

Щелкните на строчку **Зона обратного просмотра** и выберите команду **Создать новую зону** (рис. 9).



**Рис. 9.** Окно мастера создания новой зоны

Далее устанавливаем верхний переключатель - рис. 12.

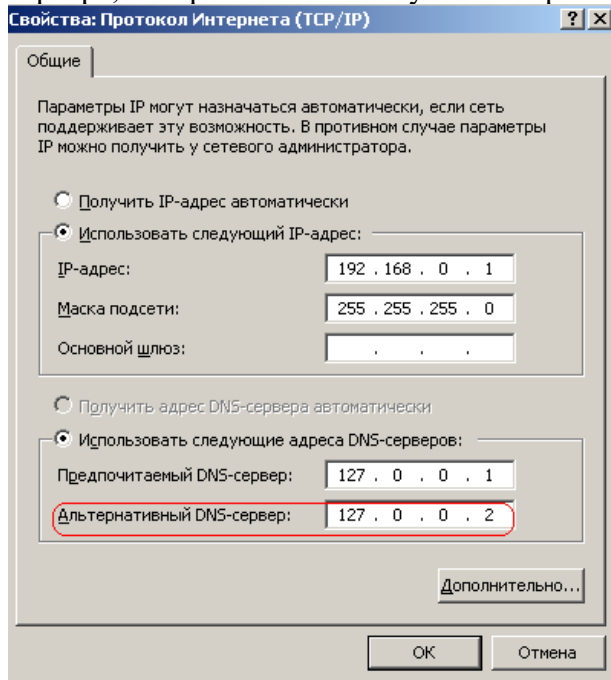


**Рис. 10.** Устанавливаем переключатель Основная зона

### Примечание

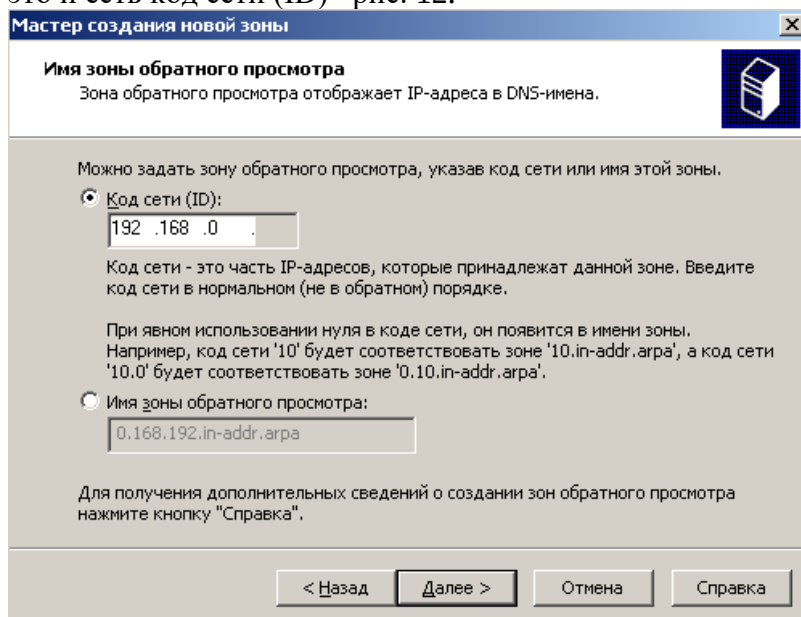
**Основная зона** устанавливается на основной сервер (она - главная), **Дополнительная зона** необходима для резервирования и разгрузки основного сервера. Если загрузка первого DNS сервера велика (или он отключился), то часть запросов можно отправить на второй, альтернативный

DNS и отказоустойчивость системы повышается ( рис. 11). Зона - заглушка содержит IP адрес сервера, который может обслужить запрос.



**Рис. 11.** Пример использования альтернативного DNS сервера

Наша сеть 192.168.0.1 относится к классу сетей C, поэтому значение 192.168.0 мы менять не можем - это и есть код сети (ID) –рис. 12.

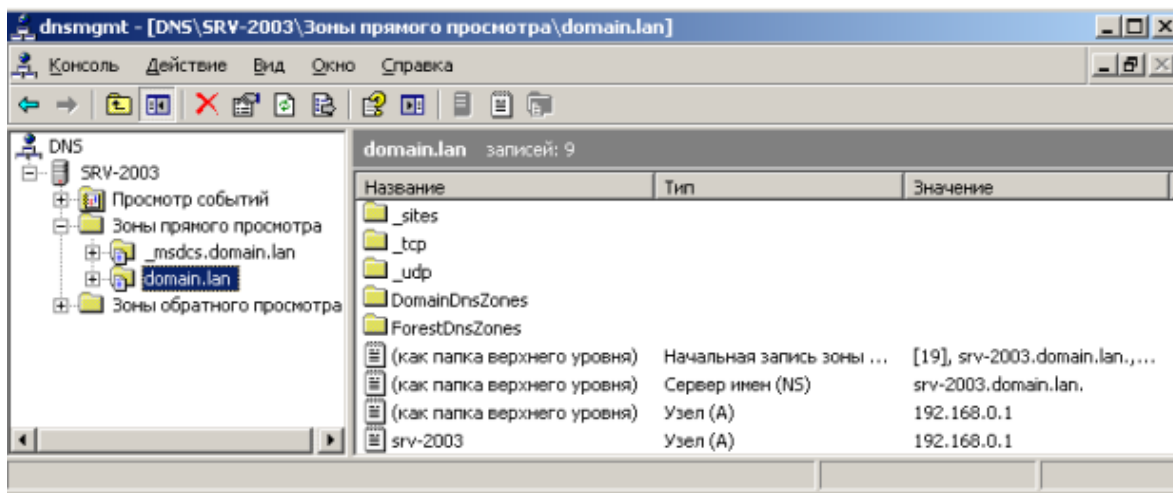


**Рис. 12.** Задаем код сети

Зона обратного просмотра создана. После подключения первого ПК здесь появится соответствие IP адреса ПК его имени.

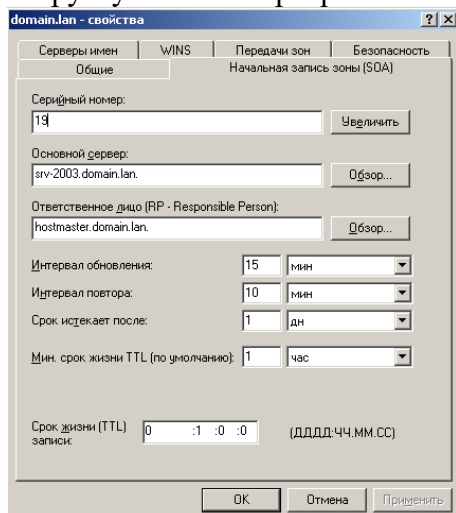
### Записи ресурсов DNS

Вся DNS состоит из этих RR записей, по которым пользователь может искать ресурсы в сети. Запустим консоль управления DNS и зайдем в зону прямого просмотра ( рис. 13).



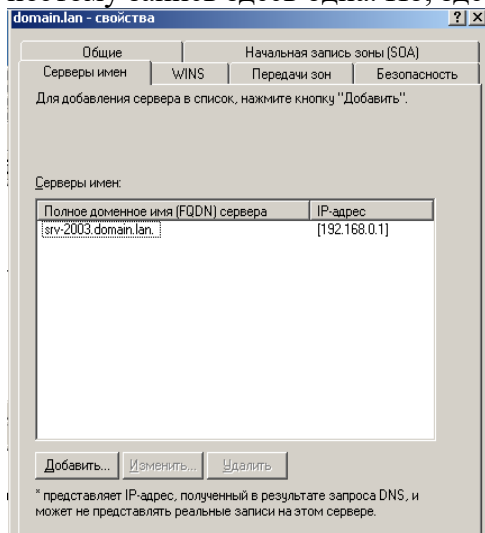
**Рис. 13.** Зоны прямого просмотра

Выполним двойной щелчок на строчке **Начальная запись зоны** (рис. 14). В данном окне наиболее интересный параметр **Срок жизни (TTL)**. Предположим, что компьютер с именем ПК-1 и IP адресом 192.168.0.1 хочет соединиться с именем ПК-2 и IP адресом 192.168.0.2. Он выдает запрос на DNS сервер и тот сообщает, что с компьютер с именем ПК-2 имеет IP адрес 192.168.0.2. Эта запись кэшируется (помещается в память) на **Срок жизни (TTL)**. Подобный подход к запросам снижает нагрузку на DNS сервер.



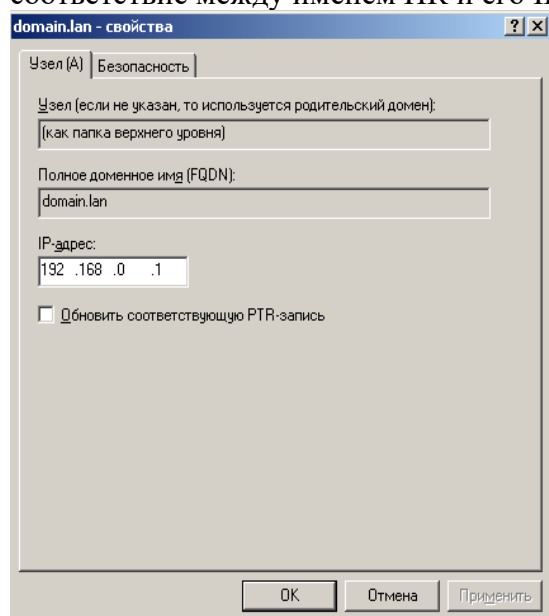
**Рис. 14.** Вкладка Начальная запись зоны

Теперь выполним двойной щелчок на строчке **Сервер имен** (рис. 15). У нас DNS сервер один, поэтому запись здесь одна. Но, здесь записей может быть столько, сколько имеется DNS серверов.



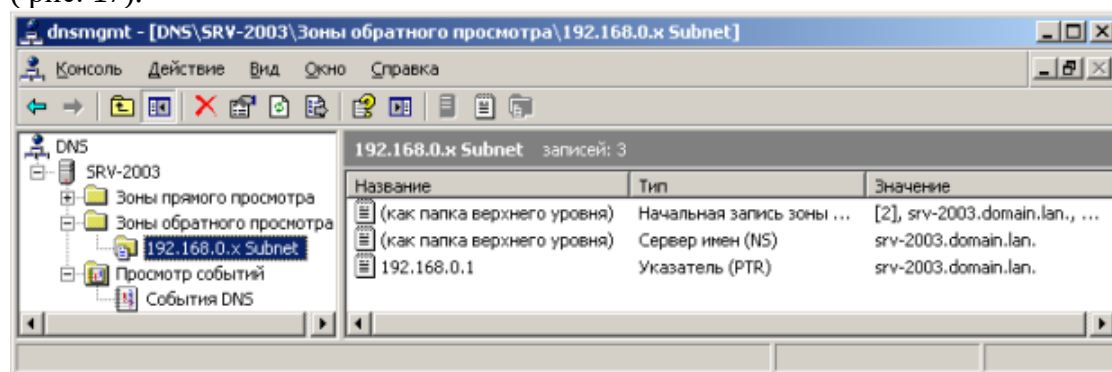
**Рис. 15.** Вкладка Серверы имен

Выполним двойной щелчок на строчке **Узел А** ( рис. 16). Эта запись устанавливает прямое соответствие между именем ПК и его IP адресом.



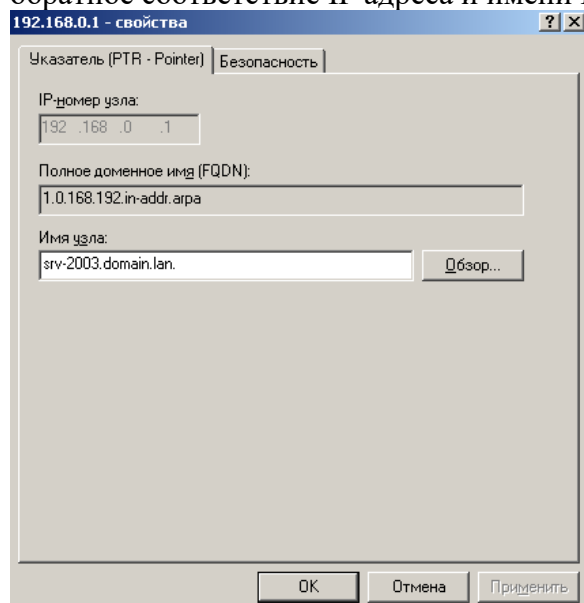
**Рис. 16.** Вкладка Узел А

Теперь изучим записи зоны обратного просмотра. После перезагрузки ПК здесь будет три записи ( рис. 17).



**Рис. 17.** Записи зоны обратного просмотра

Двойным щелчком мыши зайдем в **Указатель (PTR)** – рис. 18. Как видим, именно здесь задается обратное соответствие IP адреса и имени ПК.



**Рис. 18.** Вкладка Указатель (PTR)

## Проверка работы зон прямого и обратного просмотра

Далее вызовем командную строку и пропингуем наш сервер ( рис. 19). Видим, что зона прямого просмотра работает нормально и имени SRV-2003 ставится в соответствие IP адрес 192.168.0.1.

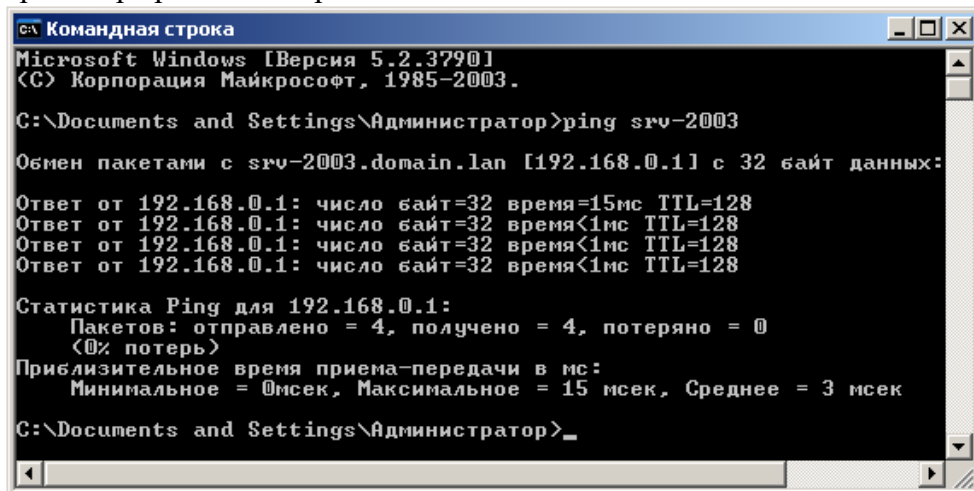


Рис. 19. Окно Командная строка

Если пропинговать не имя, а IP адрес и использовать ключ "-а", то увидим, что зона обратного просмотра также работает хорошо (рис. 20). Обмен данными идет нормально.

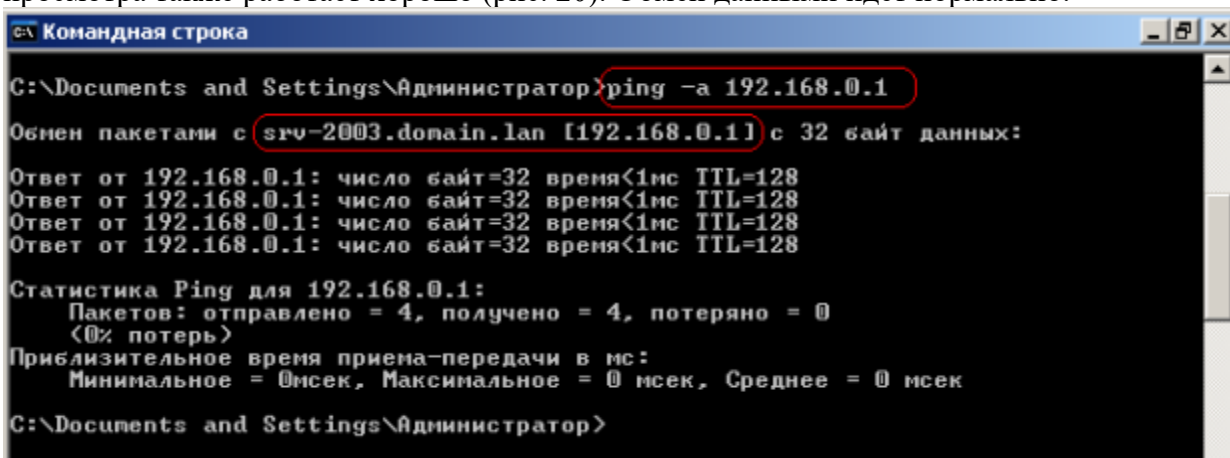


Рис. 20. Зона обратного просмотра работает хорошо

### Контрольные вопросы:

1. Какую функцию выполняет DNS сервер?
2. В чем отличие статического IP адреса от динамического?
3. Для чего нужна дополнительная зона?

### ОТЧЕТ

Сохранить работу.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала          | Творческая составляющая                     | Перевод баллов в оценку                                             |
|-------------------------------------------------|----------------------------------------|---------------------------------------------|---------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов | Защита работы аудиторно – 20 баллов         | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10    | Защита своей работы персонально – 10 баллов | Менее 20 баллов – оценка «2»                                        |

|                                             |                                                  |                                                           |  |
|---------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|--|
|                                             | баллов                                           |                                                           |  |
| Не своевременно выполнена работа – 0 баллов | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |  |

## Лабораторное занятие №7

**Тема:** Формирование аппаратных требований и схемы банка данных (3 часа)

### Цель занятия:

1. Научится определять аппаратные требования и схемы банков данных.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** Компьютер, операционная система Windows, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

**Краткие теоретические и учебно-методические материалы по теме практической работы:**

*Сервер* - в локальных вычислительных сетях - специализированная ЭВМ, управляющая использованием разделяемых между терминалами сети дорогостоящих ресурсов системы.

*Сервер* (англ. *server* от англ. *to serve* — служить) — в информационных технологиях — программный компонент вычислительной системы, выполняющий сервисные функции по запросу клиента, предоставляя ему доступ к определённым ресурсам.

*Сервер сети* (*Server*) - это компьютер, подключенный к сети и предоставляющий пользователям сети определенные услуги, например, хранение данных общего пользования, печать заданий, обработка запроса к СУБД, удаленная обработка заданий и т.д. Сервер работает по заданиям клиентов. После выполнения задания сервер посылает полученные результаты клиенту, инициировавшему это задание.

Обычно связь между клиентом и сервером поддерживается посредством передачи сообщений, и при этом используется определенный протокол для кодирования запросов клиента и ответов сервера. Виды серверов: FTP; Файловый; Web; Телефонный; Терминальный; Факс; Суперсервер и т.д.

1. *Файл-серверы* представляют собой серверы для обеспечения доступа к файлам на диске сервера. Прежде всего это серверы передачи файлов по заказу, по протоколам **FTP** и **HTTP**. Протокол **HTTP** ориентирован на передачу текстовых файлов, но серверы могут отдавать в качестве запрошенных файлов и произвольные данные, например динамически созданные веб-страницы, картинки, музыку и т. п. *Другие серверы* позволяют монтировать дисковые разделы сервера в дисковое пространство клиента и п

*FTP-сервер* - это понятие, за которым скрывается обычный компьютер. Но так как он содержит общедоступные файлы и настроен на поддержку протокола **FTP**, то его называют сервером - поставщиком информации. *FTP-клиент* - это сервисная программа, с помощью которой можно произвести соединение с **FTP** сервером. Обычно эта программа имеет командную строку, но некоторые имеют оконный интерфейс и не требуют запоминания команд. *WEB-сервер* необходим для обслуживания WEB-страниц вашего сайта

Доступ к WEB-серверу имеет пять уровней:

- Общедоступный с возможностью только чтения всех **URL** за исключением тех, что помещены в каталогах */private*.
- Доступ сотрудников организации, которой принадлежит сервер. Здесь также допустимо только чтение, но доступны и секции каталога */private*.
- Разработчики **WEB-сервера**. Имеют возможность модифицировать содержимое сервера, устанавливать CGI-скрипты, прерывать работу сервера.

- Администраторы узла (сервера). Имеют те же привилегии, что и разработчики, но могут также реконфигурировать сервер и определять категорию доступа.
- Системные администраторы. Имеют идентичные привилегии с администраторами сервера.

**Оснастка Internet Information Service (IIS)** обеспечивает средства управления сервером для контроля над доступом и содержимым веб-узлов и узлов **FTP**. Например, разработчикам это средство позволит выполнить доскональную проверку работы узла перед окончательной загрузкой на сервер интрасети организации или Интернета. Оснастка **IIS** имеет следующие особенности:

1. дополнительные параметры настройки сервера, в частности, для управления узлом **FTP**, независимого выполнения приложений, настройки типов **MIME** и назначения дополнительных средств обработки сценариев.
2. мастер создания виртуальных каталогов.
3. возможность управления установками **Internet Information Services** в сети.

На сегодняшний день существует огромное множество программного обеспечения для работы с протоколом **FTP** под все операционные системы. Все это множество программного обеспечения можно разделить на две части: *серверное ПО* и *клиентское ПО*. *Серверное ПО* служит для создания и управления ftp-сервером. *Клиентское ПО* используется для просмотра ресурсов на ftp-сервере. Этот класс программ призван обеспечить комфортную работу с удаленными ресурсами. Сюда относятся такие программы как:

1. **ftp.exe** – стандартное приложение **Windows**;
2. **FileZilla** – мощный ftp-клиент с открытым исходным кодом (т.е. при желании вы можете что-нибудь новое добавить в эту программу самостоятельно);
3. **RigthFTP, CuteFTP** – графические ftp-клиенты;
4. **Total commander** (или любой другой с интерфейсом **Norton Commander**)– имеет встроенный ftp клиент;
5. **Explorer.exe** – стандартное приложение **Windows**;
6. Любой браузер.

#### **Задания для практического занятия:**

##### **Задание 1. Подготовьте файловый сервер.**

1. Подключите к виртуальной машине **VM-2** образ установочного диска **win2003-2.iso**.
2. Запустите виртуальную машину **VM-2**.
3. Добавьте новую **роль** серверу – *Файл-сервер*:
  - а. откройте диалоговое окно **Управление данным сервером** (*Пуск/администрирование/Управление Данным Сервером*);
  - б. активизируйте добавление ролей кнопкой *Добавить или удалить роль*;
  - с. выберите **Файловый сервер** и щелкните *Далее*;
  - д. установите параметры файлового сервера:
    - i. Предоставить доступ UNIX-системам к файлам;
    - ii. Предоставить доступ Apple--системам к файлам;
    - iii. подтвердите введенные параметры кнопкой *Далее*;
  - е. запустите установку роли сервера кнопкой *Далее*.
4. Перезагрузите виртуальный компьютер кнопкой *Перезагрузить*.
5. Откройте диалоговое окно **Настройки файлового сервера** (*Пуск/администрирование/Управление Данным Сервером/Управление этим файловым сервером*).
6. Установите стандартные квоты использования места на диске:
  - а. установите флажок *Установить дисковые квоты по умолчанию для новых пользователей данного сервера*;
  - б. укажите **размер квот - 50Мб**;
  - с. установите **предупреждение о квоте - 40Мб**;
  - д. установите флажок *Не выделять место на диске при превышении дискового пространства*;
  - е. завершите ввод стандартных квот кнопкой *Далее*.
7. Откажитесь от включения службы индексирования.

8. Укажите папку на сервере, для хранения файлов, например **C:\Documents and settings\Администратор\Рабочий стол\PUB**.
9. Далее мастер установки завершит свою работу. Попробуйте теперь зайти на созданную вами сетевую папку с другого компьютера сети. Обратите внимание на способ подключения. Попробуйте заполнить папку для превышения квоты.

## Задание 2. Настройте Web-сервер.

- Установите **Internet Information Service (IIS)** (*Пуск/администрирование/Управление Данным Сервером/Сервер приложений IIS*):
- Подготовьте тестовую страницу:
  - создайте временную страницу, вызываемую по умолчанию: наберите в **Блокноте** и сохраните в файле с именем **Default.html** в каталоге **\Inetpub\wwwroot**.

```
<HTML>
  <HEAD>
    <TITLE>Тестовая страница</TITLE>
  </HEAD>
  <BODY>
    <H1 align="center">Тестовая страница</H1>
    <P align="center">
      <FONT color="green">
        На этом месте будет размещена страница нашей организации. В настоящий момент сайт находится в стадии разработки
      </FONT>
    </P>
  </BODY>
</HTML>
```

- Настройте **Web-сервер**:
  - откройте консоль управления сервером **IIS** (*Пуск/администрирование/Управление Данным Сервером/Управление этим сервером приложений*);
  - перейдите к web-узлу, заданному по умолчанию (*Диспетчер служб IIS/Веб-узлы/Веб-узел по умолчанию*);

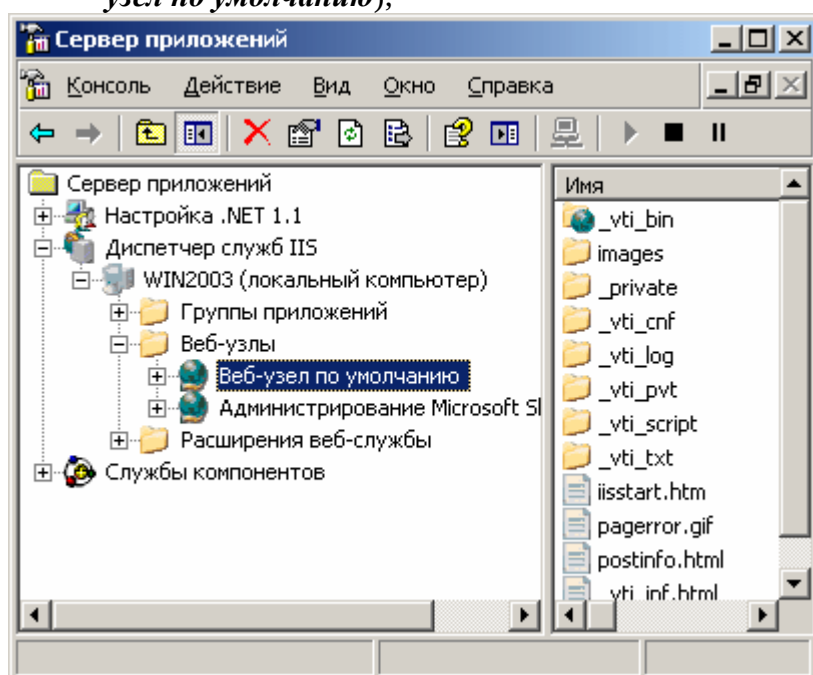


Рисунок 1. Консоль управления сервером приложений (IIS).

- откройте диалоговое окно **Свойства узла по умолчанию** (*контекстное меню/Свойства*);
- добавьте страницу по умолчанию:

- перейдите на вкладку **Документы**;
  - установите флажок *Задать страницу содержания по умолчанию*;
  - откройте окно добавления кнопкой **Добавить**;
  - введите в поле **Default.html**;
  - подтвердите добавление кнопкой **ОК**.
- закройте окно свойств кнопкой **ОК**.
- Проверьте настройку **Web-сервера**:
  - на вашем компьютере откройте **Internet Explorer** (*Пуск/Программы/Internet Explorer*);
  - наберите в адресной строке **http://127.0.0.1/**;
  - сделайте скриншот происходящего на экране и сохраните его в своей папке.

### Задание 3. Установите и настройте сервер FTP

- Установите сервер FTP - **FileZilla**.
- Запустите **FileZilla Server Interface**.

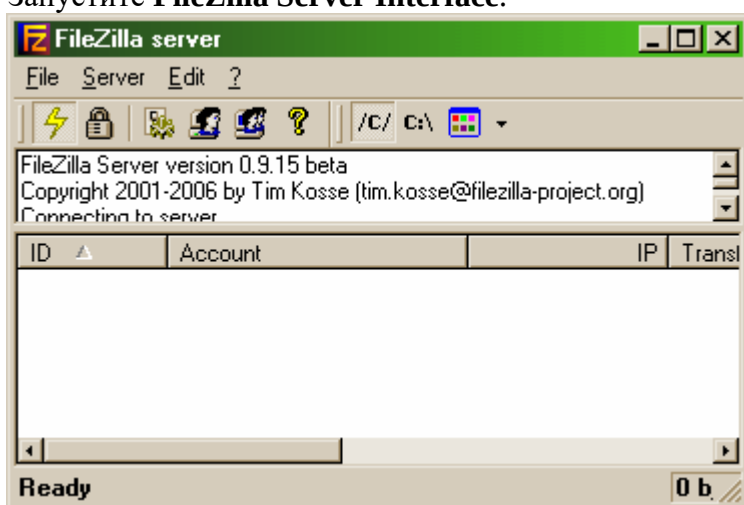


Рисунок 2. Интерфейс управления FTP-сервером **FileZilla**

- Ограничьте количество одновременных подключений к серверу:
  - откройте окно настройки сервера (**Edit/Settings**);

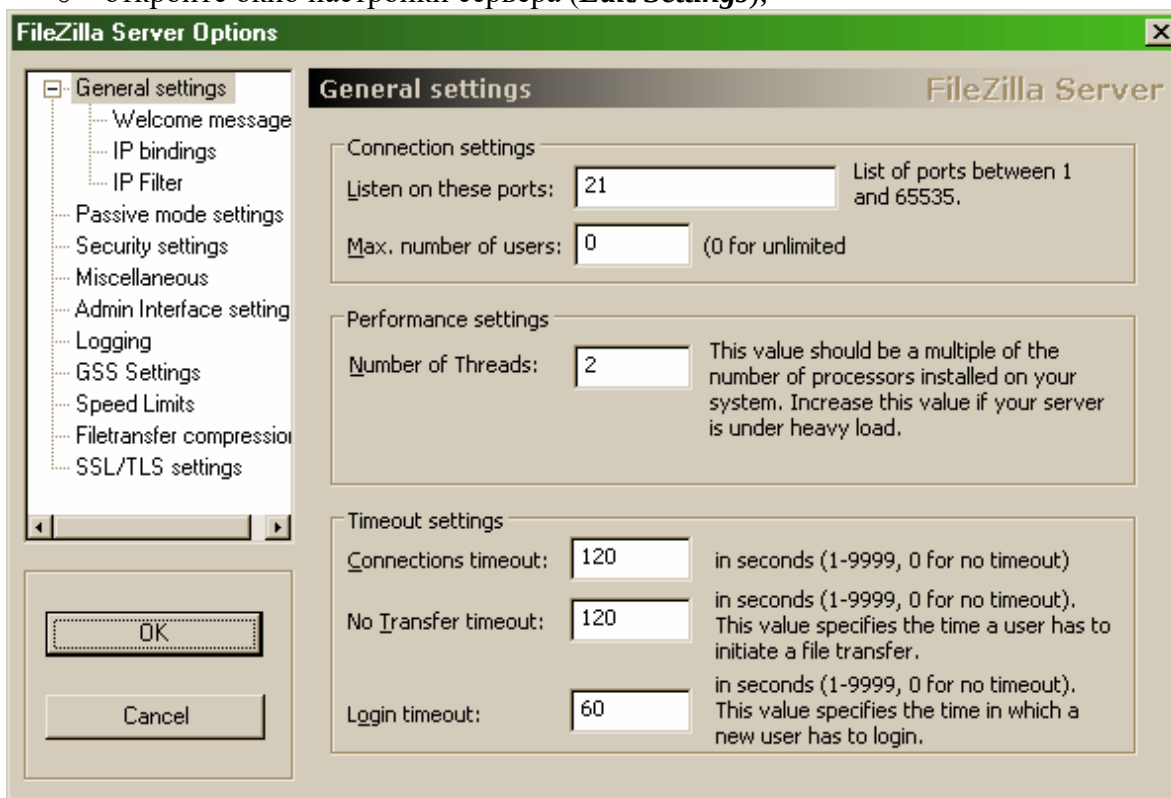


Рисунок 3. Настройки FTP-сервера

- перейдите в раздел **General Settings** (общие настройки);
- введите в поле **Max.number of users** – 2;
- Установите текст приветствия:
  - перейдите в раздел **Welcome message**;
  - введите в поле **Custom welcome message** – *Добро пожаловать на мой сервер*;
  - Установите ограничения по скорости:
    - перейдите в раздел **Speed Limits** (ограничения скорости);
    - включите использование правил ограничения скорости радиокнопкой *Use Speed Limit rules*;
    - добавьте ограничение по скорости **не более 3 Кб/с в понедельник**;
    - откройте окно задания параметров ограничений кнопкой **Add** (Добавить);
    - сбросьте все флажки кроме *Monday* (Понедельник);
    - введите в поле **Speed** – 3;
    - подтвердите ввод данных кнопкой **OK**;
    - примените параметры кнопкой **OK**.
  - Создайте группы пользователей **FTP-сервера**:
    - откройте диалоговое окно **добавления групп** кнопкой на панели инструментов;
    - активируйте добавление групп кнопкой **Add** (Добавить);
    - введите **имя группы**, например *Students* (**OK**);
    - задайте общую папку для созданной группы:
      - перейдите в раздел **Shared Folders** (Общие папки);
      - активируйте добавление папок кнопкой **Add** (Добавить);
      - укажите общую папку, например *C:\Documents and settings\Администратор\Рабочий стол* и подтвердите выбор кнопкой **OK**;
    - разрешите чтение и удаление содержимого общей папки – установите флажок *Write u Delete*;
    - завершите добавление групп пользователей кнопкой **OK**.
  - Добавьте нового пользователя:
    - откройте диалоговое окно добавления пользователей кнопкой на панели инструментов;
    - активируйте добавление пользователей кнопкой **Add** (Добавить);
    - введите **имя группы**, например *justuser*;
    - выберите в списке **User should be member of the following group** созданную ранее группу и подтвердите создание пользователя кнопкой **OK**;
    - установите пароль для созданного пользователя:
    - перейдите на вкладку **General** (Общие);
    - введите в поле **Password** новый пароль, например *123*;
    - завершите добавление групп пользователей кнопкой **OK**.
  - Проверьте работу сервера:
    - запустите командную строку (*Пуск/Программы/Стандартные/Командная строка*);
    - введите команду для подключения к FTP-серверу на текущем компьютере: **FTP 127.0.0.1**
    - введите имя пользователя - *justuser* (**ENTER**);
    - введите пароль - *123* (**ENTER**);
    - просмотрите содержимое домашней папки: **DIR**
    - отключитесь от сервера: **QUIT**
    - закройте командную строку.
  - Закройте интерфейс управления **FTP-сервером**.

**Задание 4. Выполните самостоятельные задания 5-6.** олноценно работать с файлами на них. Это позволяют серверы протоколов **NFS** и **SMB**. Серверы **NFS** и **SMB** работают через интерфейс **RPC**.

#### Недостатки файл-серверной системы:

4. Очень большая нагрузка на сеть, повышенные требования к пропускной способности. На практике это делает практически невозможной одновременную работу большого числа пользователей с большими объемами данных.
5. Обработка данных осуществляется на компьютере пользователей. Это влечет повышенные требования к аппаратному обеспечению каждого пользователя. Чем больше пользователей, тем больше денег придется потратить на оснащение их компьютеров.

6. Блокировка данных при редактировании одним пользователем делает невозможной работу с этими данными других пользователей.
7. Безопасность. Для обеспечения возможности работы с такой системой Вам будет необходимо дать каждому пользователю полный доступ к целому файлу, в котором его может интересовать только одно поле

Файловый сервер выполняет следующие функции:

2. хранение данных,
3. архивирование данных,
4. согласование изменений данных, выполняемых разными пользователями, передача данных.

### Отчет

Сохраните работу.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Тема 1.3. Администрирование баз данных и серверов

### Лабораторное занятие №8

**Тема:** Установка и настройка сервера MySQL (2 часа)

### Цель занятия:

1. Научится устанавливать и настраивать сервер баз данных MySQL в операционной системе Windows
- 2.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

##### Начало загрузки

1. Перейдите по адресу <http://dev.mysql.com/downloads/installer/>.
2. Нажмите кнопку 'Загрузить'.
3. Сохраните файл программы установки на вашем компьютере.

##### Начало установки

После завершения загрузки запустите программу установки следующим образом.

1. Щелкните правой кнопкой мыши загруженный установочный файл (например, mysql-installer-community-5.6.14.0.msi) и выберите пункт 'Выполнить'. Запустится программа установки MySQL.
2. На панели приветствия выберите 'Установить продукты MySQL'.

3. На панели информации о лицензии ознакомьтесь с лицензионным соглашением, установите флажок принятия и нажмите кнопку 'Далее'.
4. На панели 'Найти последние продукты' нажмите кнопку 'Выполнить'. После завершения операции нажмите кнопку 'Далее'.
5. На панели "Тип настройки" выберите параметр "Пользовательская", а затем нажмите кнопку "Далее".
6. На панели 'Выбор компонентов обеспечения' убедитесь, что выбран MySQL Server 5.6.x, и нажмите кнопку 'Далее'.
7. На панели 'Проверить требования' нажмите кнопку 'Далее'.
8. На панели 'Установка' нажмите кнопку 'Выполнить'. После успешного завершения установки сервера на панели 'Установка' отображается информационное сообщение. Нажмите кнопку "Далее".
9. На странице 'Настройка' нажмите кнопку 'Далее'.
10. На первой странице конфигурации сервера MySQL (1/3) установите следующие параметры:

- **Тип конфигурации сервера.** Выберите вариант 'Компьютер для разработки'.
- **Включите поддержку сети TCP/IP.** Убедитесь, что флажок установлен, и задайте следующие параметры ниже:
  - **Номер порта.** Укажите порт подключения. По умолчанию установлено значение 3306; не следует изменять его без необходимости.
  - **Откройте порт брандмауэра для доступа к сети.** Выберите исключение добавления брандмауэра для указанного порта.
- **Расширенная настройка.** Выберите флажок 'Показать расширенные параметры' для отображения дополнительной страницы конфигурации для настройки расширенных параметров для экземпляра сервера (если требуется).

**Примечание.** При выборе этого параметра необходимо перейти к панели для установки параметров сети, где будет отключен брандмауэр для порта, используемого сервером MySQL.

11. Нажмите кнопку "Далее".

12. На второй странице конфигурации сервера MySQL (2/3) установите следующие параметры:

- **Пароль учетной записи root.**
  - **Пароль root для MySQL.** Введите пароль пользователя root.
  - **Повторите ввод пароля.** Повторно введите пароль пользователя root.

**Примечание.** Пользователь root - это пользователь, который имеет полный доступ к серверу баз данных MySQL - создание, обновление и удаление пользователей и так далее. Запомните пароль пользователя root (администратора) – он понадобится вам при создании примера базы данных.

- **Учетные записи пользователя MySQL.** Нажмите кнопку 'Добавить пользователя' для создания учетной записи пользователя. В диалоговом окне 'Сведения о пользователе MySQL' введите имя пользователя, роль базы данных и пароль (например, !phruser). Нажмите кнопку "ОК".

Нажмите кнопку "Далее".

13. На третьей странице конфигурации сервера MySQL (3/3) установите следующие параметры:

- **Имя службы Windows.** Укажите имя службы Windows, которая будет использоваться для экземпляра сервера MySQL.
- **Запустите сервер MySQL при запуске системы.** Не снимайте этот флажок, если сервер MySQL требуется для автоматического запуска при запуске системы.
- **Запуск службы Windows в качестве.** Возможны следующие варианты.
  - **Стандартная системная учетная запись.** Рекомендуются для большинства сценариев.
  - **Нестандартный пользователь.** Существующая учетная запись пользователя рекомендуется для сложных сценариев.

Нажмите кнопку "Далее".

14. На странице 'Обзор конфигурации' нажмите кнопку 'Далее'.

15. После успешного завершения настройки на панели 'Завершение' появляется информационное сообщение. Нажмите кнопку "Завершить".

**Примечание.** Для проверки успешности настройки запустите диспетчер задач. Если MySQLd-nt.exe присутствует в списке 'Процессы', сервер базы данных запущен.

#### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### Лабораторное занятие №9

**Тема:** Установка и настройка сервера под UNIX (2 часа)

#### Цель занятия:

1. Научится устанавливать и настраивать сервера под UNIX

#### ИНСТРУКЦИОННАЯ КАРТА

##### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Unix

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

##### Практические задания

Задаем пароль для пользователя root следующей командой:

```
passwd root
```

Создаем нового пользователя для работы с сервером.

а) если Linux CentOS:

```
useradd kdo -G wheel -m
```

\* где **kdo** — имя учетной записи; **wheel** — группа, которая позволяет входить в систему в привилегированном режиме (**su**).

б) если Linux Ubuntu:

```
useradd kdo -G sudo -m
```

\* где **kdo** — имя учетной записи; **sudo** — группа, которая позволяет запускать команды в привилегированном (**sudo**) режиме.

в) если BSD:

```
pw useradd kdo -G wheel -m
```

После того, как создали пользователя, обязательно задаем пароль:

```
passwd kdo
```

## 2. Имя сервера

Linux:

```
vi /etc/hostname
```

```
server.dmosk.ru
```

или одной командой:

```
hostnamectl set-hostname server.dmosk.ru
```

BSD:

ee /etc/rc.local

hostname="server.dmosk.ru"

### 3. Обновление

#### Ручное

CentOS / Red Hat

yum update

Ubuntu / Debian

apt-get update && apt-get upgrade

FreeBSD

pkg update && pkg upgrade

#### Автоматическая загрузка обновлений

Для удобства обновления системы, установим следующий пакет.

а) если Linux CentOS:

yum install yum-cron

б) если Linux Ubuntu:

apt-get install cron-apt

После установки данного пакета, будет создано задание в cron.daily, которое будет искать обновления для системы и загружать их. В будущем, это сократит время ожидания на скачивание файлов.

### 4. Правильное время

Настраиваем временную зону:

timedatectl set-timezone Europe/Moscow

*\* в данном примере мы задаем зону по московскому времени. Список все доступных зон можно посмотреть командой **timedatectl list-timezones**.*

Устанавливаем утилиту для синхронизации времени, разрешаем запуск демона и запускаем его.

а) если на системе **CentOS / Red Hat**:

yum install chrony

systemctl enable chronyd --now

б) если на системе **Ubuntu / Debian**:

apt-get install chrony

systemctl enable chrony

### 5. Автозавершение ввода команд

Значительно упрощает работу с консолью. Выполняется только на Linux — на BSD настроен по умолчанию.

Открываем на редактирование следующий файл:

vi /etc/inputrc

И добавляем:

set show-all-if-ambiguous On

"\e[A": history-search-backward

"\e[B": history-search-forward

Выходим из системы и входим снова для применения настроек.

### 6. Отключение IPv6

Если мы не используем IPv6, его присутствие, в некоторых случаях, может мешать работе, и даже, приводить к сбоям.

Отключение IPv6 для различных систем читайте в статье Как отключить IP версии 6 в Linux.

### 7. Цветовая схема vi (для Ubuntu)

Если мы планируем использовать текстовый редактор vi и у нас Ubuntu, рекомендуется сменить цветовую схему — по умолчанию, используется синий цвет, и если консоль будет черной (как большинство таковых по умолчанию), то будет неудобно читать текст.

Открываем конфигурационный файл vi.

а) для настройки всем пользователям:

vi /etc/vim/vimrc

б) только для текущего:

```
vi ~/.vimrc
```

Добавим строку:

```
color desert
```

\* где **desert** — устанавливаемый цвет.

## 8. Настройка SSH (опционально)

Все настройки выполняем в файле:

```
vi /etc/ssh/sshd_config
```

### Безопасность

Отключаем SSH-вход от пользователя root:

```
PermitRootLogin no
```

\* в системе должна быть другая учетная запись, под которой можно войти по SSH. В противном случае, возможность удаленного управления будет потеряна.

Можно даже явно указать пользователей и группы, которым можно подключаться удаленно:

```
AllowGroups wheel developer
```

```
AllowUsers kdo operator
```

При необходимости, заставляем SSH слушать на определенном адресе или сетевом порту:

```
Port 4444
```

```
ListenAddress 192.168.0.15
```

### Производительность

Немного увеличим производительность:

```
UseDNS no
```

```
GSSAPIAuthentication no
```

```
GSSAPICleanupCredentials yes
```

Для применения настроек перезагружаем службу:

```
systemctl restart sshd
```

или

```
service sshd restart
```

## 9. Настройка хранения истории команд (опционально)

По умолчанию, history возвращает историю команд в неудобном формате — без возможности увидеть дату и время. Чтобы исправить ситуацию, открываем на редактирование следующий файл:

```
vi /etc/environment
```

И добавляем следующие строки:

```
export HISTTIMEFORMAT='%F - %H:%M:%S '
```

```
export HISTSIZE=1000
```

```
export HISTFILESIZE=1000
```

\* где **HISTTIMEFORMAT** добавит перед каждой командой дату и время ее выполнения; **HISTSIZE** — количество команд, которые хранятся в истории; **HISTFILESIZE** — количество команд, хранящихся в файле с историей.

### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала              | Творческая составляющая                     | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------|---------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов     | Защита работы аудиторно – 20 баллов         | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов | Защита своей работы персонально – 10 баллов |                                                                                                     |
| Не своевременно                                 | Полное                                     | Оформление своей                            |                                                                                                     |

|                                |                                                    |                                             |  |
|--------------------------------|----------------------------------------------------|---------------------------------------------|--|
| выполнена работа – 0<br>баллов | содержание<br>выполненной<br>работы – 10<br>баллов | работы несколькими<br>способами – 20 баллов |  |
|--------------------------------|----------------------------------------------------|---------------------------------------------|--|

### Лабораторное занятие №10

**Тема:** Выполнение запросов к базе данных (2 часа)

#### Цель занятия:

1. Научиться создавать запросы простые и сложные к готовой базе данных

#### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** Компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

##### Задание 1. Создание базы данных

1. Любым, изученным, способом создайте в новой базе данных 2 таблицы **Студент** и **Работник** и заполните их данными по образцу.

Студент

**Код**

**Студент**

**Фамилия**

**Имя**

**Отчество**

**Адрес**

**Номер телефона**

**Специализация**

1

Иванов

Иван

Иванович

г. Тула

457896

администратор

2

Петров

Сергей

Петрович

г. Москва

7458962

технолог

3

Голубева

Ольга

Ивановна

г. Белев

3698521

бухгалтер

4

Соколова

Инна

Олеговна

г. Тула

852967

бухгалтер

5

Мухина

Олеся

Петровна

г. Москва

8625471

технолог

6

Андреева

Анна

Романовна

г. Люберцы

748596

повар

7

Галкина

Дина

Евгеньевна

г. Люберцы

919597

технолог

8

Сорина

Ольга

Сергеевна

г. Москва

9191954

бухгалтер

9

Сидоров

Илья

Владимирович

г. Волгоград

126578

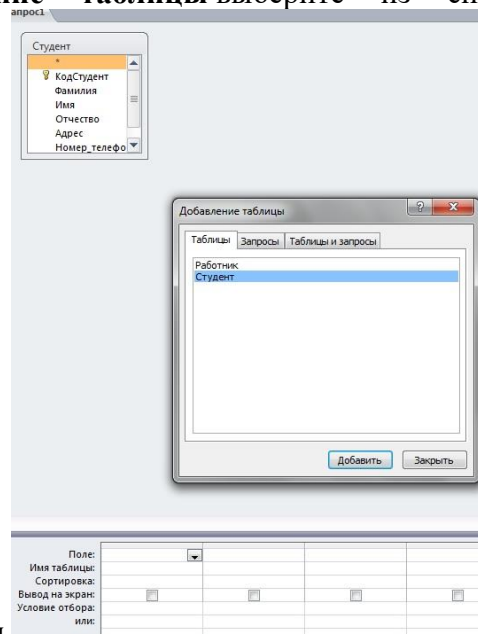
слесарь

Работник

##### Задание 2. Создание запроса на выборку.

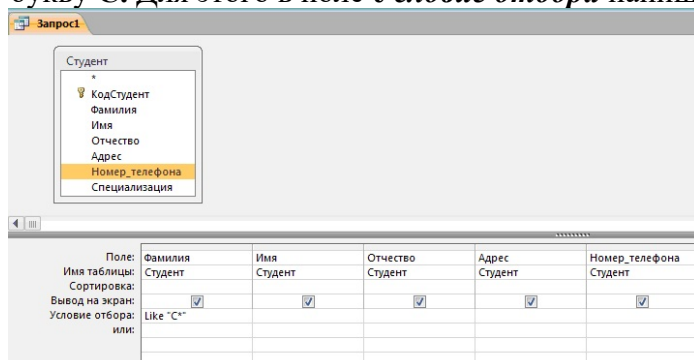
1. Выполните команду **Создание – Конструктор запросов**.

- В появившемся диалоговом окне **Добавление таблицы** выберите из списка имя



таблицы **Студент**, щелкните по кнопке **Добавить**

- Закончите выбор, щелкнув по кнопке **Заккрыть**. Появится возможность выбора полей из таблицы “Студент”. Для этого достаточно дважды щелкнуть по именам полей или перетащить мышью названия полей в клетку запроса.
- Создайте телефонную книгу для всех студентов, фамилии которых начинаются на букву **С**. Для этого в поле **Условие отбора** напишите условие Like “С\*”



- Сохраните запрос, щелкнув по кнопке **Сохранить**. Введите имя запроса **Телефонная книга** и щелкните по кнопке **ОК**.
- Щелкните по кнопке **Выполнить** для представления запроса. Закройте запрос.
- Убедитесь в правильности полученного запроса, щелкнув по имени запроса **Телефонная книга** слева в окне **Все объекты Access**. Закройте таблицу.
- Создайте запрос на выборку тех студентов, которые приехали из Москвы или Люберцы.
- Для этого выполните команду **Создание – Конструктор запросов**.
- В появившемся диалоговом окне **Добавление таблицы** выберите из списка имя таблицы **Студент**, щелкните по кнопке **Добавить**
- Закончите выбор, щелкнув по кнопке **Заккрыть**. Появится возможность выбора полей из таблицы “Студент”. Для этого достаточно дважды щелкнуть по именам полей или перетащить мышью названия полей в клетку запроса.
- В поле **Условие отбора** напишите условия для поля **Адрес** так, как показано на рисунке

Адрес

Студент

- КодСтудент
- Фамилия
- Имя
- Отчество
- Адрес
- Номер\_телефона
- Специализация

| Поле:           | Фамилия                             | Имя                                 | Отчество                            | Адрес                               |
|-----------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Имя таблицы:    | Студент                             | Студент                             | Студент                             | Студент                             |
| Сортировка:     |                                     |                                     |                                     |                                     |
| Вывод на экран: | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Условие отбора: |                                     |                                     |                                     | "г. Москва"                         |
| или:            |                                     |                                     |                                     | г. Люберцы                          |

13. Сохраните запрос, щелкнув по кнопке **Сохранить**. Введите имя запроса **Адрес** и щелкните по кнопке **ОК**.

14. Щелкните по кнопке **Выполнить** для представления запроса. Закройте запрос.

#### Самостоятельное задание.

1. Составьте запрос на выборку тех студенток, имя которых – Ольга.
2. Составьте запрос на выборку работников организаций, названия которых начинаются на букву **Р**, используя таблицу **Работник**.
3. Составьте запрос на выборку всех студентов, которые обучаются по специальности технолога.
4. Составьте запрос на выборку работников организаций, которые работают по должности **инженер** или **бухгалтер**.
5. Результаты предъявите учителю.

#### Задание 3. Завершение работы с программой Access.

1. Выполните команду *Файл – Выход*.
2. Если вы производили редактирование в базе данных, появится вопрос о сохранении изменений. Ответьте утвердительно.

#### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### Лабораторное занятие №11

**Тема:** Выполнение изменений в базе данных, создание триггеров (2 часа)

#### Цель занятия:

1. Сформировать знания о создании и изменении триггера; привести примеры практического использования триггера;

#### ИНСТРУКЦИОННАЯ КАРТА

## **Материально - техническое оснащение**

**Оборудование:** компьютер, выход в Интернет

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

### **Практические задания**

Триггер представляет собой процедуру, которая находится на сервере БД и вызывается автоматически при модификации записей БД, т. е. при изменении столбцов или при их удалении и добавлении. В отличие от хранимых процедур, триггеры нельзя вызывать из приложения клиента, а также передавать им параметры и получать от них результаты.

Триггер по своей сути похож на обработчики событий BeforeEdit, AfterEdit, BeforeInsert, AfterInsert, BeforeDelete и AfterDelete, связанных с модификацией таблиц. Триггер может вызываться при редактировании, добавлении или удалении записей до и/или после этих событий.

### **Замечание**

Изменения, внесенные триггером в транзакции, которая оказалась отмененной, также отменяются.

Триггеры обычно используются для программной реализации так называемых бизнес-правил. С их помощью удобно реализовывать различные ограничения, например, ограничения на значения столбцов или ограничения ссылочной целостности, а также выполнять такие действия, как накопление статистики, работы с БД или резервное копирование записей.

### **Создание и изменение триггера**

Создание триггера выполняется инструкцией CREATE TRIGGER, имеющей формат:

CREATE TRIGGER <Имя триггера> FOR <Имя таблицы>

[ACTIVE | INACTIVE]

{BEFORE | AFTER}

{UPDATE | INSERT | DELETE}

[POSITION <Число>]

AS <Тело триггера>

Описатели ACTIVE и INACTIVE определяют активность триггера сразу после его создания. По умолчанию действует ACTIVE, и созданный триггер активен, т. е. при наступлении соответствующего события будет выполняться. Если триггер неактивен, то при наступлении соответствующего события он не вызывается. Ранее созданный триггер можно активизировать или, наоборот, деактивизировать.

Описатели BEFORE и AFTER задают момент начала выполнения триггера до или после наступления соответствующего события, связанного с изменением записей.

Описатели UPDATE, INSERT и DELETE определяют, при наступлении какого события вызывается триггер — при редактировании, добавлении или удалении записей соответственно.

Для одного события можно создать несколько триггеров, каждый из которых будет автоматически выполнен (если находится в активном состоянии). При наличии нескольких триггеров порядок их вызова (выполнения) определяет число, указанное в операнде POSITION. Триггеры выполняются в порядке возрастания этих чисел.

Созданный триггер можно удалить или изменить. Удаляется триггер инструкцией

DROP TRIGGER <Имя триггера>.

Изменение триггера выполняется инструкцией ALTER TRIGGER, формат которой не отличается от формата инструкции создания триггера. После выполнения инструкции ALTER TRIGGER предыдущее описание триггера с указанным именем заменяется на новое.

Программирование триггера аналогично программированию хранимой процедуры, для чего используется специальный язык, позволяющий также создавать хранимые процедуры.

Отметим, что для доступа к значениям столбца используются конструкции формата:

OLD.<Имя столбца>

NEW.<Имя столбца>

Первая из них позволяет обратиться к старому (до внесения изменений), а вторая — к новому (после внесения изменений) значениям столбца.

### **Значения OLD и NEW**

Значение OLD.ИмяСтолбца позволяет обратиться к состоянию столбца, имевшему место до внесения изменений, а значение NEW.ИмяСтолбца — к состоянию столбца после внесения изменений.

Например, механизм обеспечения ссылочной целостности "cascade", в случае изменения значения первичного ключа родительской таблицы, на mnemonic языке можно описать так:

```
IF (OLD.PrimaryKeyРодителя <> NEW.PrimaryKeyРодителя) THEN
UPDATE ДочерняяТаблица
SET ForeignKeyДочернейТаблицы = NEW.PrimaryKeyРодителя
WHERE ForeignKeyДочернейТаблицы = OLD.PrimaryKeyРодителя;
```

В качестве примера приведем код триггера для родительской таблицы Realty, который при изменении значения ее первичного ключа будет автоматически изменять значение внешнего ключа дочерней таблицы Lease. Другими словами, если в таблице Realty изменилось значение поля Adr, то триггер изменит значение поля Adr в соответствующей записи таблицы Lease.

```
CREATE TRIGGER UPDAT_REALTY FOR Realty
ACTIVE
BEFORE UPDATE
AS
BEGIN
IF (OLD.Adr <> NEW.Adr)
THEN UPDATE Lease
SET Adr = NEW.Adr
WHERE Adr = OLD.Adr;
END
```

В том случае, если значение в столбце не изменилось, то OLD.ИмяСтолбца будет равно NEW.ИмяСтолбца.

### **Примеры использования триггера**

Рассмотрим использование триггера для реализации ограничений ссылочной целостности и для занесения в ключевые столбцы уникальных значений.

В связи с тем, что InterBase не поддерживает автоинкрементные поля, при создании ключевого столбца, требующего уникальности значений, рекомендуется поступать так:

1. При создании таблицы задать ключевой столбец целочисленного типа.
2. Создать генератор, который при обращении к нему возвращает уникальное целочисленное значение.
3. Создать триггер, который при добавлении к таблице новой записи обращается к генератору и заносит возвращаемое им значение в ключевое поле.
4. Следующий пример иллюстрирует описанную последовательность действий: /\* Создание таблицы \*/

```
5. CREATE TABLE Store
6. (S_Code INTEGER NOT NULL,
7. ... PRIMARY KEY (S_Code));
8. /* Создание генератора */
CREATE GENERATOR GenStore;
SET GENERATOR GenStore TO 1;
/* Создание триггера */
CREATE TRIGGER CodeStOre FOR Store
ACTIVE
BEFORE INSERT
AS
BEGIN
NEW.S_Code = GEN_ID(GenStore, 1);
END
```

После добавления к таблице store новой записи ключевому столбцу s\_Code этой записи автоматически присваивается уникальное значение. Это обеспечивается обращением GEN\_ID к

генератору GenStore, который создается отдельно от триггера (работа с генераторами рассмотрена ниже).

Ограничения ссылочной целостности для связанных таблиц включают в себя:

1. каскадное удаление записей;
2. запрет на редактирование ключевых столбцов.

Рассмотрим, как можно реализовать каскадное удаление записей с участием триггера.

Пусть имеются две таблицы: главная Store и подчиненная Cards, связанные по полям кода s\_code и c\_code2 соответственно:

```
CREATE TABLE Store
(S_Code INTEGER NOT NULL,
...
PRIMARY KEY (S_Code));
CREATE TABLE Cards
(C_Code INTEGER NOT NULL,
C_Code2 INTEGER NOT NULL,
...
PRIMARY KEY (C_Code) );
```

Каскадное удаление записей для связанных таблиц заключается в том, что если из главной таблицы удаляется запись, то и в подчиненной таблице должны быть удалены все соответствующие ей записи.

В нашем случае это выполняется следующим образом:

```
CREATE TRIGGER DeleteStore FOR Store
ACTIVE
AFTER DELETE
AS
BEGIN
DELETE FROM Cards WHERE Store.S_Code = Cards.C_Code2;
END
```

После удаления записи в таблице store (склад) будут автоматически удалены все соответствующие записи в таблице Cards (движение товара).

#### **Замечание**

Для таблиц не должны действовать ограничения ссылочной целостности, заданные на физическом уровне, например, так:

```
CONSTRAINT rStoreCards
FOREIGN KEY (C_Code2) REFERENCES Store
```

В противном случае при попытке удалить запись из главной таблицы генерируется исключение. Если есть такие ограничения, то их можно удалить, например, с помощью программы SQL Explorer.

Аналогичным способом можно реализовать и обновление столбцов связи (ключевых столбцов) связанных таблиц, заключающееся в том, что при изменении значения столбца связи главной таблицы соответственно изменяются значения столбца связи всех связанных записей подчиненной таблицы.

Например:

```
CREATE TRIGGER ChangeStore FOR Store
ACTIVE
BEFORE UPDATE
AS
BEGIN
IF (OLD.S_Code <> NEW.S_Code)
THEN UPDATE Cards
SET C_Code2 = NEW.S_Code
WHERE C_Code2 = OLD.S_Code;
```

END

При изменении столбца S\_code, используемого для связи главной таблицы store с подчиненной таблицей cards, автоматически изменяются значения столбца связи C\_code2 соответствующих записей подчиненной таблицы.

#### **Замечание**

Чтобы столбец связи главной таблицы можно было редактировать, по нему не должен быть создан ключ. Если ключ создан, то его следует удалить, например, с помощью программы SQL Explorer. Более кардинальным решением проблемы является создание таблицы заново с помощью инструкции ALTER TABLE.

Поскольку использование триггеров не допускает существования ограничений на физическом уровне (при определении структуры таблиц), необходимо обеспечить также занесение значения в столбец связи подчиненной таблицы при добавлении в нее новой записи.

#### **Создание генераторов**

Напомним, что, в отличие от базы данных Paradox, для таблиц InterBase отсутствует автоинкрементный тип, обеспечивающий автоматическую установку уникальных значений. Поэтому для обеспечения уникальности значений ключевых столбцов совместно с триггерами используются генераторы. Генератор возвращает уникальное целочисленное значение.

С помощью языка SQL-сервера можно создать генератор и установить для него начальное значение.

Генератор создается следующей инструкцией:

```
CREATE GENERATOR <Имя генератора>;
```

Начальное значение задается инструкцией:

```
SET GENERATOR <Имя генератора> TO <Начальное значение>;
```

Начальное значение представляет собой целое число, начиная с которого формируется числовой ряд.

Обращение к созданному генератору выполняется с помощью функции:

```
GET_ID (<Имя генератора>,<Шаг>);
```

Эта функция возвращает значение, увеличенное на целочисленный шаг относительно предыдущего значения генератора.

Пример:

```
CREATE GENERATOR GenStore;
```

```
SET GENERATOR GenStore TO 1;
```

Здесь создается генератор GenStore, имеющий начальное значение 1. Пример обращения к этому генератору GEN\_ID(GenStore, 1).

#### **IV. Закрепление изученного материала (15 мин.).**

1. Запустите утилиту Interactive SQL.

2. Создайте триггер UPDAT\_REALTY.

3. Попробуйте изменить значение адреса в таблице Realty. Это не удастся. Дело в том, что сервер InterBase поддерживает механизм обеспечения ссылочной целостности "prohibit". В данном случае он реализован на основании декларации связи таблиц FOREIGN KEY (AdR) REFERENCES Realty (AdR) в скрипте tables.sql при создании таблицы Lease.

4. Удалите эту связь (ограничение INTEG\_7):

```
ALTER TABLE Lease
```

```
DROP CONSTRAINT INTEG_7
```

5. Можете убедиться, что она отсутствует: теперь таблица Lease не связана с Realty.

6. Измените значение адреса в таблице Realty.

Как видите, теперь при изменении значения ее первичного ключа триггер автоматически изменяет значение внешнего ключа дочерней таблицы Lease. Другими словами, реализован механизм обеспечения ссылочной целостности "cascade".

## Отчет

Сохраните работу.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №12

**Тема:** Создание запросов и процедур на изменение структуры базы данных (2 часа)

### Цель занятия:

1. Научится создавать запросы и процедуры на изменение структуры базы данных.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** Компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

#### Краткие теоретические сведения

СУБД MS Access имеет следующую структуру:

- таблицы – для постоянного хранения данных;
- запросы – для вызова данных из таблиц по определенному шаблону и их дальнейшей обработки;
- формы – для удобного ввода и просмотра данных;
- отчеты – для удобного вывода заданной информации на печать;
- макросы – для задания свойств открываемых объектов и настройки порядка их работы;
- модули – для упрощения вызова, ранее созданных функций.

Таблицы состоят из полей и записей. Полями называются столбцы, а строки – записями. Внести запись в таблицу означает заполнить данными какую-нибудь строку. Чтобы создать таблицу необходимо определить ее поля, типы данных этих полей и, иногда, некоторые дополнительные свойства этих полей. Не все данные занимают в компьютере одинаковое место. Для их компактного хранения необходимо четко определить: что это текст или число, дата или логический символ. В базах данных под каждый тип данных резервируется некоторое пространство, и если известно, наперед, что оно не будет использовано до конца, его необходимо уменьшить. Как это сделать вы увидите по ходу выполнения данной работы.

#### ЗАДАНИЕ 1. Создание таблиц

1. Запустите программу **MS Access**.
2. Создайте новую базу данных (**Файл** → **Создать** → В области задач **Создание файла**, в группе **Создание** выберите **Новая база данных**
3. В следующем диалоговом окне вы должны выбрать папку для сохранения базы данных и задать имя файла: **Европа.mdb**

На экране появится окно с вкладками, это и есть ваша база данных (она пока пустая).

4. Перейдите на вкладку **Таблицы**.
5. Щелкните мышью по кнопке **Создать**
6. В появившемся диалоговом окне, выберите режим создания **Конструктор** и подтвердите данную операцию.

В столбец **Имя поля** мы будем заносить имена столбцов нашей будущей таблицы (при этом нельзя использовать некоторые символы, в том числе точки и запятые). В столбце **Тип данных** будем выбирать (используя кнопку вызова списка) тип данных. А то, что заносится в столбец **Описание** затем появляется, в виде комментариев, в строке состояния (для проверки в одной из строк этого столбца напишите фразу: моя первая база данных).

Как видно, из ниже перечисленных данных, нам необходимо создать следующие поля:

**Название поля**

**Тип данных**

В области **Свойства поля** на вкладке **Общие** укажите новый **Размер поля**

Код страны

Счетчик

Длинное целое

Страна

Текстовый

15

Столица

Текстовый

10

Площадь

Числовой

Длинное целое

Население

Числовой

Длинное целое

Религия

Текстовый

50

Деньги

Текстовый

10

Строй

Текстовый

15

7. После ввода полей и типов данных желательно задать ключевое поле. Так как, значения в ключевом поле должны быть уникальными, т.е. не повторяющимися, то в этом качестве следует выбрать поле **Код страны**.

Для этого необходимо щелкнуть правой клавишей мыши по заданному полю и, в появившемся меню, выполнить команду **Ключевое поле**.

8. Закройте окно конструктора и, при запросе о сохранении таблицы, задайте имя **Страны Европы**
9. Откройте таблицу для заполнения данными. (Заметьте, что поле с типом данных счетчик будет заполняться автоматически).

**Данные для ввода:**

**Бухарест**

Площадь: **237 500** кв. км

Число жителей: **23 014 000**

Основная религия:

***Христианство (православные)***

Денежная единица: ***Лей***

Гос. строй: ***Республика***

**Великобритания**

Столица: ***Лондон***

Площадь: ***244 110*** кв. км

Число жителей: ***57 006 000***

Основная религия:

***Христианство (протестанты)***

Денежная единица: ***Фунт***

Гос. строй: ***Монархия***

**Албания**

Столица: ***Тирана***

Площадь: ***28 748*** кв. км

Число жителей: ***3 149 000***

Основная религия: ***Атеизм***

Денежная единица: ***Лек***

Гос. строй: ***Республика***

**Лихтенштейн**

Столица: ***Вадуц***

Площадь: ***160*** кв. км

Число жителей: ***27 840***

Основная религия: ***Христианство (католики)***

Денежная единица: ***Франк***

Гос. строй: ***Монархия***

**Дания**

Столица: ***Копенгаген***

Площадь: ***43 092*** кв. км

Число жителей: ***5 130 000***

Основная религия:

***Христианство (протестанты)***

Денежная единица: ***Крона***

Гос. строй: ***Монархия***

**Греция**

Столица: ***Афины***

Площадь: ***131 957*** кв. км

Число жителей: ***10 055 000***

Основная религия:

***Христианство (православные)***

Денежная единица: ***Драхма***

Гос. строй: ***Республика***

**Монако**

Столица: ***Монако***

Площадь: ***2*** кв. км

Число жителей: ***28 000***

Основная религия: ***Христианство (католики)***

Денежная единица: ***Франк***

Гос. строй: ***Княжество***

**Люксембург**

Столица: ***Люксембург***

Площадь: ***2 586*** кв. км

Число жителей: ***372 000***

Основная религия: ***Христианство (католики)***

Денежная единица: **Франк**  
Гос. строй: **Герцогство**

10. Закройте таблицу с сохранением.

11. Сохраните данную базу данных. В отличие от ранее изученных программ для этого достаточно закрыть Access.

### **Краткие теоретические сведения**

В процессе ввода данных очень часто возникает необходимость защитить оператора от ошибки и разграничить доступ к важной информации. Для этого данные разных категорий разделяют по разным таблицам, кроме этого, как правило, такое разделение позволяет более компактно хранить информацию. Данные в главной таблице индексируют и, используя эти индексы, связывают с подчиненными таблицами. Базы данных с такими связями называют реляционными. В данной лабораторной работе мы будем приводить, созданную ранее базу данных, к такому виду.

### **ЗАДАНИЕ 2. Разделение данных на две таблицы**

1. Откройте базу данных **Европа**
2. Создайте таблицу **Религия** с полями:

**Название поля**

**Тип данных**

Код\_религии

Счетчик

Религия

Текстовый

3. Поле **Код\_религии** сделайте ключевым
4. Создайте таблицу **Строй** с полями:

**Название поля**

**Тип данных**

Код\_строя

Счетчик

Строй

Текстовый

5. Поле **Код\_строя** сделайте ключевым
6. Заполните эти таблицы, не допуская повторений (таким образом, напротив каждой религии и каждого строя будет стоять его код)

7. Откройте таблицу **Страны Европы** и замените названия в полях **Строй** и **Религия** на соответствующие им коды в ранее созданных таблицах **Религия** и **Строй** (для автоматизации попробуйте использовать команду замены из меню Правка)

8. Откройте таблицу **Страны Европы** в режиме конструктора

9. Для полей **Строй** и **Религия** измените тип данных на числовой

### **10. ЗАДАНИЕ 3. Установка связи между двумя таблицами**

1. Выполните команду **Схема данных** из меню **Сервис**

2. В диалоговом окне добавления таблиц добавьте в схему все три таблицы

3. На поле **Код\_религии** таблицы **Религия** нажмите левую клавишу мыши и удерживая ее перетащите на поле **Религия** таблицы **Страны Европы**

11. В появившемся диалоговом окне необходимо установить нужную связь: флажок напротив опции **обеспечение целостности данных** означает, что перед тем как занести данные в подчиненную таблицу, программа будет проверять их на соответствие главной. (Таблица **Страны Европы** является подчиненной для таблиц **Религия** и **Строй**). Флажок напротив опции **каскадное обновление связанных полей** означает, что изменения в главной таблице автоматически будут влиять на подчиненную. Флажок напротив опции **каскадное удаление связанных полей** означает, что поля удаленные в главной таблице будут удалены и в подчиненной.

4. Установите все эти флажки

- Аналогичную операцию проделайте с таблицами **Страны Европы** и **Строй** (Если связь между полями **Код\_строя** и **Строй** не устанавливается, проверьте типы данных связываемых полей)
- Закройте схему с сохранением
- Попробуйте в таблице **Страны Европы** в полях **Строй** и **Религия** поменять коды на несуществующие в главных таблицах. Получилось? Почему?
- Сохраните базу данных

### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### Лабораторное занятие №13

**Тема:** Работа с журналом аудита базы данных (2 часа)

#### Цель занятия:

- Изучить основные принципы аудита безопасности Windows и Linux, управление политикой и правилами аудита, журналами безопасности системы.

#### ИНСТРУКЦИОННАЯ КАРТА

##### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

##### Практические задания

##### Теоретические сведения

#### 1. Общие понятия

**Аудит** — это одно из основных средств защиты ОС. Аудит позволяет отслеживать и журналировать события, связанные с безопасностью. Примерами событий для аудита можно назвать доступ к файлу, вход в систему или изменение системной конфигурации.

**Журнал безопасности** представляет собой базу данных или файл, в котором регистрируются события, связанные с безопасностью системы.

Благодаря системе аудита, администратор может узнать, кто, каким образом и когда воспользовался (или пытался воспользоваться, но получил отказ в доступе) интересующими его ресурсами.

Настройка средств аудита позволяет выбрать типы событий, подлежащих регистрации, и определить, какие именно параметры будут регистрироваться.

Наиболее общими типами событий для аудита безопасности являются:

- доступ к файлам и каталогам;
- управление учетными записями пользователей и групп;
- вход пользователей в систему и выход из нее.

Как правило, фиксируются следующие параметры, касающиеся действий, совершаемых пользователями:

- выполненное действие;
- идентификаторы пользователей и групп, выполнивших действие;
- дата и время выполнения.

Аудит приводит к дополнительной нагрузке на систему, поэтому необходимо регистрировать лишь события, действительно представляющие значение с точки зрения безопасности.

## 2. Управление аудитом в Windows

### Политика аудита

Перед выполнением аудита необходимо выбрать **политику аудита**. Политика аудита указывает типы событий, которые будут регистрироваться в журнале безопасности. При первой установке Windows XP SP2 все категории аудита выключены. Включая аудит различных категорий событий, можно создавать политику аудита, удовлетворяющую необходимым требованиям.

Настройка политик аудита доступна в оснастке Групповая политика. Существуют следующие категории аудита:

- *Аудит событий входа в систему* отслеживает события, связанные с входом пользователя в систему и выходом из неё;
- *Аудит управления учетными записями* отслеживает все события, связанные с управлением учетными записями. Записи аудита появляются при создании, изменении или удалении учетных записей пользователя или группы;
- *Аудит доступа к службе каталогов* отслеживает события доступа к каталогу Active Directory. Записи аудита создаются каждый раз при доступе пользователей или компьютеров к каталогу;
- *Аудит входа в систему* отслеживает события входа в систему или выхода из нее, а также удаленные сетевые подключения;
- *Аудит доступа к объектам* отслеживает использование системных ресурсов файлами, каталогами, общими ресурсами, и объектами Active Directory;
- *Аудит изменения политики* отслеживает изменения политик назначения прав пользователей, политик аудита или политик доверительных отношений;
- *Аудит использования привилегий* отслеживает каждую попытку применения пользователем предоставленного ему права или привилегии.

**Примечание.** Политика Аудит использования привилегий не отслеживает события, связанные с доступом к системе, такие, как использование права на интерактивный вход в систему или на доступ к компьютеру из сети. Эти события отслеживаются с помощью политики Аудит входа в систему.

- *Аудит отслеживания процессов* отслеживает системные процессы и ресурсы, используемые ими.

- *Аудит системных событий* отслеживает события включения, перезагрузки или выключения компьютера, а также события, влияющие на системную безопасность или отражаемые в журнале безопасности.

Рассмотрим перечисленные выше политики аудита более подробно.

#### Аудит событий входа учетных записей в систему

Этот параметр политики определяет необходимость аудита каждого входа пользователя в систему или выхода из нее с другого компьютера, проверяющего учетную запись. При проверке подлинности локального пользователя на локальном компьютере формируется событие входа в систему, регистрируемое в локальном журнале безопасности. События выхода из системы не регистрируются.

#### Аудит управления учетными записями

Этот параметр политики определяет необходимость аудита каждого события управления учетными записями на компьютере. Примеры событий управления учетными записями:

- создание, изменение или удаление учетной записи пользователя;
- переименование, отключение или включение учетной записи пользователя;

- задание или изменение пароля

Организациям нужна возможность определения того, кто создает, изменяет и удаляет учетные записи доменов и локальных систем. Несанкционированные изменения могут быть как ошибками администраторов, не соблюдающих политики организации, так и признаками умышленных атак.

Например, события отказов при управлении учетными записями часто свидетельствуют о попытках расширения привилегий, предпринимаемых администратором более низкого уровня или злоумышленником, получившим доступ к его учетной записи. Соответствующие журналы помогают узнать, какие учетные записи изменил или создал хакер.

Наилучший подход к управлению доступом состоит в том, чтобы предоставлять доступ группам, а не отдельным пользователям. С помощью категории событий Управление учетными записями можно легко идентифицировать случаи изменения членства в группах. Во многих случаях злоумышленники, получившие права административного доступа к системе, сначала создают новую учетную запись, а затем используют ее для дальнейших атак. С помощью событий категории Управление учетными записями факт создания новой учетной записи может быть легко выявлен.

#### Аудит доступа к службе каталогов

Категория Аудит доступа к службе каталогов обеспечивает низкоуровневый аудит объектов ActiveDirectory (AD) и их свойств. Поскольку данная категория имеет непосредственное отношение к AD, то активировать аудит подобных событий на системах, которые не являются контроллерами домена, не имеет ни малейшего смысла.

#### Аудит событий входа в систему

Этот параметр политики определяет необходимость аудита каждого входа пользователя в систему или выхода из нее. Параметр Аудит событий входа в систему регламентирует создание записей, служащих для слежения за активностью локальных учетных записей.

Если присвоить параметру Аудит событий входа в систему значение **Нет аудита**, будет сложно или невозможно узнать, какие пользователи входили на компьютеры в организации или пытались это сделать. Если пользователь входит в систему с локальной учетной записью, а параметр Аудит событий входа учетных записей в систему имеет значение **Включен**, при входе в систему формируются два события.

#### Аудит доступа к объектам

Сам по себе этот параметр политики не запускает аудита каких-либо событий. Параметр Аудит доступа к объектам определяет необходимость аудита событий доступа к объекту (например к файлу, папке, разделу реестра или принтеру), для которого задан системный список управления доступом (SACL).

System Access-Control List (SACL) - список, похожий на ACL, но отвечающий не за разрешение или запрет на доступ, а за аудит (протоколирование в журнале безопасности) успешных и безуспешных попыток доступа к объекту. SACL состоит из записей управления доступом. Каждая запись управления доступом включает сведения трех типов:

- участник безопасности (пользователь, компьютер или группа), для которого будет выполняться аудит;
- определенный тип доступа, для которого будет выполняться аудит (маска доступа);
- флаг, указывающий на необходимость аудита событий сбоя доступа, успешного доступа или того и другого вида событий.

Если параметр Аудит доступа к объектам настроен для регистрации значений **Успех**, запись аудита создается каждый раз, когда пользователь успешно получает доступ к объекту с указанным системным списком управления доступом. Если этот параметр настроен для регистрации значений **Отказ**, запись аудита создается каждый раз, когда пользователь безуспешно пытается получить доступ к объекту с указанным системным списком управления доступом.

При настройке системных списков управления доступом организации должны определять только те действия, которые необходимо включить. Например, может потребоваться включить параметр **Аудит записи и добавления данных** для EXE-файлов, чтобы отслеживать их изменение или замену, так как вирусы, черви и «троянские кони» обычно воздействуют на EXE-файлы. Кроме того, может потребоваться отслеживание доступа к конфиденциальным документам и их изменения.

В принципе с помощью категории **Аудит доступа к объектам** можно следить за доступом к файлам, папкам, принтерам, разделам реестра и системным службам, но в большинстве случаев

данная категория используется для отслеживания доступа к файлам и папкам. Как только будет включен аудит по данной категории, в журнале безопасности сразу же отобразится некоторое количество событий, касающихся доступа к объектам в базе безопасности SAM. Однако каких-либо других событий, связанных с доступом к файлам или другим объектам, вы здесь не увидите, поскольку каждый объект имеет свои настройки параметров аудита, а по умолчанию почти у всех объектов аудит отключен. Это правильная практика, поскольку, если в системе будет включен аудит попыток доступа к каждому файлу или объекту, то данная система до своей полной остановки будет заниматься только обработкой этих событий, а ее журнал безопасности быстро переполнится, вне зависимости от назначенного ему объема. Рекомендуется применять эту категорию только к критически важным файлам, действительно требующим механизмов слежения за доступом к ним.

#### Аудит изменения политики

Этот параметр политики определяет необходимость аудита каждого изменения политик назначения прав пользователям, политик доверия или самой политики аудита.

Если параметр Аудит изменения политики настроен для регистрации значений **Успех**, запись аудита создается при каждом успешном изменении политик назначения прав пользователям, политик доверия или политик аудита. Если этот параметр политики настроен для регистрации значений **Отказ**, запись аудита создается при каждой неудачной попытке изменения политик назначения прав пользователям, политик доверия или политик аудита.

Рекомендованный способ настройки этого параметра позволяет узнать, какие привилегии учетной записи злоумышленник пытается повысить или получить (например привилегии **Отладка программ** или **Архивация файлов и каталогов**).

#### Аудит использования привилегий

Для обеспечения контроля возможностей выполнения пользователями функций системного уровня, таких как изменение системного времени или выключение, в Windows применяется система прав пользователей (привилегий).

Этот параметр политики определяет необходимость аудита каждого применения права пользователя. Если параметр Аудит использования привилегий настроен на регистрацию значений типа **Успех**, запись аудита создается при каждом успешном применении права пользователя. Если этот параметр настроен на регистрацию значений типа **Отказ**, запись аудита создается при каждой неудачной попытке применения права пользователя.

При применении указанных ниже прав пользователя аудит не записывается, даже если задан параметр Аудит использования привилегий, поскольку для этих прав регистрируется большое число событий в журнале безопасности. Аудит следующих прав пользователя отрицательно сказывался бы на производительности компьютеров:

- Обход перекрестной проверки
- Отладка программ
- Создание маркерного объекта
- Замена маркера уровня процесса
- Создание аудитов безопасности
- Архивация файлов и каталогов
- Восстановление файлов и каталогов

**Примечание.** Если нужно осуществлять аудит этих прав, необходимо включить в групповой политике параметр **Аудит: аудит прав на архивацию и восстановление**.

#### Аудит отслеживания процессов

Этот параметр политики определяет необходимость аудита подробной информации о таких событиях, как активация программ, завершение процессов, дублирование дескрипторов и косвенный доступ к объектам. Если этот параметр настроен на регистрацию значений типа **Успех**, запись аудита создается при каждой операции, успешно выполненной отслеживаемым процессом. Если этот параметр настроен на регистрацию значений типа **Отказ**, запись аудита создается при каждой неудачной попытке выполнения операции отслеживаемым процессом.

Если параметр Аудит отслеживания процессов задан, регистрируется большое количество событий, поэтому обычно ему присваивают значение **Нет аудита**. Однако этот параметр может оказаться очень полезным при реагировании на инциденты, потому что он позволяет получить подробные сведения о запущенных процессах и времени запуска каждого из них.

### Аудит системных событий

Этот параметр политики определяет необходимость аудита, когда пользователь перезагружает или выключает компьютер, либо когда происходит событие, влияющее на безопасность или журнал безопасности компьютера (например, очистка журнала событий). Если этот параметр настроен на регистрацию значений типа **Успех**, запись аудита создается при успешном выполнении системного события. Если этот параметр настроен на регистрацию значений типа **Отказ**, запись аудита создается при неудачной попытке выполнения системного события.

#### **События аудита**

Как уже упоминалось ранее, Windows XP регистрирует в журналах происходящие события, которые отслеживаются политикой аудита. Пользуясь информацией журналов событий, можно получить сведения о неполадках аппаратного и программного обеспечения, а также наблюдать за событиями безопасности Windows. Управлять и просматривать содержимое журналов событий можно с помощью утилиты **Просмотр событий** (Event Viewer).

Windows XP записывает события в три журнала:

- *Системный журнал* (system log) содержит сообщения об ошибках, предупреждения и другую информацию, исходящую от операционной системы и компонентов сторонних производителей. Список событий, регистрируемых в этом журнале, предопределен операционной системой и компонентами сторонних производителей и не может быть изменен пользователем. Журнал находится в файле Sysevent.evt.
- *Журнал безопасности* (Security Log) содержит информацию об успешных и неудачных попытках выполнения действий, регистрируемых средствами аудита. События, регистрируемые в этом журнале, определяются заданной вами стратегией аудита. Журнал находится в файле Secevent.evt.
- *Журнал приложений* (Application Log) содержит сообщения об ошибках, предупреждения и другую информацию, выдаваемую различными приложениями. Список событий, регистрируемых в этом журнале, определяется разработчиками приложений. Журнал находится в файле Appevent.evt.

Все журналы размещены в папке %Systemroot%\System32\Config. Журналы событий можно просматривать с любого компьютера локальной сети, при наличии прав администратора на компьютере, где расположен журнал. Нас будет интересовать прежде всего *Журнал безопасности*, т.к. именно в нем регистрируются события аудита безопасности.

При выборе событий для проведения аудита следует учитывать возможность переполнения журнала. Чтобы иметь возможность просматривать содержимое журналов за длительный промежуток времени, рекомендуется периодически архивировать текущие журналы. Дополнительные журналы могут быть добавлены при установке дополнительных служб.

В таблице 1 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита событий управления учетными записями.

Таблица 1. События управления учетными записями

| <b>Код события</b> | <b>Описание события</b>                   |
|--------------------|-------------------------------------------|
| 624                | Создана учетная запись пользователя.      |
| 627                | Изменен пароль пользователя.              |
| 628                | Задан пароль пользователя.                |
| 630                | Удалена учетная запись пользователя.      |
| 635                | Создана локальная группа.                 |
| 636                | В локальную группу добавлен член.         |
| 637                | Из локальной группы удален член.          |
| 638                | Удалена локальная группа.                 |
| 639                | Изменена учетная запись локальной группы. |
| 642                | Изменена учетная запись пользователя.     |

| <b>Код<br/>события</b> | <b>Описание события</b>      |
|------------------------|------------------------------|
| 685                    | Изменено имя учетной записи. |

В таблице 2 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита событий входа в систему.

Таблица 2. События аудита входа в систему

| <b>Код<br/>события</b> | <b>Описание события</b>                                                                                                                                                  |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 528                    | Пользователь успешно вошел в систему.                                                                                                                                    |
| 529                    | Сбой при входе в систему. Предпринята попытка входа в систему с использованием неизвестного имени пользователя или известного имени пользователя с неправильным паролем. |
| 530                    | Сбой при входе в систему. Предпринята попытка входа в систему вне допустимого интервала времени.                                                                         |
| 531                    | Сбой при входе в систему. Предпринята попытка входа в систему с отключенной учетной записью.                                                                             |
| 532                    | Сбой при входе в систему. Предпринята попытка входа в систему с устаревшей учетной записью.                                                                              |
| 533                    | Сбой при входе в систему. В систему попытался войти пользователь, не имеющий на это права.                                                                               |
| 534                    | Сбой при входе в систему. Пользователь попытался войти в систему с использованием пароля недопустимого типа.                                                             |
| 535                    | Сбой при входе в систему. Пароль указанной учетной записи устарел.                                                                                                       |
| 537                    | Сбой при входе в систему. Попытка входа в систему завершилась неудачей по иным причинам.                                                                                 |
| 538                    | Процесс выхода пользователя из системы завершен.                                                                                                                         |
| 539                    | Сбой при входе в систему. Ко времени попытки входа в систему учетная запись была заблокирована.                                                                          |
| 543                    | Работа в основном режиме завершена.                                                                                                                                      |
| 551                    | Пользователь инициировал процесс выхода из системы.                                                                                                                      |
| 552                    | Пользователь успешно вошел в систему с явно заданными учетными данными, уже будучи зарегистрированным в системе в качестве другого пользователя.                         |

В таблице 3 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита событий доступа к объектам.

Таблица 3. События доступа к объектам

| <b>Код<br/>события</b> | <b>Описание события</b>                                                                                                                                                                       |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 560                    | Предоставлен доступ к существующему объекту.                                                                                                                                                  |
| 562                    | Закрыт дескриптор объекта.                                                                                                                                                                    |
| 563                    | Выполнена попытка открыть объект с целью его удаления.                                                                                                                                        |
| 564                    | Удален защищенный объект.                                                                                                                                                                     |
| 565                    | Предоставлен доступ к существующему типу объекта.                                                                                                                                             |
| 567                    | Использовано разрешение, связанное с дескриптором. Дескриптор создается с определенными разрешениями (например разрешениями на чтение и запись). При использовании дескриптора для каждого из |

| Код события | Описание события                                                                                                         |
|-------------|--------------------------------------------------------------------------------------------------------------------------|
|             | использованных разрешений может быть создана запись аудита.                                                              |
| 570         | Клиент попытался получить доступ к объекту. Это событие формируется при каждой попытке выполнения операции над объектом. |

В таблице 4 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита событий изменения политики безопасности.

Таблица 4. События аудита изменения политики

| Код события | Описание события                                       |
|-------------|--------------------------------------------------------|
| 608         | Предоставлено право пользователя.                      |
| 609         | Удалено право пользователя.                            |
| 612         | Изменена политика аудита.                              |
| 618         | Изменена политика восстановления зашифрованных данных. |
| 621         | Учетной записи предоставлен доступ к системе.          |
| 622         | Для учетной записи заблокирован доступ к системе.      |
| 623         | Политика аудита задана для каждого пользователя.       |

В таблице 5 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита событий отслеживания процессов.

Таблица 5. События отслеживания процессов

| Код события | Описание события                          |
|-------------|-------------------------------------------|
| 592         | Создан процесс.                           |
| 593         | Процесс завершил работу.                  |
| 601         | Пользователь попытался установить службу. |
| 602         | Создано задание планировщика.             |

В таблице 6 приведены основные события безопасности, которые регистрируются в журнале безопасности, если задана политика аудита системных событий.

Таблица 6. Сообщения о системных событиях для аудита системных событий

| Код события | Описание события                                                                                                                                         |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 512         | Запуск системы Windows.                                                                                                                                  |
| 513         | Завершение работы системы Windows.                                                                                                                       |
| 516         | Внутренние ресурсы, выделенные очереди сообщений о событиях безопасности, исчерпались, что привело к утрате некоторых сообщений о событиях безопасности. |
| 517         | Очищен журнал аудита.                                                                                                                                    |
| 520         | Изменено системное время.                                                                                                                                |

### Управление аудитом

Применение политик аудита существенно повышает безопасность и целостность системы. Практически каждая компьютерная система в сети должна быть настроена с ведением журналов безопасности.

Администраторам следует создать политику аудита, определяющую содержимое отчетов о событиях безопасности и регистрирующую действия пользователей или компьютеров, относящиеся к указанным категориям событий.

Перед реализацией политики аудита нужно решить, для каких категорий событий следует выполнять аудит. От параметров аудита, заданных администратором для категорий событий, зависит

политика аудита организации. Если параметры аудита для конкретных категорий событий определены, администраторы могут создать политику аудита, соответствующую требованиям организации.

Если политика аудита не задана, определить, что произошло при инциденте в сфере безопасности, будет сложно или невозможно. Если же настроить параметры аудита так, чтобы события регистрировались при многих санкционированных действиях, журнал безопасности может быть заполнен бесполезными данными.

В сетях с минимальным требованиям к безопасности подвергайте аудиту:

- успешное использование ресурсов, только в том случае, если эта информация вам необходима для планирования;
- успешное использование важной и конфиденциальной информации.

В сетях со средними требованиями к безопасности подвергайте аудиту:

- успешное использование важных ресурсов;
- удачные и неудачные попытки изменения стратегии безопасности и административной политики;
- успешное использование важной и конфиденциальной информации.

В сетях с высокими требованиями к безопасности подвергайте аудиту:

- удачные и неудачные попытки регистрации пользователей;
- удачное и неудачное использование любых ресурсов;
- удачные и неудачные попытки изменения стратегии безопасности и административной политики.

Журналы сбоев часто оказываются более информативными, чем журналы успешного выполнения операций, потому что сбои обычно свидетельствуют об ошибках. Например, успешный вход пользователя в систему обычно считается нормальным развитием событий. Если кто-то безуспешно пытается несколько раз войти в систему, это может быть признаком использования чужих учетных данных. События, происходящие на компьютере, регистрируются в журналах событий.

Перед реализацией политики аудита в организации следует определить способы сбора, организации и анализа данных. От большого объема данных аудита мало пользы, если отсутствует план их использования. Кроме того, аудит компьютерных сетей может отрицательно сказываться на производительности систем. Даже если влияние определенного сочетания параметров на компьютер конечного пользователя практически незаметно, на сервере с высокой нагрузкой оно может оказаться существенным. Таким образом, перед заданием новых параметров аудита в рабочей среде следует проверить, не ухудшит ли это производительность систем.

### 3. Управление аудитом в Linux

В ОС Linux, начиная с версии ядра 2.6, существует возможность журналировать такие события безопасности, как доступ к файлам/каталогам и выполнение системных вызовов. Регистрацией таких событий занимается служба (демон) **auditd**.

Процесс организации аудита событий безопасности ОС Linux состоит из следующих действий:

1. Установка и настройка **auditd**.
2. Настройка правил аудита.
3. Запуск **auditd**.
4. Анализ полученных данных и создание отчетов аудита.

**Примечание.** Для работы **auditd**, необходимо чтобы ядро было собрано с опциями **AUDIT** и **AUDITSYSCALL**. Это можно проверить с помощью команды **grep AUDIT /boot/config-`uname -r`**.

Для того чтобы использовать аудит, в системе должен быть установлен пакет **auditd**. Демон **auditd** позволяет системному администратору настраивать процесс аудита, а именно:

- Задавать отдельный файл для журналирования событий.
- Определять ротацию журнального файла событий.
- Задавать оповещения в случае переполнения журнала событий.
- Определять уровень детализации событий.
- Настраивать правила аудита.

В процессе своей работы демон **auditd** использует конфигурационный файл **/etc/audit/auditd.conf**. Ниже приведен пример содержимого данного конфигурационного файла.

```
log_file=/var/log/audit/audit.log
```

```
log_format=RAW
```

```
priority_boost=3
```

```
flush=incremental
```

```
freq=20
```

```
num_logs=4
```

```
dispatcher=/sbin/audispd
```

```
disp_qos=lossy
```

```
max_log_file=5
```

```
max_log_file_action=rotate
```

```
space_left=75
```

```
space_left_action=syslog
```

```
action_mail_acct=root
```

```
admin_space_left=50
```

```
admin_space_left_action=suspend
```

```
disk_full_action=suspend
```

```
disk_error_action=suspend
```

В директиве **log\_file** указывается файл, куда будут журналироваться события. По умолчанию, таким файлом является файл **/var/log/audit/audit.log**. В директиве **log\_format** указывается формат данных, которые необходимо журналировать. В случае указания значения **RAW** в данной директиве, события записываются в файл в том виде, в котором они поступают из ядра. В директиве **flush** определяется частота журналирования событий. Если для данной директивы указано значение **INCREMENTAL**, то очистка журнала определяется на основе директивы **freq**, в которой указывается количество записей, которые принимает демон **auditd**, прежде чем записать их в журнал.

Если в файле **auditd.conf** указана директива **max\_log\_file\_action** со значением **ROTATE**, то журнальные файлы будут ротироваться, достигнув размера, заданного в директиве **max\_log\_file** (в Мб). В директиве **num\_logs** определяется количество журнальных файлов, которые должны быть сохранены, прежде чем будет выполнена ротация данных. В директиве **dispatcher** задается программа, которая обрабатывает все события аудита. Она может использоваться для формирования отчетов или конвертации журнала в формат, понимаемый другими программами анализа журнальных файлов. Директива **disp\_qos** определяет параметры взаимодействия данной программы с демоном **auditd**. Если в данной директиве указано значение **lossy**, то события, посланные данной программой, удаляются, если буфер обмена данными между демоном **auditd** и данной программой полностью заполнен (по умолчанию размер данного буфера составляет 128Кб). Запись событий по-прежнему осуществляется, если в директиве **log\_format** не указано значение **nolog**. В директивах **space\_left** и **space\_left\_action** задается, соответственно, предел свободного дискового пространства раздела, на котором располагаются данные аудита, и действие, которое нужно выполнить в случае превышения указанного предела. Если указано значение **SYSLOG**, то в журнал сервиса **syslog** будет записано соответствующее предупреждение. В директивах **disk\_full\_action** и **disk\_error\_action** указываются, соответственно, действие выполняемое в случае заполнения дискового пространства на разделе, содержащем журнальные файлы аудита, и действие выполняемое в случае обнаружения ошибки записи данных в журнальный файл или его ротации. По умолчанию, для данных директив задано значение **SUSPEND**, при котором следующие события записываться в журнальный файл аудита не будут, а действия, выполняемые в настоящий момент над журнальным файлом, завершатся.

Для настройки правил аудита системных вызовов и слежения за файлами и каталогами используется утилита **auditctl**. Если сервис **auditd** настроен на автозапуск, то правила аудита системных вызовов и слежения за объектами файловой системы можно задать в файле **les**. Каждое правило аудита должно задаваться в отдельной строке. Форма записей правил и средств имеет вид аргументов командной строки для команды **auditctl**. Демон **auditd** считывает правила сверху вниз и

если будет обнаружено два конфликтующих правила, то предпочтение будет отдано первому найденному правилу.

В качестве параметра **auditctl** указывается список событий, в который добавляется данное правило. В настоящий момент поддерживаются следующие списки:

- **task** - используется для ведения событий, связанных с созданием процессов;
- **user** - используется для ведения событий, в которых указываются параметры пользовательского пространства, такие как **uid**, **pid** и **gid**;
- **exclude** - используется для запрета аудита указанных в данном списке событий;
- **entry** - используется для регистрации событий системных вызовов;
- **exit** - используется для регистрации событий системных вызовов.

В качестве параметра также указывается одно из действий, предпринимаемых по отношению к указанным в списках событиям. Доступно всего два действия: **never** и **always**. При указании действия **never**, события не записываются в журнал аудита. Указание действия **always** имеет противоположный эффект.

В аргументах также задаются опции, являющиеся фильтрами, при помощи которых можно детализировать события аудита. Полный список опций можно посмотреть в руководстве man для команды **auditdctl**. В качестве опций можно задавать уникальные идентификаторы пользователя или процесса, название системного вызова и многое другое. При этом, возможно использовать логические выражения для формирования комбинированных опций.

Добавление правил аудита происходит при помощи команды **auditdctl** с ключом **-a**. При этом правила, добавляются в конец файла **auditd.conf**. Для того, чтобы добавить правило в начало данного файла, необходимо использовать ключ **-A**.

Рассмотрим примеры добавления правил аудита:

```
auditdctl -a exit,always -s open -f uid=502 -f key=502open
```

```
auditdctl -a entry,always -s chown
```

В первом случае задается правило, которое фиксирует все события, связанные с системным вызовом **open**, выполняемым от пользователя с идентификатором 502. Записи, удовлетворяющие данному условию, помечаются в файле **audit.log** меткой **502open**. Во втором случае задается правило, которое фиксирует все системные вызовы **chown**. Когда заданное в правиле действие будет выполнено, результат его выполнения заносится в журнал **/var/log/audit/audit.log**.

Помимо слежения за системными вызовами, система аудита ОС Linux позволяет следить за доступом к файлам и каталогам. Для этого в файле **les** необходимо создать правило имеющее следующий синтаксис:

**-w -k**

В параметре **-w** указывается полный путь к файлу или каталогу, за которым необходимо следить. Параметр **-k** используется для пометки всех событий, связанных с доступом указанному файлу. Это позволяет быстро находить интересующую информацию в журнальном файле. В следующем примере приведены два правила, в которых осуществляется слежение за файлом **/etc/passwd**, а также слежение за доступом к команде **iptables**:

```
auditdctl -w /etc/passwd -k passwd
```

```
auditdctl -w /sbin/iptables -k iptables -p x
```

В ключах **-k** указываются соответствующие метки событий. Ключ **-p x** во втором правиле указанного примера обозначает, что необходимо рассматривать только события, связанные с выполнением данного файла.

Для просмотра активных правил аудита необходимо выполнить команду **auditctl -l**. Для определения статуса системы аудита необходимо выполнить команду **auditctl -s**.

Обычно журнальный файл системы аудита событий ОС Linux можно просматривать при помощи любого текстового редактора. Однако для выполнения общего анализа событий требуется суммировать отдельные записи в данном файле, что практически нереально выполнить, используя только возможности текстового редактора. Для выполнения анализа и отображения статистики аудита событий в состав пакета **audit** входят утилиты **ausearch** и **aureport**. Утилита **ausearch** может выполнять поиск записей аудита на основе различных критериев поиска, таких как имя файла,

идентификатор пользователя, название системного вызова и прочее. Утилита **aureport** используется для формирования отчетов на основе журнального файла аудита *audit.log*.

### Порядок выполнения работы

1. Изучить теоретический материал лабораторной работы.
2. Изучить управление политиками аудита безопасности в системе Windows.

Управление политиками аудита можно осуществлять через оснастку «Локальные параметры безопасности».

#### - оснастка «Локальные параметры безопасности»

1. Для вызова оснастки «Локальные параметры безопасности» нажмите «Пуск» - «Выполнить», наберите *secpol.msc* и нажмите «ОК».
2. Далее выберите «Локальные политики» - «Политика аудита»
3. Изучить аудит событий Windows, объектов файловой системы и реестра.

После выбора политики аудита изучение процесса аудита следует осуществлять экспериментальным путем в два этапа:

- выполнение действия, подлежащего аудиту (например, добавление новой учетной записи при включенной политике «Аудита управления учетными записями»);

- просмотр зарегистрированного события в журнале безопасности Windows

При включении соответствующей политики аудита, регистрация событий аудита начинается автоматически. Это справедливо для всех политик аудита, кроме **«Политики аудита доступа к объектам»**. Данная политика используется для регистрации событий доступа к основным объектам ОС (файловой системе, реестру, принтерам и т.д.). Для того, чтобы регистрация событий началась, необходимо явно указать объект и виды доступа, подлежащие аудиту. Для файловой системы и реестра это делается через закладку «Безопасность» диалогового окна свойств объекта (см. л/р №2). На закладке «Безопасность» следует нажать кнопку «Дополнительно» и далее перейти а закладку «Аудит».

4. Изучить управление журналами событий и безопасности Windows

Для просмотра и управления журналом безопасности Windows используют оснастку «Просмотр событий».

Для вызова оснастки необходимо:

1. Нажать Пуск > Выполнить
2. Набрать *eventvwr.msc* и нажать ОК
2. Изучить управление аудитом безопасности в ОС Linux.

#### - через командную строку

Для управления аудитом безопасности необходимо:

1. При необходимости установить в систему пакет **auditd** в режиме автозапуска
2. В графическом режиме (например, в оболочке Gnome) выбрать **Приложения -> Стандартные -> Rootterminal**
3. Ввести пароль учетной записи root
4. Использовать команды **auditctl**, **aureport**

### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала          | Творческая составляющая              | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|----------------------------------------|--------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов | Защита работы аудиторно – 20 баллов  | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения                | Защита своей работы персонально – 10 |                                                                                                     |

|                                             |                                                  |                                                           |  |
|---------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|--|
|                                             | работы – 10 баллов                               | баллов                                                    |  |
| Не своевременно выполнена работа – 0 баллов | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |  |

## Лабораторное занятие №14

**Тема:** Мониторинг нагрузки сервера (3 часа)

### Цель занятия:

1. Изучить принципы работы простейших средств мониторинга сети. Получить навыки решения задач, связанных с мониторингом сети.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

#### Теоретические основы

##### 1. Протокол ICMP

Протокол ICMP (Интернет-протокол контрольных сообщений) стека протоколов TCP/IP предназначен для передачи между сетевыми устройствами сообщений об ошибках и контрольных сообщений при помощи IP-пакетов.

В протоколе ICMP определены несколько типов сообщений, в том числе:

|                         |                       |                   |
|-------------------------|-----------------------|-------------------|
| Destination Unreachable | Time to Live Exceeded | Parameter Problem |
| Source Quench           | Redirect              | Echo              |
| Echo Reply              | Timestamp             | Timestamp Reply   |
| Information Request     | Information Reply     | Address Request   |
| Address Reply           |                       |                   |

Например, если маршрутизатор получает пакет, который он не может доставить по указанному в нем адресу, отправителю передается ICMP-сообщение о недостижимости адреса (Destination Unreachable).

##### 2. PING: Проверка соединения с определенным интерфейсом.

Программа ping использует протокол ICMP.

Эта команда посылает пакет эхо-запроса на другой IP-адрес и ожидает ответа. Она чаще всего используется для того, чтобы посмотреть, «жив ли» другой компьютер. Ответ на запрос содержит также данные о том, как долго пакет путешествовал до адресата. Можно использовать команду ping с различными опциями: число посланных пакетов (от 1 до 10), время жизни пакета (time to live –TTL, от 1 до 255ms), размер пакета (от 16 до 8192 байт), время ожидания (timeout, до 9999 ms) и разрешать или нет фрагментацию каждого пакета.

Формат команды в ОС Windows:

```
ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
[-r count] [-s count] [[-j host-list] | [-k host-list]]
[-w timeout] destination-list
```

Options:

-t Выполнение команды до прерывания (Ctrl+C)

-a Разрешать адреса в имена

- n count Число отправляемых пакетов.
- l size Размер буфера отправки
- f Установить флаг "Не фрагментировать".
- i TTL Установить время жизни.
- w timeout Время ожидания ответа в мс.
- v TOS Задание типа службы (поле "Type Of Service").
- r count Запись маршрута для указанного числа переходов.
- s count Штамп времени для указанного числа переходов.
- j host-list Свободный выбор маршрута по списку узлов.
- k host-list Жесткий выбор маршрута по списку узлов.
- destination-list Список рассылки.

3. Программа `tracert`. Определение промежуточных сетевых интерфейсов между хостами.

#### Трассировка маршрута

Программа трассировки маршрута использует протокол ICMP.

Эта утилита очень похожа на `Ping`, за исключением того, что она показывает все другие IP-адреса (интерфейсы), которые пакет проходит до своего места назначения. Дополнительно можно изменять различные опции, ассоциированные с `Trace Route`: максимальное число дозволяемых промежуточных узлов (`maximum hops`, от 1 до 255) и `timeout` (до 9999 ms).

Формат команды в ОС Windows:

```
tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] target_name
```

Options:

- d Не разрешать адреса в имена.
- h maximum\_hops Наибольшее число промежуточных узлов.
- j host-list Трассировка через определенный список хостов
- w timeout Время ожидания каждого ответа в мс.

4. Программа `netstat`. Сетевая статистика.

Программа `netstat` используется для просмотра активных соединений каждого протокола, таблиц маршрутизации, а так же детализирует статистику передачи данных.

Формат команды в ОС Windows:

```
netstat [-a] [-e] [-n] [-s] [-p имя] [-r] [интервал]
```

-a Отображение всех подключений и ожидающих портов.

(Подключения со стороны сервера обычно не отображаются).

-e Отображение статистики Ethernet. Этот ключ может применяться вместе с ключом -s.

-n Отображение адресов и номеров портов в числовом формате.

-p имя Отображение подключений для протокола "имя": `tcp` или `udp`. Используется вместе с ключом -s для отображения статистики по протоколам. Допустимые значения "имя": `tcp`, `udp` или `ip`.

-r Отображение содержимого таблицы маршрутов.

-s Отображение статистики по протоколам. По умолчанию выводятся данные для `TCP`, `UDP` и `IP`. Ключ -p позволяет указать подмножество выводимых данных.

**интервал** Повторный вывод статистических данных через указанный интервал в секундах.

Для прекращения вывода данных нажмите клавиши `CTRL+C`. Если параметр не задан, сведения о текущей конфигурации выводятся один раз.

#### Средства/Подготовка

В работе будут использоваться сети, соединенные с рабочей станцией линками разных типов. Первая сеть (194.85.33.0) доступна через спутниковый канал связи, проходящий через спутниковый ретранслятор на геостационарной орбите с пропускной способностью 512 кбит/с. Вторая сеть 217.23.64.0 доступна через наземный волоконно-оптический канал связи с пропускной способностью 2 Мбит/с, третья сеть 212.194.38.0 входит в состав корпоративной сети ВУЗа и доступна в локальной сети с пропускной способностью 10/100 мбит/с.

В качестве альтернативного инструмента анализа сети используется интерфейс `Looking Glass` на сайте <http://noc.runnet.ru> (IP-адрес 194.85.35.100).

#### Ход работы Задание 1. Ping

Выполните команду `ping` в командной строке с различными значениями параметров -t, -n, -l, -i, -w. Какие наблюдения и выводы вы сделали?

ping www.seun.ru  
ping www.sgu.ru  
ping [www.microsoft.com](http://www.microsoft.com)  
ping [www.sun.com](http://www.sun.com)  
ping 212.193.38.83

---

Выполните ping к тем же хостам с параметром -f, увеличивая параметр -l size. При каком значении размера перестают получаться ответы?

---

## **Задание 2. Tracert**

Выполните команду tracert в командной строке с различными значениями параметров. Какие наблюдения и выводы вы сделали?

Используйте, например,  
tracert www.seun.ru  
tracert www.sgu.ru  
tracert [www.microsoft.com](http://www.microsoft.com)  
tracert [www.sun.com](http://www.sun.com)  
tracert 212.193.38.83

## **Задание 3. Поисковые сервисы Европейского и Российского ip-регистров**

Определите, кому принадлежат сети 194.85.33.0, 217.23.64.0, 212.193.38.0. Для этого используйте поисковые аппараты <http://www.ripe.net/db/whois/whois.html> и <http://www.ripn.net:8080/nic/whois/index.html>.

Пользуясь данными этих информационных систем, попробуйте определить географическое расположение сетей. Попробуйте изобразить топологическую схему соединения этих сетей.

## **Задание 4. Использование программы ping для исследования параметров сети.**

1. Приведите сравнительные результаты выполнения команд ping по адресам 194.85.33.29, 194.85.33.30, 217.23.64.2, 212.193.38.248, 212.193.35.10 по параметрам «время отклика», TTL в форме таблицы. Объясните полученные различия.

2. Соберите средние времена прохождения 10 пакетов на указанные адреса. Сравните с результатами, полученными при использовании сервиса ping в интерфейсе Looking Glass на сайте <http://noc.runnet.ru>. Объясните полученные различия.

3. Соберите усредненные времена прохождения 10 пакетов увеличивающегося размера по указанным адресам. Начните с 64 байт и каждый раз удваивайте размер пакета. При каком размере пакета потери превышают 50 %. Как влияет время ожидания отклика на процент прохождения пакетов этого размера. При каком времени ожидания отклика потери для пакетов зафиксированного размера не возникают?

Представьте результаты измерений в форме таблиц, наилучшим образом проявляющим, по вашему мнению, обнаруженные зависимости.

4. Используя программу ping, оцените вклад разных сетевых участков, по которым проходит эхо-пакет между вашей рабочей станцией и интерфейсом 194.85.33.29.

## **Задание 5. Использование программы tracert для анализа соединений в сети.**

1. Приведите сравнительные результаты выполнения команд tracert по адресам 194.85.33.29, 194.85.33.30, 217.23.64.2, 212.193.38.248, 212.193.35.10. Объясните полученные различия.

2. Выполните трассировку к адресу 212.193.38.248 и к адресу 217.23.64.2 со стороны сайта <http://noc.runnet.ru>. Приведите полученные результаты.

3. Используя данные, полученные в результате выполнения трассировки и отправки эхо-пакетов между интерфейсами 212.193.38.248 и 194.85.35.100, оцените вклад разных участков сетей, соединяющих эти интерфейсы, в среднее время прохождения пакетов между ними.

4. Используя полученную в ходе выполнения всех заданий информацию, уточните схему задания 1, изобразите на ней обнаруженные вами промежуточные интерфейсы и линки сети, объединяющей подсети 194.85.33.0, 217.23.64.0, 212.193.38.0.

**Отчет**

Сохраните работу.

**Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |
|                                                 |                                                  |                                                           |                                                                                                     |

## Раздел 2. Обеспечение качества и сертификация информационных систем

### МДК. 07.02 Сертификация информационных систем

#### Тема 2.1. Защита и сохранность информации баз данных

##### Лабораторное занятие №1

Тема: Настройка политики безопасности (1 час)

##### Цель занятия:

1. ознакомиться с методами ограничения доступа к информации

##### ИНСТРУКЦИОННАЯ КАРТА

##### Материально - техническое оснащение

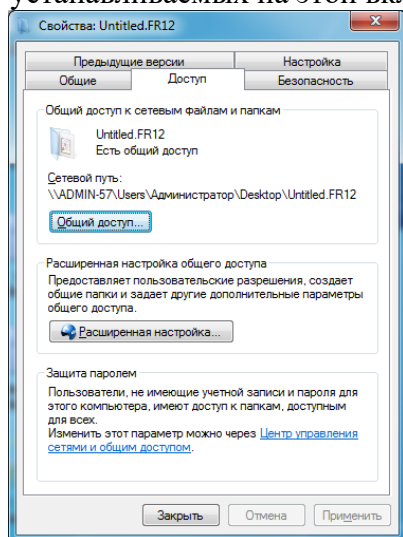
**Оборудование:** Компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

##### Практические задания

1. Освоить средства разграничения доступа пользователей к папкам:

- выполнить команду «Общий доступ и безопасность» контекстного меню папки (если эта команда недоступна, то выключить режим «Использовать простой общий доступ к файлам» на вкладке «Вид» окна свойств папки) или команду «Свойства»;
- открыть вкладку «Безопасность» и включить в отчет сведения о субъектах, которым разрешен доступ к папке и о разрешенных для них видах доступа;
- с помощью кнопки «Дополнительно» открыть окно дополнительных параметров безопасности папки (вкладка «Разрешения»);
- включить в отчет сведения о полном наборе прав доступа к папке для каждого из имеющихся в списке субъектов;
- открыть вкладку «Владелец», включить в отчет сведения о владельце папки и о возможности его изменения обычным пользователем;
- открыть папку «Аудит», включить в отчет сведения о назначении параметров аудита, устанавливаемых на этой вкладке, и о возможности их установки обычным пользователем;



- закрыть окно дополнительных параметров безопасности.
2. Освоить средства разграничения доступа пользователей к файлам:
  - выполнить команду «Свойства» контекстного меню файла;
  - повторить все задания п. 1, но применительно не к папке, а к файлу.
  3. Освоить средства разграничения доступа к принтерам:
  - выполнить команду «Принтеры и факсы» меню «Пуск»;
  - выполнить команду «Свойства» контекстного меню установленного в системе принтера;
  - повторить все задания п. 1, но применительно не к папке, а к принтеру.
  4. Освоить средства разграничения доступа к разделам реестра операционной системы:

- с помощью команды «Выполнить» меню «Пуск» запустить программу редактирования системного реестра regedit (regedt32);
- с помощью команды «Разрешения» меню «Правка» редактора реестра определить сведения о правах доступа пользователей к корневым разделам реестра, их владельцах и параметрах политики аудита;
- включить в отчет сведения о правах доступа пользователей к данной папке и о ее владельце. Отчет должен содержать:

### **Контрольные вопросы:**

1. В чем достоинства дискреционной политики безопасности?
2. В чем недостатки мандатной политики безопасности?
3. Кто определяет права доступа к папкам, файлам, принтерам при использовании дискреционной политики безопасности?
4. В чем отличие определения прав на доступ к файлам по сравнению с определением прав на доступ к папкам?
5. В чем отличие определения прав на доступ к принтерам по сравнению с определением прав на доступ к папкам и файлам?
6. В чем отличие определения прав на доступ к разделам реестра по сравнению с определением прав на доступ к папкам и файлам?

### **Отчет**

1. Тема и цель работы
2. Экранные копии выполнения работы
3. Ответы на контрольные вопросы.

### **Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## **Лабораторное занятие №2**

**Тема:** Создание резервных копий базы данных (1 час)

### **Цель занятия:**

- 1.знакомиться с основными конструкциями SQL, технологиями среды MS SQL Server Management, объектами SMO (среды MS Visual Studio) для резервного копирования и восстановления БД.

## **ИНСТРУКЦИОННАЯ КАРТА**

### **Материально - техническое оснащение**

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

### **Практические задания**

**Задание №1.** необходимо создать резервные копии базы данных «МММ» с использованием полного резервного копирования, разностного резервного копирования и резервного копирования журнала транзакций.

Ход работы:

1. Запустите SQL Server Management Studio (SSMS), подключитесь к своему экземпляру SQL Server, используя технологию 1.
2. Создайте папку с именем c:\Student\ВашаПапка\test.
3. Откройте окно нового запроса. Измените контекст на базу данных master, используя технологию 6. Наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных:

**BACKUP DATABASE MMM TO DISK = 'C:\.....TEST\AW.BAK'**

*Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.*

4. Внесите изменение в таблицу «Модель» базы данных МММ. Добавьте одну запись (придумайте сами)/
5. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать резервную копию журнала транзакций и сохранить только что внесенное изменение:

**BACKUP LOG MMM TO DISK = 'C:\.....TEST\AW1.TRN'**

*Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.*

6. Внесите еще одно изменение в таблицу «Модель».
7. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать разностную резервную копию базы данных:

**BACKUP DATABASE MMM TO DISK = 'C:\.....\TEST\AWDIFF1.BAK' WITH DIFFERENTIAL**

*Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.*

8. Внесите еще одно изменение в таблицу «Модель».
9. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных в указанном месте на диске:

**BACKUP LOG MMM TO DISK = 'C:\....TEST\AW2.TRN'**

*Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.*

**Задание №2.** необходимо провести восстановление базы данных «МММ» из сделанных в задании №1 резервных копий.

Ход работы:

1. Если необходимо, запустите SSMS, подключитесь к своему экземпляру SQL Server, используя технологию 1.
2. Выполните восстановление БД из первой полной резервной копии (C:\...TEST\AW.BAK) средствами оболочки SSMS. Для этого выполните:
  - В обозревателе объектов вызовите контекстное меню на вашей БД и выберите задачу восстановления базы данных (см. рисунок 6).

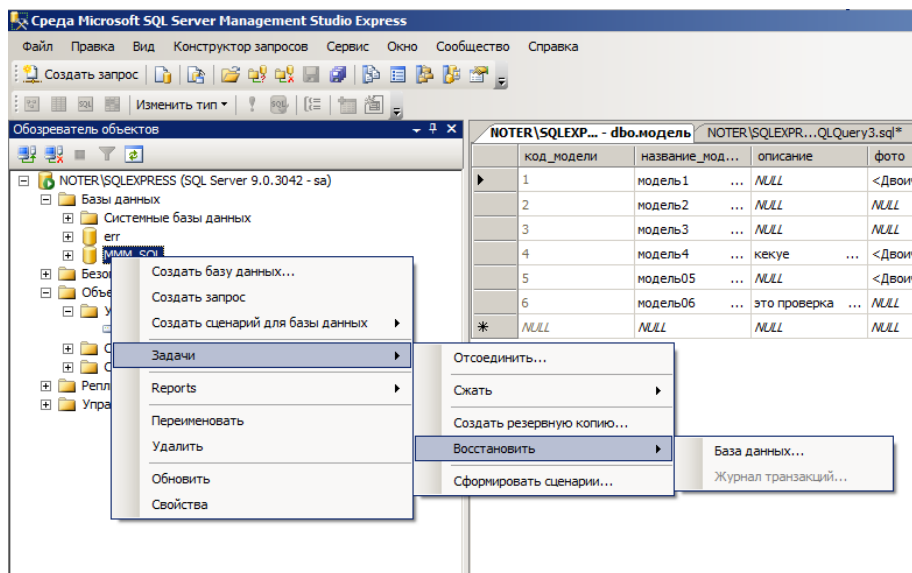


Рисунок 6 – Восстановление БД

- В открывшемся окне необходимо задать следующие параметры восстановления  
На закладке «Общие» необходимо выбрать:
1. Базу данных для восстановления (вашу MMM)
  2. Выбрать источник набора данных для восстановления с устройства  $\diamond$  файл C:\...TEST\AW.BAK
  3. После определения файла-источника данных необходимо флажком выбрать базу данных для восстановления (рисунок 7).

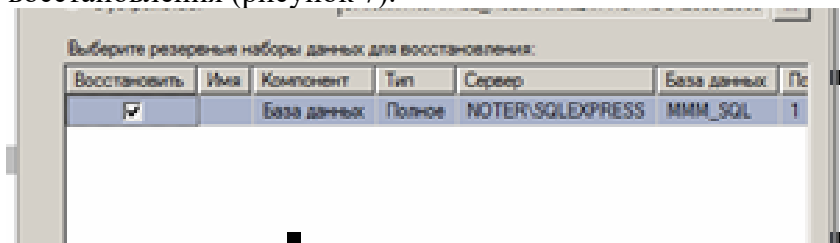


Рисунок 7- Выбор БД для восстановления

На закладке «Параметры»

1. необходимо включить опцию «Перезаписать БД» и «оставить БД готовой к использованию», (рисунок 8).

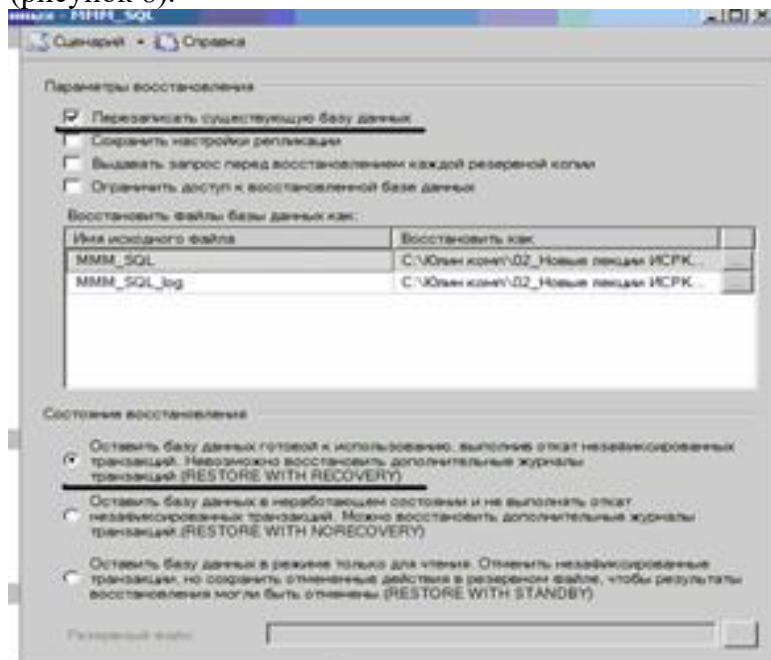


Рисунок 8 – Задание параметров восстановления

3. Нажмите OK

- После восстановления БД, откройте таблицу «Модель» и убедитесь, что она не содержит всех добавлений, вносимых вами в процессе выполнения упражнения, так как восстановление происходило из первой резервной копии (без изменений).

**Задание №3.** необходимо организовывать со стороны клиентского приложения, созданного в Visual Studio удаленное администрирование БД (резервное копирование).

Ход работы:

### В Visual Studio

- Создайте новый проект Windows Application и сохраните его в своей папке под именем Лабы\_MMM\_2 семестр.
- В главную форму добавьте меню, изображенное на рисунке 9:

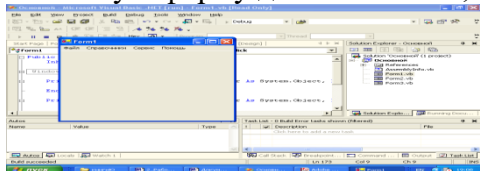


Рисунок 9 – Главное меню проекта

*Файл (Открыть, Закрыть, Выход)*

*Справочники (Модель, Магазин, Дерево моделей)*

*Заказы (Работа с заказами)*

*Отчеты (Прайс-лист, Бланк заказов)*

*Администрирование БД (Резервное копирование, Безопасность)*

*Сервис (Калькулятор)*

*Помощь (Справка, О программе)*

- Добавьте новую форму в проект
- Добавьте на только что созданную форму компоненты в соответствии с рисунком 10.

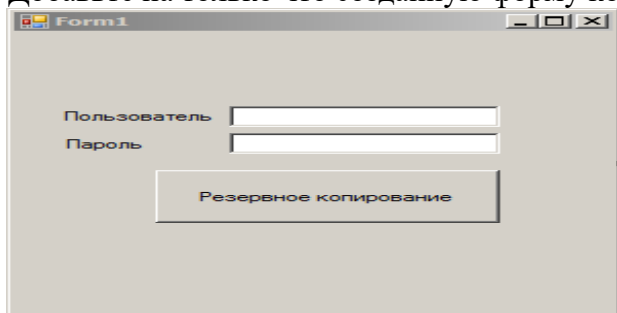


Рисунок 10 – Форма для подключения к серверу

- Обеспечьте функциональную работу формы (напишите обработчик кнопки «Резервное копирование» с использованием объектов SMO. Описание объектов SMO, их свойств и методов см. в лекционном материале.)
- Добавьте возможность открытия данной формы при выборе в главной форме пункта меню Администрирование БД ▾ Резервное копирование
- Запустите проект, проверьте работу формы.
- Закройте проект
- Убедитесь в появлении файла резервной копии на диске (файл, который указан в тексте программы).
- Откройте SSMS. Добавьте в таблицу «Модель» новую строку данных (самостоятельно).
- Средствами оболочки SSMS, выполните восстановление БД из резервной копии, созданной вашей программой
- Убедитесь, что после восстановления добавленных строк в таблице «Модель» нет.

**Задание №4.** Ответьте на вопросы теста и представьте результаты преподавателю.

- Вы выполняете разностное резервное копирование базы данных AdventureWorks каждые четыре часа, начиная с 04:00. полная резервная копия создается в полночь. Какие данные будут содержаться в разностной резервной копии сделанной в полдень?
  - А Страницы данных, измененные после полуночи.
  - В. Экстенды, измененные после полуночи.
  - С. Страницы данных, измененные после 08:00

4. D. Экстенты, измененные после 08:00.

2. Вы выполняете полное резервное копирование базы данных Adventure Works,, которое завершается в полночь. Разностное резервное копирование выполняется по расписанию каждые четыре часа, начиная с 04:00. Резервное копирование журнала транзакций происходит по расписанию каждые пять минут. Какую информацию будет содержать резервная копия журнала транзакций, созданная в 09:15?

1. A. Все транзакции, начатые после 09:10.
2. B. Транзакции, завершённые после 09:10.
3. C. Страницы, изменённые после 09:10.
4. D. Экстенты, изменённые после 09:10.

#### Отчет

Сохраните работу.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### Лабораторное занятие №3

**Тема:** Восстановление базы данных (2 часа)

#### Цель занятия:

1.Получение практических навыков администрирования и сопровождения логической и физической структур базы данных

#### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

Лабораторная работа направлена на ознакомление с основами администрирования СУБД Oracle: работа с табличными пространствами, файлами данных, резервное копирование и восстановление базы данных.

Требования к выполнению лабораторного практикума:

- Выполнять все действия строго в том порядке, указанном в задании;
- Составить отчет о проделанной работе;
- В отчете должны содержаться снимки экрана, показывающие процесс выполнения заданий;
- Отчет должен содержать выводы по результатам каждого выполненного задания;
- Работа должна быть выполнена согласно требованиям к выполнению лабораторной работы;

- Особое внимание следует уделить правильности задания переменной окружения ORACLE\_SID; перед выполнением лабораторной работы рекомендуется сделать резервную копию файлов, расположенных в каталоге /opt/oracle/OraHome1/oradata/testN, N=1..10.

Перед выполнением упражнений вам предстоит выполнить следующие действия:

Проверьте переменную окружения ORACLE\_SID:

```
echo $ORACLE_SID;
```

Задайте

```
ORACLE_SID=testN, N=1..10, выполнив команду
```

```
#export ORACLE_SID=testN;
```

3. Подсоединитесь к БД с помощью программы svrmgrl

```
#svrmgrl
```

```
SVRMGR> connect internal
```

При составлении и оформлении отчета следует придерживаться «Требований к оформлению дипломных, курсовых, лабораторных работ», расположенных на странице <http://unesco.kemsu.ru/student/rule/rule.html>.

### 3. Теоретический материал

#### Введение

Для небольшой базы данных достаточно создать одно табличное пространство SYSTEM; однако, Oracle рекомендует создавать дополнительные табличные пространства для хранения данных и индексов пользователя, сегментов отмены, временных сегментов отдельно от словаря данных. Это обеспечивает вам большую гибкость в выполнении различных задач администрирования и уменьшает конкуренцию при обращении к объектам словаря и схемы.

Администратор может создавать новые табличные пространства, изменять размер файлов данных, добавлять файлы к табличным пространствам, устанавливать и изменять параметры хранения по умолчанию сегментов в табличном пространстве, переводить табличное пространство в состояние «только чтение» или «чтение-запись», делать табличное пространство временным или постоянным или удалить его.

#### Табличное пространство system и другие

1. Табличное пространство system:

- создается во время создания базы данных
- содержит словарь данных
- содержит сегмент отмены system

2. Другие табличные пространства:

- отделяют сегменты
- обеспечивают большую гибкость решения задач администрирования пространства
- дают возможность контролировать выделение пространства пользователю

#### Создание табличных пространств

Табличное пространство может быть создано при помощи следующей команды:

```
CREATE TABLESPACE табличное_пространство
```

```
[DATAFILE фраза_файла_данных]
```

```
[MINIMUM EXTENT целое[K|M]]
```

```
[BLOCKSIZE целое [K]]
```

```
[LOGGING|NOLOGGING]
```

```
[DEFAULT фраза_хранения ]
```

```
[ONLINE | OFFLINE]
```

```
[PERMANENT | TEMPORARY]
```

```
[extent_management_clause]
```

```
[autoextend_clause]
```

Файлы параметров инициализации:

табл\_пространство – имя табличного пространства, которое требуется создать.

DATAFILE – задает файл или файлы данных, составляющие это табличное пространство. Для временных табличных пространств можно использовать TEMPFILE.

MINIMUM EXTENT – обеспечивает то, что размер каждого экстенда этого табличного пространства кратен целому (используйте К и М для указания размера в килобайтах и мегабайтах).

BLOCKSIZE – указывает размер блока данных, с которым будет создано табличное пространство. Необходимо указать параметр инициализации DB\_nK\_CACHE\_SIZE (n- 2,4,8,16 или 32, размер блока) для этого размера блока. Он устанавливает размер кэша буферов для обслуживания табличных пространств с указанным размером блока. Можно указать до 4 параметров. По умолчанию используется стандартный размер блока и кэш буферов по умолчанию, заданный параметром инициализации DB\_CACHE\_SIZE.

LOGGING – указывает, что по умолчанию все изменения таблиц, индексов и секций табличного пространства записываются в журнал (режим LOGGIN установлен в команде по умолчанию).

NOLOGGING – указывает, что по умолчанию все изменения таблиц, индексов и секций табличного пространства не записываются в журнал (режим NOLOGGIN затрагивает только некоторые команды DML и DDL, например, использующие прямую загрузку).

DEFAULT – задает параметры хранения по умолчанию для всех объектов, которые будут созданы в данном табличном пространстве.

ONLINE – делает табличное пространство доступным сразу после создания.

OFFLINE – сразу после создания табличное пространство будет недоступно.

PERMANENT – указывает на то, что это табличное пространство может быть использовано для хранения постоянных объектов.

TEMPORARY – указывает на то, что данное табличное пространство может хранить только временные объекты, например, сегменты, используемые фразой ORDER BY для неявной сортировки. Используется стандартный размер блока.

SIZE – задает размер файла (используйте К и М для задания размера файла).

REUSE – разрешает серверу Oracle повторно использовать существующий файл.

autoextend\_clause OFF/ON – разрешает или запрещает автоматическое расширение файла данных: NEXT- какими кусками будет расширяться файл, MAXSIZE/UNLIMITED- до какого максимального размера.

Пример создания нового табличного пространства:

```
CREATE TABLESPACE userdata  
DATAFILE 'u01/oradata/userdata01.dbf'  
SIZE 100M  
AUTOEXTEND ON NEXT 5M  
MAXSIZE 200M;
```

### **Табличные пространства «только для чтения»**

Команда alter tablespace...read only.

Перевод табличного пространства в режим только для чтения запрещает последующие операции записи в файлы данных. Табличные пространства «только для чтения» используются для предотвращения каких-либо изменений и для отмены необходимости выполнять резервирование и восстановление больших, статичных областей базы данных. Сервер Oracle никогда не обновляет файлы табличного пространства, используемого только для чтения, и, поэтому эти файлы могут располагаться на носителях, запись на которые невозможна, таких как CD-ROM.

Табличное пространство может быть переведено в режим только для чтения или «чтение-запись» при помощи команды ALTER TABLESPACE:

```
ALTER TABLESPACE табличное_пространство READ [ONLY | WRITE]
```

### **Перевод табличного пространства в режим**

#### **«только чтение»**

Команда ALTER TABLESPACE...READ ONLY переводит табличное пространство в режим «только чтение», не дожидаясь завершения всех активных транзакций. В этом режиме не разрешаются никакие последующие операции записи в табличное пространство, за исключением отката текущих транзакций, которые до этого модифицировали блоки табличного пространства.

После того, как выполнится **фиксация или откат всех текущих транзакций**, команда **alter tablespace ... read only** завершается и табличное пространство переводится в режим «только чтение».

Вы можете удалять из табличного пространства «только чтение» такие объекты, как таблицы и индексы, так как эти команды вносят изменения только в словарь данных, но не в файлы данных табличного пространства.

Перед переводом табличного пространства «только чтение» в режим «чтение-запись», все файлы данных табличного пространства должны быть в оперативном режиме.

- Перевод табличного пространства в режим «только чтение»
- Активизирует контрольную точку для файлов данных табличного пространства.

#### **Автономный режим**

Табличное пространство, находящееся в автономном режиме, не разрешает доступа к данным.

Некоторые табличные пространства должны находиться всегда в оперативном режиме:

- SYSTEM;
- табличные пространства, содержащие активные сегменты отмены;
- временное табличное пространство по умолчанию;

Перевод в автономный режим: ALTER TABLESPACE userdata OFFLINE;

Перевод в оперативный режим: ALTER TABLESPACE userdata ONLINE;

#### **Перевод табличных пространств в автономный режим (offline)**

Пользователи могут получить доступ к табличному пространству, только если оно находится в оперативном режиме. Табличное пространство может быть переведено администратором базы данных в автономный режим для того, чтобы:

- сделать недоступной часть базы данных, тогда как оставшаяся ее часть будет работать в нормальном режиме;
- выполнить резервирование табличного пространства в автономном режиме (хотя можно производить резервирование табличного пространства, которое находится в оперативном режиме и используется);
- восстановить табличное пространство или файл данных, когда база данных открыта;
- изменить местоположение файлов данных, когда база данных открыта.

#### **Автономный режим табличного пространства**

- Сервер Oracle не позволяет никаким командам SQL выполнять операции над объектами, содержащимися в автономном табличном пространстве. Если пользователи пытаются получить доступ к объектам автономного табличного пространства либо непосредственно, либо при проверке ссылочной целостности, они получают сообщение об ошибке.

- Информация о переходе табличного пространства в автономный режим или о возвращении в оперативный сохраняется в словаре данных и в управляющих файлах. Если табличное пространство находится в автономном режиме во время остановки базы данных, то оно останется таковым и не будет проверяться при последующем монтировании и открытии базы данных.

- Экземпляр Oracle автоматически переключает табличное пространство из оперативного режима в автономный, когда возникают ошибки определенного вида (например, когда процесс Database Writer в ходе нескольких попыток не может произвести запись в файл данных табличного пространства).

#### **Изменение размера табличного пространства**

1. **Добавление** файлов данных
2. **Изменение** размеров файла данных:

- **автоматически**
- **вручную**

#### **Установка автоматического расширения файлов данных**

#### **Указание параметра autoextent для нового файла данных**

В следующих командах с помощью фразы AUTOEXTEND включается или отключается автоматическое расширение файла данных:

- CREATE DATABASE
- CREATE TABLESPACE ... DATAFILE

## - ALTER TABLESPACE ... ADD DATAFILE

Используйте команду ALTER DATABASE, чтобы изменить файл данных и предоставить возможностью его автоматического расширения:

ALTER DATABASE DATAFILE спецификация\_файла [фраза\_авторасширения].

Если в табличном пространстве существует несколько файлов, расширяться будет тот, в котором, сервер захочет выделить экстенст. Если в файле нет места, и он не может расширяться, будет взят другой файл. Если ни в одном файле нет места, и они не могут расширяться дальше, пользователь, чья команда требует, расширения сегмента получит ошибку.

фраза\_авторасширения ::= [ AUTOEXTEND { OFF | ON [NEXT целое [K |M]] [MAXSIZE UNLIMITED | целое[K|M]] } ],

где:

AUTOEXTEND OFF выключает автоматическое расширение файла данных.

AUTOEXTEND ON включает автоматическое расширение файла данных. NEXT устанавливает размер выделяемого дискового пространства, когда требуются дополнительные экстенсты.

MAX SIZE определяет максимальный размер дискового пространства, который может быть выделен файлу данных.

UNLIMITED снимает ограничение на максимальный размер дискового пространства для файла данных.

Пример установки автоматического расширения файла данных:

ALTER DATABASE DATAFILE

'u01/oradata/app\_data\_04.dbf'

SIZE 200M AUTOEXTEND ON NEXT 10M MAXSIZE 500M;

## Изменение установки autoextend для существующего файла данных

Для включения или отключения автоматического расширения существующего файла данных используется команда ALTER DATABASE:

ALTER DATABASE [database] DATAFILE 'имя\_файла','имя\_файла']...фраза\_авторасширения

Определение параметров AUTOEXTEND:

DBA\_DATA\_FILES есть столбцы, показывающие параметры Авторасширения. Столбец AUTOEXTENSIBLE показывает включено или нет авторасширение:

SQL> select tablespace\_name, file\_name, autoextensible from dba\_data\_files;

Например:

| TABLESPACE_NAME | FILE_NAME | AUTOEXTENSIBLE |
|-----------------|-----------|----------------|
|-----------------|-----------|----------------|

|        |                                      |     |
|--------|--------------------------------------|-----|
| SYSTEM | /home/dba01/ORADATA/u01/system01.dbf | YES |
|--------|--------------------------------------|-----|

|        |                                    |    |
|--------|------------------------------------|----|
| DATA01 | /home/dba01/ORADATA/u04/data01.dbf | NO |
|--------|------------------------------------|----|

|       |                                     |    |
|-------|-------------------------------------|----|
| USERS | /home/dba01/ORADATA/u03/users01.dbf | NO |
|-------|-------------------------------------|----|

|        |                                   |    |
|--------|-----------------------------------|----|
| UNDO02 | /home/dba01/ORADATA/u01/UND02.dbf | NO |
|--------|-----------------------------------|----|

## Изменение размера файлов данных вручную

Команда ALTER DATABASE DATAFILE RESIZE

Администратор может изменить размер существующего файла данных вместо того, чтобы увеличивать пространство базы данных при помощи создания новых файлов. Администратор может исправить ошибки, допущенные при планировании базы данных, и освободить неиспользуемое пространство. Для того чтобы уменьшить или увеличить размер файла данных вручную используется команда ALTER DATABASE следующего вида:

ALTER DATABASE [база\_данных]

DATAFILE 'имя\_файла','имя\_файла']...

RESIZE целое[K|M]

где: целое- требуемый размер файла данных.

Если объекты базы данных располагаются в файле данных за указанным размером, то файл данных уменьшается только до последнего блока последнего объекта базы данных.

## Добавление файлов данных к табличному пространству

Команда ALTER TABLESPACE ADD DATAFILE

При помощи следующей команды ALTER TABLESPACE ADD можно добавить к табличному пространству файл данных, чтобы увеличить общее количество дискового пространства, отведенного для этого табличного пространства :

```
ALTER TABLESPACE табличное_пространство
ADD [DATAFILE I TEMPFILE ]
спецификация_файла [фраза_авторасширения]
[,спецификация_файла [фраза_авторасширения]]...
```

Файлы добавляются, если в текущем разделе диска нет места или превышено ограничение на максимальный размер файла в операционной системе.

### **ALTER TABLESPACE: перемещение файлов данных**

Методы перемещение файлов данных.

Администратор базы данных может изменять местоположение файлов данных в зависимости от вида табличного пространства одним из следующих двух способов: при помощи команд ALTER TABLESPACE или ALTER DATABASE.

Использование команды ALTER TABLESPACE.

Команда ALTER TABLESPACE следующего вида применяется только для файлов данных, не принадлежащих табличному пространству SYSTEM, которое, к тому же, не содержит активных сегментов отмены или временных сегментов. Более того, она работает только в режиме открытой базы данных.

```
ALTER TABESPACE табличное_пространство RENAME DATAFILE 'имя_файла'[,
'имя_файла']... TO 'имя_файла'[, 'имя_файла' ]
```

Для переименования файла данных выполняется следующая последовательность шагов:

1. Переведите табличное пространство в автономный режим.
2. Переместите или скопируйте файлы при помощи команд операционной системы.
3. Выполните команду ALTER TABLESPACE RENAME DATAFILE.
4. Верните табличное пространство в оперативный режим.
5. Если требуется, удалите файл при помощи команды операционной системы. Имя исходного файла должно совпадать с именем в управляющем файле.

### **ALTER DATABASE: перемещение файлов данных**

Для перемещения любого вида файла данных может быть использована команда ALTER DATABASE (см. занятие "Сопровождение журнальных файлов"). В отличие от предыдущей команды, она работает как в режиме открытой базы данных, так и в режиме смонтированной базы.

```
ALTER DATABASE [база_данных]RENAME FILE
'имя_файла'[, 'имя_файла']... TO 'имя_файла'[, 'имя_файла']...
```

Файлы табличного пространства SYSTEM , которые не могут быть переведены в автономный режим, переименовываются следующим образом:

1. Остановите экземпляр.
2. Переместите файлы при помощи команды операционной системы.
3. Смонтируйте базу данных.
4. Выполните команду ALTER DATABASE RENAME FILE.
5. Откройте базу данных.

### **Удаление табличных пространств**

- Нельзя удалять табличное пространство SYSTEM и содержащее активные сегменты отката.
- Информация о табличном пространстве удаляется из словаря данных.
- В команде удаления табличного пространства необходимо указывать режим удаления его содержимого.
- При помощи команды DROP TABLESPACE табличные пространства можно удалить из базы данных, когда отпала надобность в них самих и в их содержимом:

```
DROP TABLESPACE табличное_пространство
[INCLUDING CONTENTS [AND DATAFILES] [CASCADE CONSTRAINTS]]
```

где:

- табличное\_пространство - имя табличного пространства, которое требуется удалить
- INCLUDING CONTENTS - удаляет все сегменты табличного пространства

- AND DATAFILES - удаляет соответствующие файлы ОС
- CASCADE CONSTRAINTS - удаляет те ссылочные правила целостности таблиц из других табличных пространств, которые ссылаются на первичные и уникальные ключи таблиц, принадлежащих удаляемому табличному пространству

Табличное пространство, содержащее данные, не может быть удалено без использования параметра INCLUDING CONTENTS. Использование этого параметра может привести к генерации большого объема информации отмены, если в табличном пространстве находится много объектов.

Данные табличного пространства перестают существовать в базе данных, как только оно удаляется.

При удалении табличного пространства удаляются только файловые указатели из управляющего файла соответствующей базы данных. Файлы базы данных остаются и должны быть удалены явно на уровне операционной системы, если в команде отсутствует фраза AND DATAFILES

Табличное пространство в режиме только для чтения тоже может быть удалено вместе со своими сегментами. Это возможно потому, что команда DROP обновляет словарь данных (который должен быть в режиме чтение-запись), а не физические файлы, составляющие базу данных.

Для того чтобы во время удаления табличного пространства не существовало транзакций, пытающихся получить доступ к сегментам этого табличного пространства, рекомендуется перевести его в автономный режим.

### Получение информации о табличном пространстве

Информация о табличном пространстве:

- DBA\_TABLESPACES
- V\$TABLESPACE

Информация о файле данных:

- DBA\_DATA\_FILES
- V\$DATAFILE

Информация о временном файле:

- DBA\_TEMP\_FILES
- V\$TEMPFILE

### Резервное копирование и восстановление

#### Восстановление вручную

1. Выполнить физическое восстановление файла означает заменить его резервной копией.
2. Восстановлению обычно подлежат следующие файлы:
  - файлы данных;
  - управляющие файлы;
  - архивные журнальные файлы;
  - серверный файл параметров.
3. В каждом случае потеря основного файла и восстановление его из резервной копии приводит к следующим последствиям при восстановлении носителя.

| Если теряется...                      | То...                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Один или несколько файлов данных      | Необходимо восстановит их из резервной копии и<br>Выполнить восстановление носителя. Восстановление (носителя) необходимо, когда SCN контрольной точки в заголовке файла данных не совпадает с SCN контрольной точки файла, отраженного в управляющем файле.                                                                                                                                                                                                                  |
| Все копии текущего управляющего файла | Необходимо восстановить управляющий файл из резервной копии и затем открыть базу данных в режиме RESETLOGS. Если резервная копия отсутствует, можно попытаться повторно создать управляющий файл. По возможности, следует использовать командный файл, включенный в выходные данные оператора ALTER DATABASE BACKUP CONTROLFILE TO TRACE. Чтобы добиться соответствия структуры управляющего файла текущей структуре базы данных могут потребоваться дополнительные действия. |
| Одна мультиточечная копия             | Необходимо скопировать один из нетронутых управляющих файлов в то место, где находится отсутствующий или поврежденный управляющий файл, и                                                                                                                                                                                                                                                                                                                                     |

|                                                                    |                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| плексерованного управляющего файла                                 | открыть базу данных. Если не удаются скопировать управляющий файл в его исходное местоположение (например, если не удастся починить дисковод), следует указать новое местоположение в файле параметров инициализации, а затем открыть базу данных.                                                                                                                                                            |
| Архивные журнальные файлы, необходимые для восстановления носителя | Необходимо восстановить эти архивные журнальные файлы из резервных копий для восстановления носителя. Журнальные файлы можно восстановить либо в местоположение по умолчанию, либо в указанный каталог. Если резервные копии отсутствуют, необходимо выполнить неполное восстановление до той точки, после которой требуется информация первого потерянного журнала, и открыть базу данных в режиме RESETLOGS |
| Серверный файл параметров                                          | Если имеется резервная копия серверного файла параметров, восстановите этот файл из нее. С другой стороны, если имеется резервная копия «клиентского» (т.е. текстового) файла параметров инициализации, можно восстановить этот файл, запустить с его помощью экземпляр и повторно создать серверный файл параметров                                                                                          |

### **Хранение записей для последующего использования в ходе восстановления**

Одним из наиболее важных аспектов резервирования и восстановления, управляемого пользователем, является хранение записей обо всех файлах текущей базы данных и резервных копиях этих файлов. Например, у вас должны быть записи о местоположении следующих файлов:

- Файлы данных;
- Управляющие файлы;
- Оперативные журнальные файлы (обратите внимание, что оперативные журналы не резервируются);
- Архивные журнальные файлы;
- Файлы параметров инициализации;
- Файлы паролей;
- Файлы настроек сетевых компонентов;

Запись местоположения файлов данных, управляющих файлов и оперативных журнальных файлов

Следующий полезный командный файл SQL выводит местоположение всех управляющих файлов, файлов данных и оперативных журнальных файлов базы данных:

```
SELECT NAME FROM V$DATAFILE
UNION ALL
SELECT MEMBER FROM V$LOGFILE
UNION ALL
SELECT NAME FROM V$CONTROFILE;
```

Выходные данные могут выглядеть следующим образом:  
NAME

```
-----
/oracle/dbs/tbs_01.f
/oracle/dbs/tbs_02.f
/oracle/dbs/tbs_11.f
/oracle/dbs/tbs_12.f
/oracle/dbs/t1_log1.f
/oracle/dbs/t1_log2.f
/oracle/dbs/cf1.f
/oracle/dbs/cf2.f
```

### **Запись местоположений резервных копий файлов**

Недостаточно просто записывать местоположение резервных копий файлов: необходимо устанавливать соответствие между резервными копиями и исходными файлами. По возможности, присваивайте резервным копиям такие же относительные имена, как и у основных файлов. Независимо от используемой системы именования, храните таблицу с соответствующей

информацией. Например, можно хранить следующую таблицу с информацией о местоположении файлов базы данных на случай восстановления.

| Номер<br>данных      | файлов                  | Табличное<br>пространство | Имя резервного файла   |
|----------------------|-------------------------|---------------------------|------------------------|
| 0 (управляющий файл) | 0 (управляющий<br>файл) |                           | /disk3/backup/cf.f     |
| 1                    | SYSTEM                  |                           | /disk3/backup/tbs_01.f |
| 2                    | undo                    |                           | /disk3/backup/tbs_02.f |
| 3                    | temp                    |                           | /disk3/backup/tbs_11.f |
| 4                    | users                   |                           | /disk3/backup/tbs_12.f |

### Определение файлов данных, требующих восстановления

Динамическое представление производительности V\$RECOVER\_FILE позволяет определить, какие файлы нужно восстановить из резервных копий при подготовке к восстановлению носителя. В этом представлении содержатся все файлы, требующие восстановления, и даются объяснения, почему они Должны быть восстановлены.

Следующий запрос отображает идентификационные номера файлов данных, требуемых для восстановления носителя, а также причину восстановления (если она известна), а также ЗСК и время, с которых нужно начать восстановление:

```
SELECT * FROM V$RECOVER_FILE;
FILE#  ONLINE  ERROR CHANGE#  TIME
```

```
-----
14 ONLINE
15 ONLINE FILE NOT FIND      0
21 ONLINE OFFLINE NORMAL    0
```

### Повторное создание файлов данных при отсутствии резервных копий

Когда файл данных поврежден, а его резервная копия недоступна, этот файл данных все же можно восстановить, если:

- доступны все архивные журнальные файлы, сгенерированные после создания исходного файла данных;
- управляющий файл содержит имя поврежденного файла (это значит, что управляющий файл является текущим или восстановленным из резервной копии, сделанной после того, как поврежденный файл данных был добавлен в базу данных).

Чтобы повторно создать файл данных для восстановления:

1. Создайте новый, пустой файл данных для замены поврежденного файла данных, у которого нет соответствующей резервной копии. Предположим, например, что поврежден файл данных /disk1/users1.f и его резервная копия недоступна. Следующий оператор повторно создает исходный файл данных (того же размера) на диске disk2:

```
ALTER DATABASE CREATE DATAFILE '/disk1/users1.f' AS '/disk2/users1.f';
```

Данный оператор создает пустой файл того же размера, что и потерянный файл. Oracle ищет информацию о размере файла в управляющем файле и словаре данных. Исходному файлу данных присваивается имя нового файла данных.

2. Выполните восстановление носителя для пустого файла данных. Например, введите:

```
RECOVER DATAFILE '/disk2/users1.f'
```

3. Все архивные журналы, сгенерированные после создания исходного файла данных, должны быть доступны и повторно применены к новой, пустой версии потерянного файла данных во время восстановления.

## Восстановление и повторное создание управляющих файлов

Если сбой носителя повредил управляющие файлы базы данных (независимо от того, мультиплексированы они или нет), база данных продолжает работать до тех пор, пока какому-либо процессу сервера Oracle не потребуется получить доступ к управляющим файлам. В этот момент база данных и экземпляр автоматически останавливаются.

Если сбой носителя был временным и база данных до сих пор не остановлена, не допускайте автоматической остановки базы данных – немедленно исправляйте сбой. Однако если остановка базы данных происходит до исправления временного сбоя носителя, ее можно повторно запустить после устранения проблемы и восстановления доступа к управляющим файлам.

Процедура восстановления после сбоя носителя, который делает невозможным доступ к управляющим файлам базы данных, зависит от того, мультиплексированы эти управляющие файлы или нет. Соответствующие процедуры описаны в следующих разделах:

- Потеря одного из элементов мультиплексированного управляющего файла.
- Потеря всех элементов мультиплексированного управляющего файла, когда резервная копия доступна.
- Потеря всех текущих и резервных управляющих файлов.

### Потеря одного из элементов мультиплексированного управляющего файла

Следующие процедуры используются для восстановления базы данных после сбоя носителя, который привел к повреждению одного или нескольких управляющих файлов базы данных, но при этом как минимум один управляющий файл *остался неповрежденным*.

Копирование мультиплексированного управляющего файла в местоположение по умолчанию

Если диск и файловая система, содержавшие потерянный управляющий файл, остались невредимы, можно просто скопировать один из неповрежденных управляющих файлов туда, где находился потерянный управляющий файл. В этом случае вам не придется изменять значение параметра инициализации CONTROL\_FILES.

Чтобы заменить поврежденный управляющий файл путем копирования мультиплексированного управляющего файла:

1. Остановить экземпляр, если он еще работает: SHUTDOWN ABORT.
2. Устраните аппаратную проблему, которая привела к сбою носителя. Если не удастся быстро решить эту проблему, переходите к восстановлению базы данных – восстановите поврежденный управляющий файл на другом запоминающем устройстве, как описано в разделе «Копирование мультиплексированного управляющего файла в местоположение, отличное от используемого по умолчанию».
3. Замените поврежденные управляющие файлы неповрежденной мультиплексированной копией текущего управляющего файла базы данных. Например, чтобы заменить файл bad\_cf.f файлом good\_cf.f введите: % cp /oracle/good\_cf.f /oracle/dbs/bad\_cf.bad
4. Запустите новый экземпляр, смонтируйте и откройте базу данных. Например, введите: STARTUP

### Потеря всех элементов мультиплексированного запоминающего файла, когда резервная копия доступна

Следующие процедуры для восстановления управляющего файла из резервной копии при сбое носителя, который привел к повреждению всех управляющих файлов базы данных. Если управляющий файл недоступен, можно запустить экземпляр, но не монтировать базу данных. Если попытаться смонтировать базу данных, когда управляющий файл недоступен, появится следующее сообщение об ошибке:

ORA-00205: ошибка определения управляющего файла, дополнительную информацию см. в сигнальном файле ALERT

Нельзя смонтировать и открыть базу данных до тех пор, пока управляющий файл не будет снова доступным. Если управляющий файл восстанавливается из резервной копии, необходимо открыть базу данных с опцией RESETLOGS.

Как видно из Таблицы, процедура восстановления управляющего файла зависит от того, доступны ли журналы.

**Таблица 1- Сценарии восстановления потерянных управляющих файлов**

| Статус оперативных журналов | Статус файлов данных | Действия                                                                                                                                                                                                                                                                                                                                 |
|-----------------------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Доступны                    | Текущие              | Если оперативные журналы содержат необходимые для восстановления, эти журналы используются в процессе восстановления управляющего файла из резервной копии. Следовательно, необходимо указать имена оперативных журналов, содержащих изменения, чтобы открыть базу данных. После восстановления откройте базу данных в режиме RESETLOGS. |
| Недоступны                  | Текущие              | Если оперативные журналы содержат записи, необходимые для восстановления, вы должны повторно создать управляющий файл. Поскольку журналы недоступны, откройте базу данных в режиме RESETLOGS.                                                                                                                                            |
| Доступны                    | Резервные            | Восстановите управляющий файл из резервной копии, выполните полное восстановление и откройте базу данных в режиме RESETLOGS.                                                                                                                                                                                                             |
| Недоступны                  | Резервные            | Восстановите управляющий файл из резервной копии, выполните неполное восстановление и откройте базу данных в режиме RESETLOGS.                                                                                                                                                                                                           |

#### ВОССТАНОВЛЕНИЕ УПРАВЛЯЮЩЕГО ФАЙЛА ИЗ РЕЗЕРВНОЙ КОПИИ В МЕСТОПОЛОЖЕНИИ ПО УМОЛЧАНИЮ

По возможности, восстанавливайте управляющий файл в его исходном местоположении. В этом случае вам не придется указывать новые местоположения управляющего файла в файле параметров инициализации.

Чтобы восстановить управляющий файл в местоположении по умолчанию:

1. Остановите экземпляр, если он еще работает: SHUTDOWN ABORT
2. Устраните аппаратную проблему, которая привела к сбою носителя.
3. Восстановите управляющий файл из резервной копии во всех местоположениях, указанных в параметре инициализации CONTROL\_FILES. Например, если в файле параметров сервера указаны местоположения управляющего файла /dsk/oracle/dbs/cf1.f и /dsk/cf2.f, используйте утилиту операционной системы для восстановления управляющего файла в этих местоположениях:

```
% cp /backup/cf.bak /dsk1 /oracle/dbs/cf1.f
```

```
% cp /backup/cf.bak /dsk2/cf2.f
```

4. Запустите новый экземпляр и смонтируйте базу данных. Например, введите: STARTUP MOUNT

5. Начните восстановление выполнением оператора RECOVER с предложением USING BACKUP CONTROLFILE. Укажите предложение UNTIL CANCEL, если вы выполняете неполное восстановление. Например, введите:

```
RECOVER DATABASE USING BACKUP CONTROLFILE UNTIL CANCEL
```

6. Примените запрошенные архивные журналы. Если появится сообщение о том, что требуемый архивный журнал отсутствует, значит, необходимая запись, вероятно, находится в оперативном журнале. Эта ситуация может возникнуть, если незаархивированные изменения находились в оперативных журналах, когда произошел сбой экземпляра. Например, предположим, что вы видите следующее сообщение:

```
ORA-00279: изменение 55636, созданное 06/08/2000 в 16:59:47, необходимое для потока 1
```

```
ORA-00289: предложение /oracle/work/arc_dest/arcr_1_111.arc
```

```
ORA-00280: изменение 55636 для потока 1 находится в последовательности #111
```

```
Задайте журнал: (<RET>=предлагаемое | имя файла | AUTO | CANCEL)
```

Можно указать имя оперативного журнала и нажать Enter (возможно, придется сделать это несколько раз до тех пор, пока не будет найден нужный журнал):

```
/oracle/dbs/t1_log1.f
```

Журнал применен.

Восстановление носителя завершено.

Если по каким-либо причинам оперативные журналы недоступны, можно прекратить восстановление и не применять оперативные журналы. Обратите внимание, что если все файлы данных являются текущими и требуемые для восстановления записи находятся в оперативных журналах, базу данных нельзя открыть без применения оперативных журналов. Если оперативные журналы недоступны, необходимо повторно создать управляющий файл.

7. Откройте базу данных в режиме RESETLOGS после завершения восстановления:

ALTER DATABASE OPENRESETLOGS;

#### 4. Порядок выполнения работы

1. Сопровождение табличных пространств и файлов данных

1. Создайте постоянные табличные пространства со следующими именами и параметрами хранения:

DATA01, управляемое с помощью словаря данных.

DATA02, с экстендами одинакового размера (размер каждого экстенда должен быть кратен 100 Кб.) (включите автоматическое расширение с выделением пространства размером 500 Кб и максимальным размером 2 Мб.

RONLY для таблиц, доступных только на чтение с параметрами хранения по умолчанию.

**НЕ СОЗДАВАЙТЕ** табличное пространство в режиме «только чтение» в данный момент времени.

2. Выведите информацию из словаря данных.

3. Выделите дополнительно 500Кб для табличного пространства DATA02 . Проверьте результат.

4. Переместите табличное пространство DATA01 в другой каталог (оба способа).

5. Добавьте файл данных для табличного пространства DATA01.

6. Измените размер файла данных для DATA01 вручную.

7. Создайте таблицу в табличном пространстве RONLY. Переведите RONLY в режим «только чтение».

8. Попробуйте создать еще одну таблицу. Удалите первую таблицу. Что произошло и почему?

9. Удалите табличное пространство RONLY и соответствующий файл данных. Проверьте результат.

2. Резервное копирование и восстановление

1. Выполните резервное копирование управляющих файлов и файлов данных.

2. Удалите один из файлов данных.

3. Выполните восстановление удаленного файла путем создания нового файла данных.

4. Удалите все управляющие файлы.

5. Восстановите управляющие файлы из резервной копии.

6. Проверьте работоспособность БД.

#### Отчет

Цель работы

Введение

Программно-аппаратные средства, используемые при выполнении работы.

Основную часть (описание самой работы), выполненную согласно требованиям к результатам выполнения лабораторного практикума.

Заключение (выводы).

Список используемой литературы.

#### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала | Творческая составляющая             | Перевод баллов в оценку                   |
|-------------------------------------------------|-------------------------------|-------------------------------------|-------------------------------------------|
| Своевременно выполнена работа- 10               | Точность выполнения           | Защита работы аудиторно – 20 баллов | 70 б – оценка «5»<br>40-60 б – оценка «4» |

|                                                |                                                     |                                                           |                                                        |
|------------------------------------------------|-----------------------------------------------------|-----------------------------------------------------------|--------------------------------------------------------|
| баллов                                         | работы – 10<br>баллов                               |                                                           | 30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов        | Эстетичность выполнения работы – 10<br>баллов       | Защита своей работы персонально – 10<br>баллов            |                                                        |
| Не своевременно выполнена работа – 0<br>баллов | Полное содержание выполненной работы – 10<br>баллов | Оформление своей работы несколькими способами – 20 баллов |                                                        |

#### Лабораторное занятие №4

Тема: Восстановление носителей информации (2 часа)

#### Цель занятия:

1. Получение теоретических и практических навыков программного восстановления данных..

#### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

#### Восстановление данных TestDisk

**TestDisk** — свободная программа для восстановления данных, предназначенная прежде всего для восстановления потерянных разделов на носителях информации, а также для восстановления загрузочного сектора, после программных или человеческих ошибок (например, потеря MBR).

- Установка **<sudo apt-get install testdisk>**.
- Запускаем TestDisk **<sudo testdisk>**.
- Появляется окошко приветствия TestDisk, нам предлагается вести лог работы (для выполнения данной работы лог не требуется).
- Выбираем нужный диск и нажимаем **Enter**.
- Предлагается выбрать тип таблицы разделов, обычно TestDisk определяет все правильно, так что нажимаем **Enter**.
- Выбираем **Analise**.
- Выбираем **QuickSearch**.
- Нам выводят таблицу разделов. Выбираем раздел и нажимаем **P**, чтобы вывести список файлов.
- Выбираем файлы для восстановления и нажимаем **C**.
- Выбираем папку, куда будут сохранены файлы и нажимаем **C**.

#### Восстановление данных PhotoRec

**PhotoRec** - это утилита, входящая в состав пакета TestDisk. Предназначена для восстановления испорченных файлов с карт памяти цифровых фотоаппаратов (CompactFlash, Secure Digital, SmartMedia, Memory Stick, Microdrive, MMC), USB flash-дисков, жестких дисков и CD/DVD. Восстанавливает файлы большинства распространенных графических форматов, включая JPEG, аудио-файлы, включая MP3, файлы документов в форматах Microsoft Office, PDF и HTML, а также архивы, включая ZIP. Может работать с файловыми системами ext2, ext3, ext4 FAT, NTFS и HFS+, причем способна восстановить графические файлы даже в том случае, когда файловая система повреждена или отформатирована.

- Установка **<sudo apt-get install testdisk>**.
- Запускаем PhotoRec **<sudo photorec>**.
- Выбираем нужный диск и нажимаем **Enter**.

- В нижнем меню можно выбрать **File Opt**, чтобы выбрать типы файлов для восстановления (по умолчанию выбраны все).
- Чтобы начать восстановление нажмите **Enter**, выбрав **Search**.
- У нас выбрана система ext4, поэтому выбираем первый вариант [ ext2/ext3 ].
- Если выбрать пункт **FREE**, то поиск будет произведен в пустом пространстве и в этом случае будут восстановлены только удаленные файлы, а если выбрать **WHOLE**, то поиск будет произведен на всем диске.
- Теперь нужно указать директорию, куда будем сохранять нужные нам файлы. Выбираем нужную папку и нажимаем **C**.
- Выбираем файлы для восстановления и нажимаем **C**.

### Восстановление данных Extundelete

**Extundelete** – утилита, позволяющая восстанавливать файлы, которые были удалены с разделов ext3/ext4.

- Установка: **<sudo apt-get install extundelete>**.
- Как только вы поняли, что удалили нужные файлы, необходимо отмонтировать раздел: **<umount /dev/<partition> >**
- Зайдите в каталог, в который будут восстанавливаться удаленные данные. Он должен быть расположен на разделе отличном от того, на котором хранились восстанавливаемые данные: **cd /<путь\_к\_каталогу\_куда\_восстанавливать\_данные>**
- Запустите **extundelete**, указав раздел, с которого будет происходить восстановление и файл, который необходимо восстановить: **sudo extundelete /dev/<partition> --restore-file /<путь\_к\_файлу>/<имя\_файла>**
- Можно так же восстанавливать содержимое каталогов: **sudo extundelete /dev/<partition> --restore-directory /<путь\_к\_директории>**

### Восстановление данных Foremost

**Foremost** - консольная программа, позволяющая искать файлы на дисках или их образах по hex-данным, характерным заголовкам и окончаниям. Программа проверяет файлы на предмет совпадения заранее определённых hex-кодов (сигнатур), соответствующих наиболее распространённым форматам файлов. После чего экстрагирует их из диска/образа и складывает в каталог, вместе с подробным отчётом о том, чего, сколько и откуда было восстановлено. Типы файлов, которые foremost может сразу восстановить: jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm, crr. Есть возможность добавлять свои форматы (в конфигурационном файле /etc/foremost.conf), о которых программа не знает.

- Установка: **<sudo apt-get install foremost>**
- Пример использования для восстановления изображений с диска /dev/sdb в каталог ~/out\_dir: **<sudo foremost -t jpg,gif,png,bmp -i /dev/sdb -o ~/out\_dir>**

### Задания к лабораторной работе

- Добавьте в виртуальную машину виртуальный жесткий диск.
- Запустите виртуальную машину с Linux.
- Запустите fdisk (gdisk или parted) и создайте таблицу разделов MBR с разделами.
- Отформатируйте созданные разделы в файловую систему ext4.
- Установите TestDisk.
- Удалите MBR (или таблицу разделов) с помощью команды DD.
- Восстановите MBR (или таблицу разделов) с помощью TestDisk.
- Смонтируйте восстановленные разделы и создайте там произвольные файлы.
- Удалите созданные файлы.
- С помощью TestDisk восстановите данные.
- Создайте произвольный каталог и запишите туда данные каталога /var/log/ .
- Удалите данные с созданного каталога.
- С помощью PhotoRec восстановите данные.
- Создайте произвольный каталог и запишите туда данные каталога /etc/ .
- С помощью Extundelete или Foremost восстановите данные.

### Вопросы к лабораторной работе

1. С помощью какой из программ, используемых в этой лабораторной работе, можно восстановить таблицу разделов?
2. Какие файловые системы поддерживает PhotoRec?
3. Какие форматы поддерживает PhotoRec?
4. Как Foremost восстанавливает файлы?
5. Можно ли восстановить данные с файловой системы NTFS, используя extundelete?
6. Все ли данные скопированные с каталога /var/log/ восстановились?
7. Все ли данные скопированные с каталога /etc/ восстановились?

#### **Отчет**

1. Тема и цель работы
2. Экранные копии выполнения работы
3. Ответы на контрольные вопросы.

#### **Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### **Лабораторное занятие №5**

**Тема:** Восстановление удаленных файлов (2 часа)

#### **Цель занятия:**

1. Освоить способы восстановления случайно удаленных файлов путем их сборки по кластерам.

#### **ИНСТРУКЦИОННАЯ КАРТА**

#### **Материально - техническое оснащение**

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### **Практические задания**

В режиме ручного восстановления можно полностью или частично восстановить промаркированный файл. Данный режим целесообразно применять для сборки текстовых файлов. Частичное восстановление не текстовых файлов сопряжено с трудностями в поиске фрагментов двоичного файла, кроме того исполнительный файл не имеет никакой практической применения, если он будет восстановлен не полностью.

Минимальной информационной единицей, с которой работает утилита, является кластер. Таким образом, можно попытаться восстановить файл путем его сборки, но кластерам, однако нельзя изменять информацию внутри этих кластеров. Модифицировать информацию внутри кластера можно, например, с помощью утилиты Disk Editor.

1) Rename - данная команда позволяет осуществить переименование уже восстановленного и немаркированного на файловой панели файла или каталога.

2) UnErase To - команда предназначена для восстановления на файловой панели немаркированного файла с одновременной его пересылкой на другой диск. Данную операцию можно провести только над файлом, но не над каталогом. Кроме того, файл одновременно может быть переименован при необходимости. При этом пользователю вначале будет предложено выбрать нужный привод, а затем спецификацию файла.

Окно просмотра содержит подкоманды:

Add cluster - Добавить найденным кластер к файлу. Просмотрев содержимое, пользователь может использовать эту подкоманду для формирования файла, если это содержимое его устраивает;

Find next - Найти следующий подходящий файл. Программа продолжит поиск следующего кластера содержащего информацию, указанную в поисковой строке;

Hex - Просмотреть содержимое в шестнадцатеричном виде.

Done - подкоманда эквивалентная нажатию клавиши <Esc>.

Add - добавить данный кластер в файл;

Next - просмотреть следующий кластер с номером на единицу больше текущего;

Prev - просмотреть кластер с номером на единицу меньше текущего, т.е. возврат к предыдущему кластеру;

Go To - перейти к кластеру за номером, который необходимо указать на запрос программы;

Hex - просмотреть содержимое кластера в шестнадцатеричном виде;

Done - выход.

## 2. Порядок выполнения работы

1. Убедился в наличии файлов unerase.exe, unerase.hlp и nlib100.rtl. Запускаю утилиту на исполнение.

2. Переходим на диск D:\; файл: Project 1.dof - average:

размер - 2012;

дата -14.07.04 ;

время -22:15;

атрибут - арх;

начальный кластер - 635;

необходимо кластеров -4

3. Восстанавливаем файл Project 1.dof с присвоением ему имени «New».

4. Используя команду «Восстановить в» перенесём файл на диск A:\.

5. Покидаем утилиту и удаляем восстановленный файл.

6. Выберем текстовый файл Текст 1.txt содержащий кириллицу. Копируем и удаляем его с дискеты, затем восстановим его с помощью ручного восстановления ч-з утилиту Unerase.exe.

7. Добавляем в восстанавливаемый файл необходимые кластеры (Добавить кластеры-Все кластеры), для данного файла понадобилось: кластер 618,619.

8. Используя команду «Перенос кластера» меняем местами кластеры клавишей «Space» при этом информация в файле изменяется, т.е. изменяются последние данные (обратный порядок).

9. При использовании команды «Удаление кластера» удаляем 618 кластер, после чего файл теряет несколько Кб данных.

- Все выше указанные команды при восстановлении файла используют общую команду Save(сохранить).

10. Покидаем утилиту. Копируем на дискету несколько файлов размером не меньшим чем 2Кб и удаляем их.

11. Загружаем утилиту Unerase.exe, с помощью ручного восстановления и команды «Добавить возможный кластер» восстановим файл (при этом информация в файле восстанавливается по частям), то есть данная команда, это аналог команды «Все кластеры», только кластеры восстанавливаются по одному.

12. Восстанавливаем следующий файл используя команду «Обзор».

13. Создаём новый файл, используя команду «Создать файл». При этом файл должен содержать слово «когда». Для решения этой задачи используем команду «Поиск данных» (утилита сама подбирает кластеры так, чтобы информация файла содержала слово «когда»).

14. . Создаём новый файл, используя команду «Создать файл». При этом файл должен содержать кластеры: 10,12,320-325. Для решения этой задачи используем команду «Номер кластера»

(Указываем диапазон поиска кластеров). Полученный файл содержит информацию из различных файлов диска A:\.

15. Покидаем утилиту. Размеры файлов:

- файл с текстом «когда» 2Кб;

- файл с кластерами 1Кб;

16. Форматируем floppy disk на 720 Кб, копируем на неё файл «MyLab.doc» и несколько других файлов. Удаляем «MyLab.doc» и скопируем на дискету файлы: config.sys и autoexe.bat.

17. Загружаем утилиту Unerase.exe.

18. Для восстановления текстового документа необходимо 1888 кластеров, имеется начальный кластер (618). Методом ручного восстановления выбираем опцию «Все кластеры». Результат: найдены все кластеры диапазон от 618 до 2505. Сохраняем этот файл - он восстановлен.

19. Удаляем файлы config.sys и autoexe.bat. С помощью утилиты Unerase.exe проведём объединение этих файлов в один «Слитый». Для этого создаём файл и добавляем в него кластеры файлов(config.sys(618) и autoexe.bat(619)) - получаем файл содержащий два файла.

Вывод

В процессе выполнения работы выучил приёмы и режимы работы утилиты Unerase в ручном режиме работы. Даная утилита позволяет восстанавливать удалённые файлы по кластерам, как автоматически, так и по одному.

**Отчет**

1. Тема и цель работы

2. Экранные копии выполнения работы

3. Ответы на контрольные вопросы.

**Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов        | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №6

**Тема:** Мониторинг активности портов (2 часа)

**Цель занятия:**

1. Научится выполнять мониторинг активных портов.

## ИНСТРУКЦИОННАЯ КАРТА

**Материально - техническое оснащение**

**Оборудование:** компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

**Практические задания**

Мониторинг работающей сети предоставляет сетевому администратору информацию для эффективного управления сетью и создания статистических отчетов об использовании сети для других специалистов. Активность подключений, частота появления ошибок и состояние каналов —

вот лишь несколько факторов, которые позволяют сетевому администратору оценить состояние и использование сети. Сбор и просмотр этой информации в течение продолжительного времени позволяют сетевому администратору анализировать и прогнозировать рост проекта, а также помогут обнаружить и заменить неисправную деталь, прежде чем она окончательно выйдет из строя.

Для успешного администрирования сети необходимо знать состояние каждого ее элемента с возможностью изменять параметры его функционирования. Обычно сеть состоит из устройств различных производителей и управлять ею было бы нелегкой задачей, если бы каждое из сетевых устройств понимало только свою систему команд. Поэтому возникла необходимость в создании единого языка управления сетевыми ресурсами, который бы понимали все устройства, и который, в силу этого, использовался бы всеми пакетами управления сетью для взаимодействия с конкретными устройствами.

Подобным языком стал SNMP - Simple Network Management Protocol. Разработанный для систем, ориентированных под операционную систему UNIX, он стал фактически общепринятым стандартом сетевых систем управления и поддерживается подавляющим большинством производителей сетевого оборудования в своих продуктах.

В силу своего названия - Простой Протокол Сетевого Управления - основной задачей при его разработке было добиться максимальной простоты его реализации. В результате возник протокол, включающий минимальный набор команд, однако позволяющий выполнять практически весь спектр задач управления сетевыми устройствами - от получения информации о местонахождении конкретного устройства, до возможности производить его тестирование.

Протокол SNMP дает возможность администраторам управлять узлами, такими как серверы, рабочие станции, маршрутизаторы, коммутаторы и устройства безопасности в сети IP. Он позволяет сетевым администраторам контролировать работу сети, выполнять поиск и разрешение сетевых проблем, а также планировать рост сети.

Сегодня SNMP является самым популярным протоколом управления различными коммерческими, университетскими и исследовательскими объединенными сетями.

Применяют SNMP для:

- Получение информации о состоянии сетевого оборудования
- Локализация неполадок в сети
- Удаленное управление узлами сети
- Сбор статистической информации о состоянии сети
- Возможно применение протокола SNMP в областях, не связанных с сетевыми

технологиями:

- о Проекты по использованию SNMP в системах управления светофорами
- о Производства, имеющие сеть измерительных приборов, разбросанных географически, нуждающиеся в мониторинге состояния и показаний этих приборов

Основные преимущества протокола SNMP:

- простота;
- доступность;
- независимость от производителей.

Основной концепцией протокола является то, что вся необходимая для управления устройством информация хранится на самом устройстве - будь то сервер, модем или маршрутизатор - в так называемой Административной Информационной Базе (MIB - Management Information Base - RFC 1213).

Применяются базы управляющей информации (MIB) для упрощения запоминания адреса объектов устройств, определяемых в цифровом формате. Базы MIB описывают структуру управляемых данных на подсистеме устройства; они используют иерархическое пространство имен, содержащее идентификаторы объектов (OID-ы). Каждый OID состоит из двух частей: текстового имени и SNMP адреса в цифровом виде. Базы MIB являются необязательными и выполняют вспомогательную роль по переводу имени объекта из человеческого формата (словесного) в формат SNMP (цифровой). Так как структура объектов на устройствах разных производителей не совпадает, без базы MIB практически невозможно определить цифровые SNMP адреса нужных объектов.

MIB представляет собой набор переменных, характеризующих состояние объекта управления. Эти переменные могут отражать такие параметры, как количество пакетов, обработанных устройством, состояние его интерфейсов, время функционирования устройства и т.п.

Для того, чтобы проконтролировать работу некоторого устройства сети, необходимо просто получить доступ к его MIB, которая постоянно обновляется самим устройством, и проанализировать значения некоторых переменных.

Модель SNMP состоит из четырех компонентов:

- управляемых узлов;
- станций управления (менеджеров);
- управляющей информации;
- протокола управления.

Агентами в SNMP являются программные модули, которые работают в управляемых устройствах. Агенты собирают информацию об управляемых устройствах, в которых они работают, и делают эту информацию доступной для систем управления сетями (network management systems - NMS) с помощью протокола SNMP.

Управляемое устройство может быть узлом любого типа, находящимся в какой-нибудь сети: это хосты, служебные устройства связи, принтеры, роутеры, мосты и концентраторы, коммутаторы.

При использовании SNMP один или более административных компьютеров (где функционируют программные средства, называемые менеджерами) выполняют отслеживание или управление группой хостов или устройств в компьютерной сети. На каждой управляемой системе есть постоянно запущенная программа, называемая агент, которая через SNMP передает информацию менеджеру.

Менеджеры SNMP обрабатывают данные о конфигурации и функционировании управляемых систем и преобразуют их во внутренний формат, удобный для поддержания протокола SNMP. Протокол также разрешает активные задачи управления, например, изменение и применение новой конфигурации через удаленное изменение этих переменных. Доступные через SNMP переменные организованы в иерархии. Эти иерархии, как и другие метаданные (например, тип и описание переменной), описываются базами управляющей информации (базы MIB, от [англ.](#) Management information base).

Управляемые протоколом SNMP сети состоят из трех ключевых компонентов:

- Управляемое устройство;
- Агент — программное обеспечение, запускаемое на управляемом устройстве, либо на устройстве, подключенном к интерфейсу управления управляемого устройства;
- Система сетевого управления (Network Management System, NMS) — программное обеспечение, взаимодействующее с менеджерами для поддержки комплексной структуры данных, отражающей состояние сети.

Управляемое устройство — элемент сети (оборудование или программное средство), реализующий интерфейс управления (не обязательно SNMP), который разрешает однонаправленный (только для чтения) или двунаправленный доступ к конкретной информации об элементе. Управляемые устройства обмениваются этой информацией с менеджером. Управляемые устройства могут относиться к любому виду устройств: маршрутизаторы, серверы доступа, коммутаторы, мосты, концентраторы, IP-телефоны, IP-видеокамеры, компьютеры-хосты, принтеры и т. п.

Агентом называется программный модуль сетевого управления, располагающийся на управляемом устройстве, либо на устройстве, подключенном к интерфейсу управления управляемого устройства. Агент обладает локальным знанием управляющей информации и переводит эту информацию в специфичную для SNMP форму или из неё (медиация данных).

В состав системы сетевого управления (NMS) входит приложение, отслеживающее и контролирующее управляемые устройства. NMS обеспечивают основную часть обработки данных, необходимых для сетевого управления. В любой управляемой сети может быть одна и более NMS.

Детали протокола

SNMP работает на прикладном уровне TCP/IP (седьмой уровень модели OSI). Агент SNMP получает запросы по UDP-порту 161. Менеджер может посылать запросы с любого доступного порта источника на порт агента. Ответ агента будет отправлен назад на порт источника на менеджере. Менеджер получает уведомления (Traps и InformRequests) по порту 162. Агент может генерировать

уведомления с любого доступного порта. При использовании [TLS](#) или [DTLS](#) запросы получаются по порту 10161, а ловушки отправляются на порт 10162.

В SNMPv1 указано пять основных протокольных единиц обмена (protocol data units — PDU). Еще две PDU, GetBulkRequest и InformRequest, были введены в SNMPv2 и перенесены в SNMPv3.

Все PDU протокола SNMP построены следующим образом:

| IP header (IP-заголовок) | UDP header (UDP-заголовок) | version (версия) | community (пароль) | DU-type (PDU-тип) | request-id (id запроса) | error-status (статус ошибки) | error-index (индекс ошибки) | variable bindings (связанные переменные) |
|--------------------------|----------------------------|------------------|--------------------|-------------------|-------------------------|------------------------------|-----------------------------|------------------------------------------|
|--------------------------|----------------------------|------------------|--------------------|-------------------|-------------------------|------------------------------|-----------------------------|------------------------------------------|

Существует семь протокольных единиц обмена SNMP, из которых два основных:

#### GetRequest

Запрос от менеджера к объекту для получения значения переменной или списка переменных. Требуемые переменные указываются в поле variable bindings (раздел поля values при этом не используется). Получение значений указанной переменной должно быть выполнено агентом как [Атомарная операция](#) (операции, выполняющиеся как единое целое, либо не выполняющиеся вовсе.). Менеджеру будет возвращен Response (ответ) с текущими значениями.

#### SetRequest

Запрос от менеджера к объекту для изменения переменной или списка переменных. Связанные переменные указываются в теле запроса. Изменения всех указанных переменных должны быть выполнены агентом как атомарная операция. Менеджеру будет возвращен Response с (текущими) новыми значениями переменных.

#### Ловушки агента SNMP

NMS периодически проводит опрос агентов SNMP, размещенных на управляемых устройствах, запрашивая данные у устройства с помощью запроса get. С помощью этого процесса приложение для управления сетями может собирать информацию для мониторинга транспортной нагрузки и проверять настройки управляемых устройств. Информация может отображаться через графический интерфейс пользователя в системе NMS. Можно вычислить минимальные, средние и максимальные значения, создать графическое представление данных или установить пороговые значения, при превышении которых будут отправляться соответствующие уведомления. Например, система NMS может контролировать использование центрального процессора маршрутизатора Cisco. Диспетчер SNMP осуществляет периодическую выборку значений и представляет эту информацию в графическом виде, чтобы сетевой администратор мог использовать её для вычисления базовых показателей.

Периодический опрос SNMP имеет свои недостатки. Во-первых, существует задержка между временем обнаружения события и временем отправки соответствующего уведомления (путём опроса) системой NMS. Во-вторых, существует компромисс между частотой опроса и использованием пропускной способности.

Чтобы смягчить воздействие этих недостатков, агенты SNMP могут создавать и отправлять ловушки, сообщая системе NMS о некоторых событиях немедленно. Ловушки — это незапрашиваемые сообщения, предупреждающие диспетчера SNMP о каком-либо условии или событии в сети. Примерами условий ловушек, помимо прочего, являются следующие: неправильная аутентификация пользователей, перезапуски, изменение состояния канала (на активное или неактивное), отслеживание MAC-адресов, закрытие подключения TCP, потеря подключения к соседнему узлу или другие важные события. Уведомления, направленные на ловушки, помогают сократить использование ресурсов сети и агентов, устраняя необходимость в некоторых запросах на опрос SNMP.

#### Версии SNMP

Существует несколько версий SNMP, включая следующие:

SNMPv1 — простой протокол управления сетями, полноценный стандарт Интернета, описанный в документе RFC 1157.

SNMPv2c — описан в серии документов RFC 1901—1908; использует среду администрирования на базе строки сообщества.

SNMPv3 — обеспечивающий взаимодействие протокол на основе стандартов, первоначально определённый в серии документов RFC 2273—2275; обеспечивает защищённый доступ к устройствам с помощью аутентификации и шифрования пакетов в сети. Данная версия протокола включает следующие функции обеспечения безопасности: контроль целостности сообщений для защиты пакетов от искажения при пересылке; аутентификация для подтверждения достоверности источника сообщения и шифрование для предотвращения прочтения содержимого сообщения несанкционированным источником.

В SNMPv1 и SNMPv2c используется модель безопасности на основе сообществ (community). Сообщество диспетчеров, имеющих доступ к базе MIB агента, определяется списком контроля доступа и паролем.

В отличие от SNMPv1, версия SNMPv2c предусматривает механизм массового извлечения записей и более подробное информирование станций управления об ошибках. Механизм массового извлечения получает таблицы и большие объёмы информации, сводя к минимуму затраты времени на двустороннее согласование. Усовершенствованная обработка ошибок в SNMPv2c предусматривает расширенные коды ошибок для различных условий возникновения ошибок. Эти условия обозначаются одним кодом ошибки в SNMPv1. Коды возврата по ошибке в SNMPv2c включают тип ошибки.

Примечание. SNMPv1 и SNMPv2c включают минимальный набор средств обеспечения безопасности. В частности, SNMPv1 и SNMPv2c не обеспечивают ни аутентификацию источника сообщения управления, ни шифрование. Наиболее обновлённое описание версии SNMPv3 содержится в серии документов RFC 3410—3415. В эту версию протокола добавлены методы обеспечения безопасной передачи наиболее важных данных между управляемыми устройствами.

SNMPv3 предусматривает как модели безопасности, так и уровни безопасности. Модель безопасности — это стратегия аутентификации, настроенная для пользователя и группы, в которой данный пользователь находится. Уровень безопасности характеризует допустимую степень безопасности в модели. Сочетание уровня безопасности и модели безопасности определяет, какой механизм безопасности будет использоваться при обработке пакета SNMP. Доступные модели безопасности — SNMPv1, SNMPv2c и SNMPv3.

Сетевой администратор должен настроить агент SNMP для использования версии SNMP, поддерживаемой станцией управления. Поскольку агент может взаимодействовать с несколькими диспетчерами SNMP, можно настраивать программное обеспечение для поддержки связи с помощью SNMPv1, SNMPv2c или SNMPv3.

## 6. Порядок выполнения работы

### 6.1 Собрать схему в соответствии с физической топологией, показанной на рис. 1.

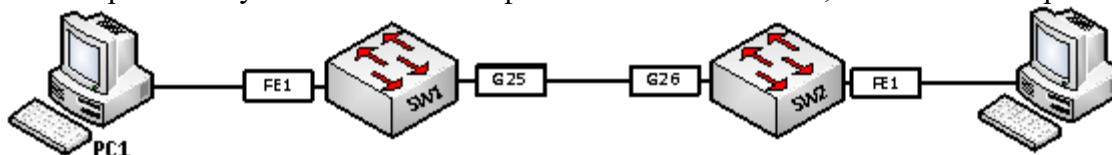


Рисунок 1. (FE1 – Порт Fast Ethernet 1, G25 – Порт Gigabit Ethernet 25).

### 6.2 Настроить адресацию сети в соответствии с диаграммой сетевого уровня, показанной на рис. 2. Коммутаторы предварительно настроены.

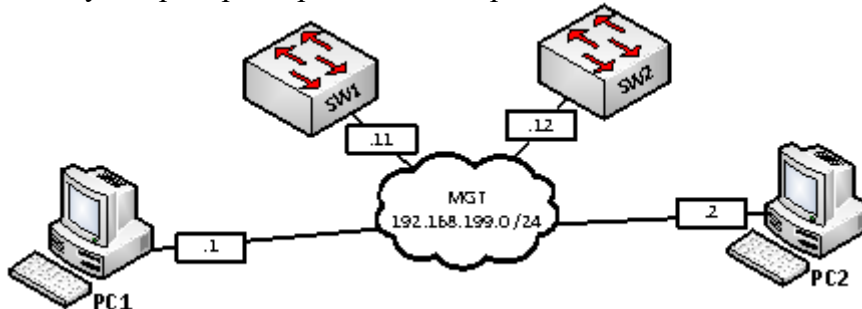


Рисунок 2. Сетевой уровень.

Все устройства в сети должны быть доступны, проверить командой «ping», например с PC1 проверить связь до SW1 командой «ping 192.168.199.11», до SW2 командой «ping 192.168.199.12», до PC2 командой «ping 192.168.199.2», как показано на рис. 3.

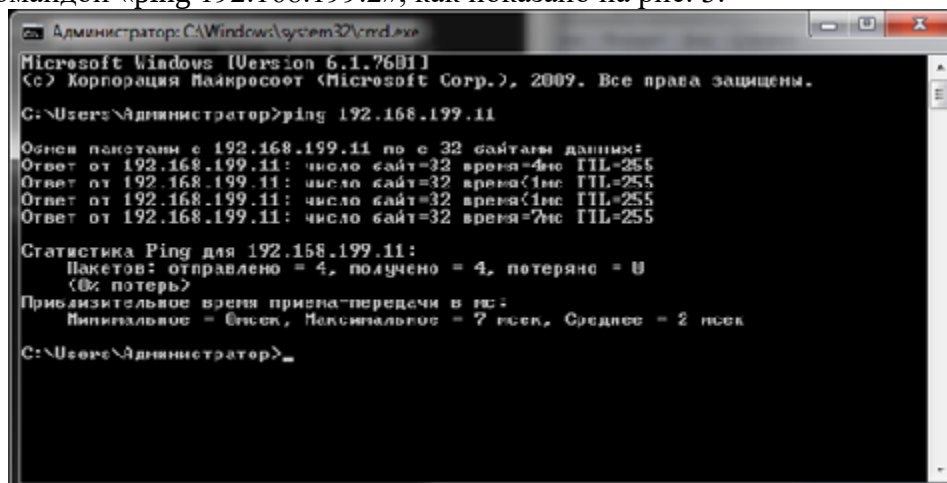
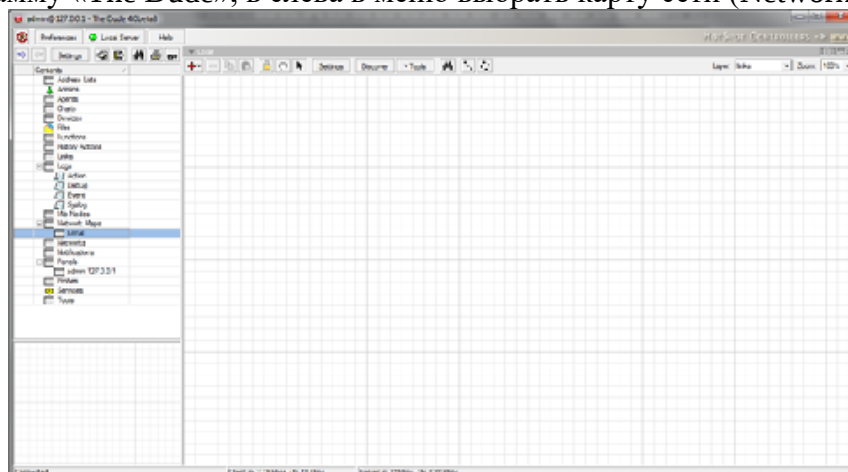
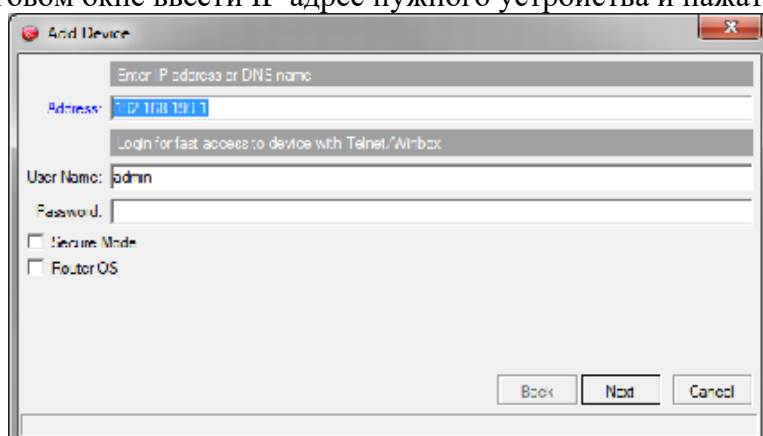


Рисунок 3. Результат проверки связи до SW1.

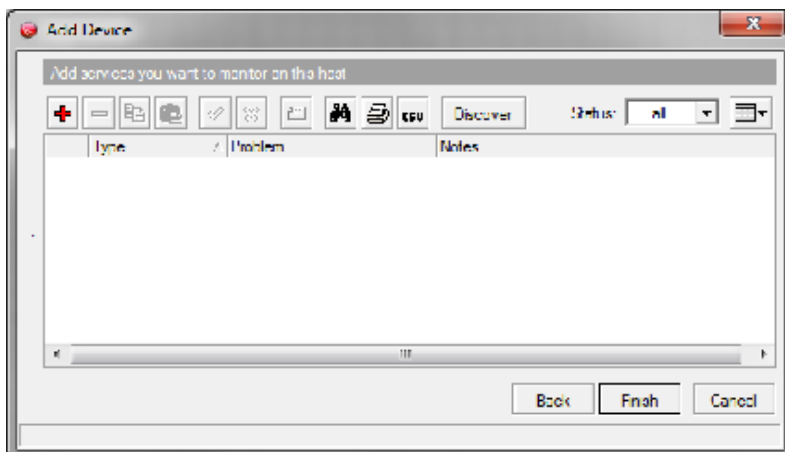
6.3. Произвести настройку SNMP сервера установленного на компьютере, для чего открыть программу «The Dude», в слева в меню выбрать карту сети (Network Maps > Local).



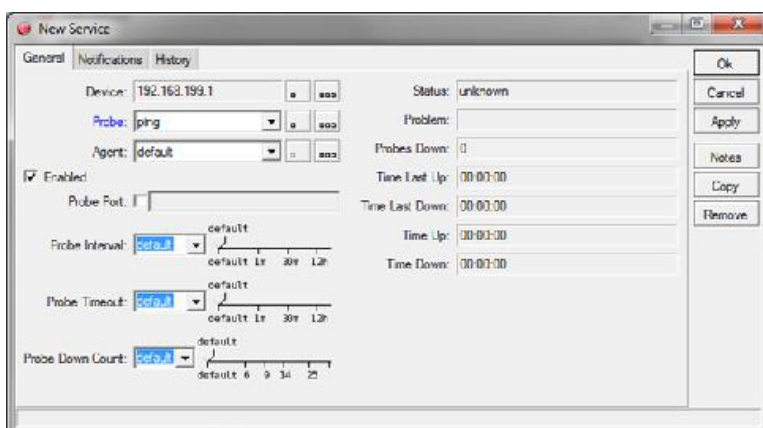
Добавить устройства в соответствии с L1 диаграммой, изображенной на рис. 1. Для этого на панели инструментов нужно нажать красный плюс и выбрать пункт «Device», в открывшемся диалоговом окне ввести IP-адрес нужного устройства и нажать «Next».



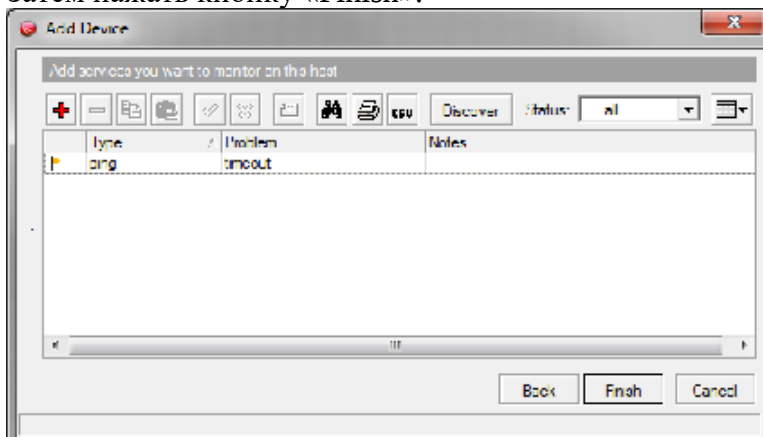
В открывшемся окне выбрать метод проверки работоспособности устройства, для этого нажать красный плюс.



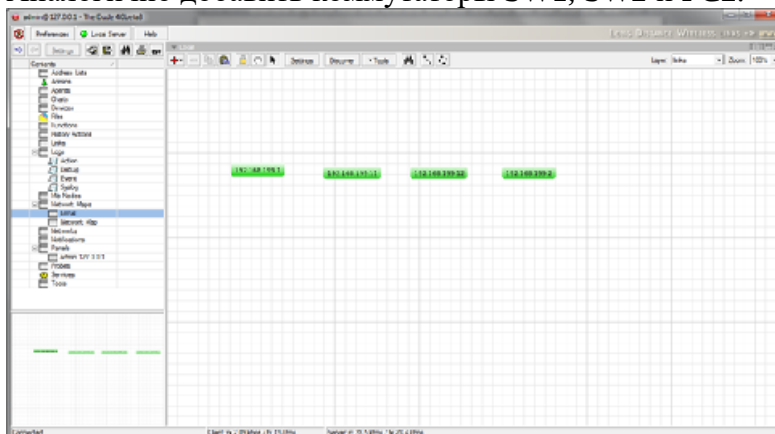
В открывшемся окне из выпадающего списка «Probe» выбрать тип «Ping» и нажать кнопку OK.



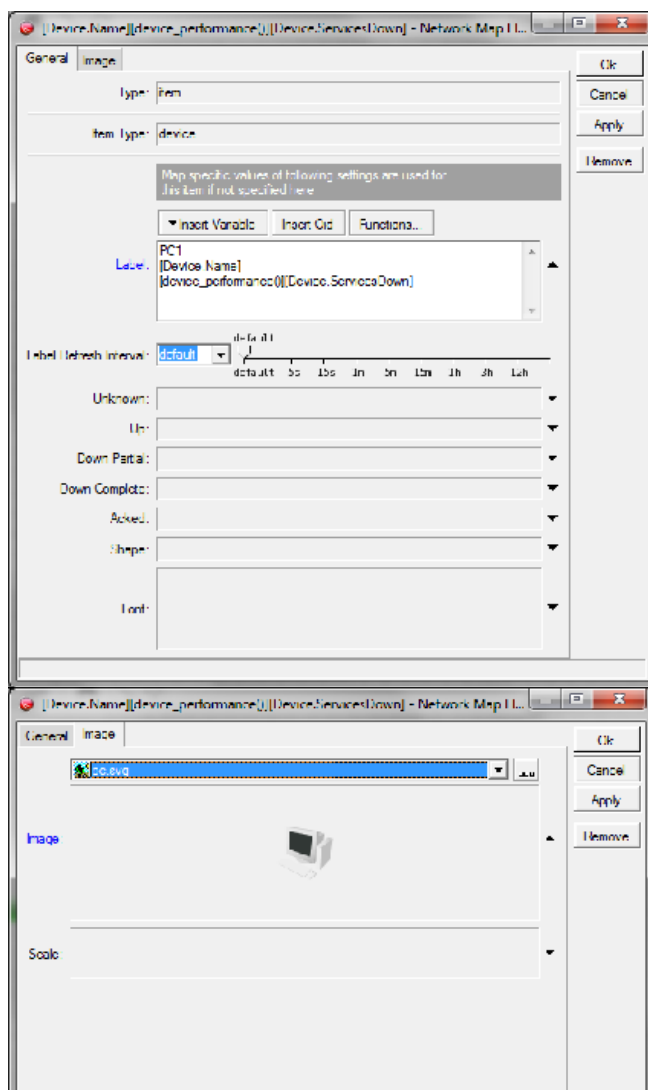
Затем нажать кнопку «Finish».



Аналогично добавить коммутаторы SW1, SW2 и PC2.



Производим настройку компьютеров, нажимаем правой кнопкой на PC1 из контекстного меню выбираем «Arrange», в открывшемся окне во вкладке «General», в поле «Label» добавляем имя PC1, а на вкладке «Image» из выпадающего списка выбрать «pc.svg» и нажать Ок.



Аналогично выполнить для PC2.

Произвести настройку коммутаторов, для этого на коммутаторе нажать правой кнопкой и выбрать пункт «Appearance», в открывшемся окне во вкладке «Image» из выпадающего списка выбрать «switch.svg». Во вкладке «General» в поле «Label», ввести параметры за которыми будет производиться мониторинг по протоколу SNMP. Для примера следующие OID:

А. 1.3.6.1.2.1.1.1.0 – Запрос по протоколу SNMP на наименование модели устройства.

Б. 1.3.6.1.2.1.1.3.0 – Запрос времени работы устройства с момента последнего включения или сбоя.

В. 1.3.6.1.4.1.171.12.1.1.6.1.0 – Запрос загрузки CPU коммутатора в процентах.

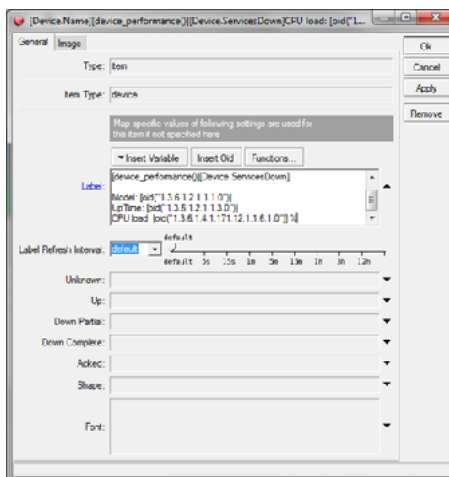
Пример формирования запроса: запрос к конкретному OID осуществляется строкой «[oid("1.3.6.1.4.1.171.12.1.1.6.1.0")]», которая будет выводить значение загрузки CPU одним числом. Для того чтобы данный параметр был интуитивно понятен на создаваемой схеме можно написать следующую строку: «CPU load: [oid("1.3.6.1.4.1.171.12.1.1.6.1.0")] %».

Все запросы могут выглядеть так:

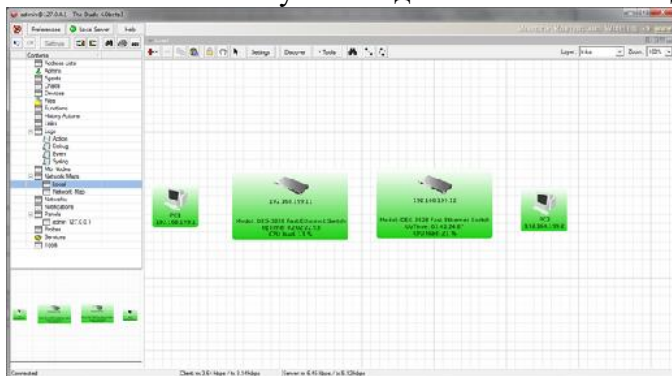
Model: [oid("1.3.6.1.2.1.1.1.0")]

UpTime: [oid("1.3.6.1.2.1.1.3.0")]

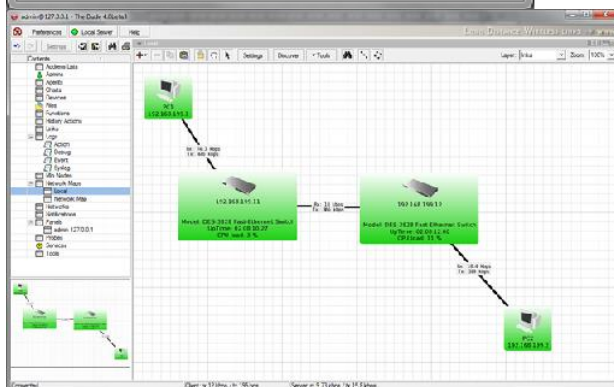
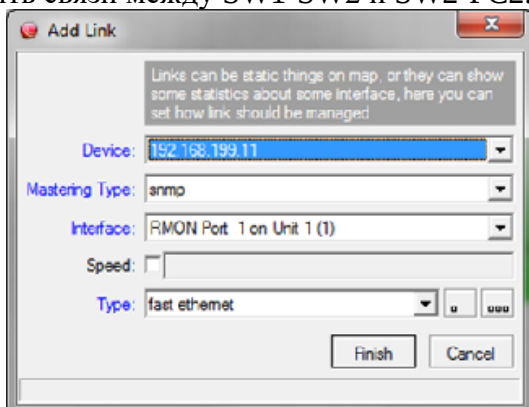
CPU load: [oid("1.3.6.1.4.1.171.12.1.1.6.1.0")] %

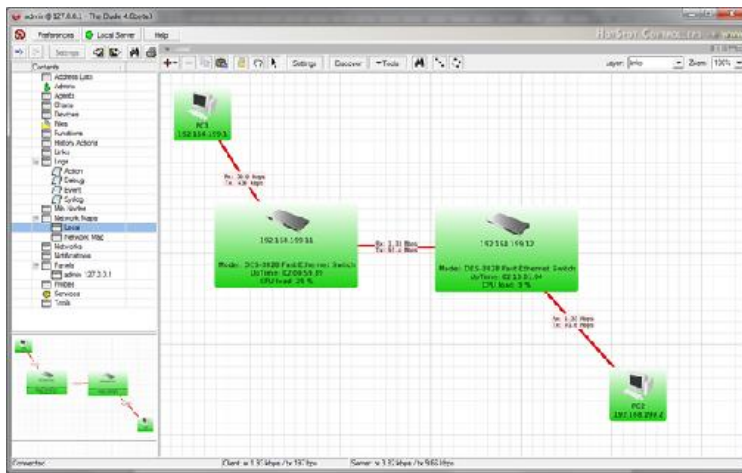


Затем нажать кнопку ОК и сделать аналогичное для коммутатора SW2.



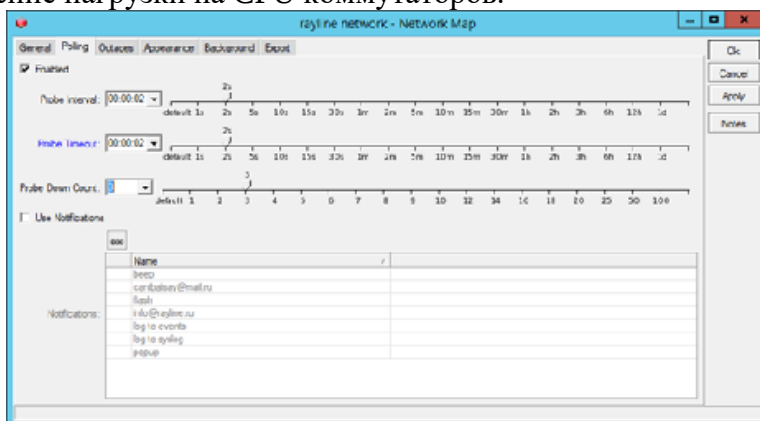
Настройка связей между устройствами. Для добавления связи необходимо нажать на красный плюс, выбрать «Link» и провести между устройствами в открывшемся окне линию, выбрать параметры в соответствии с тем между какими устройствами настраивается связь. Например, если это связь между PC1 и SW, то необходимо выбрать Device – 192.168.199.11, Mastering Type – snmp, Interface – RMON Port 1 on Unit 1 (1), Type – Fast Ethernet, затем нажать кнопку «Finish», аналогично добавить связи между SW1-SW2 и SW2-PC2.





#### 6.4 Проверка выполненной лабораторной работы.

1. За всеми устройствами, включенными в сеть, должен выполняться мониторинг, об этом должен свидетельствовать зеленый цвет устройств на карте сети.
2. Мониторинг вышедшего из строя оборудования – для проверки необходимо отключать кабель в разных участках сети и наблюдайте за картой сети, устройства до которых была потеряна связь будут отмечены красным, при возобновлении связи устройства станут опять зелеными. Если устройства не меняют свой цвет, то необходимо уменьшить время опроса оборудования, для этого зайдите в настройки (settings), затем на вкладке Polling выставите время опроса 2 секунды, как показано на рисунке. И затем повторно выполните проверку пункта
3. Между всеми устройствами должен осуществляться мониторинг пропускной способности в реальном времени – об этом свидетельствуют цифры в килобит/с или мегабит/с на связях между устройствами. Для проверки работы запустите тест проверки скорости с компьютеров и наблюдайте за показаниями реальной скорости в системе мониторинга. При проверке скорости между PC1 и PC2 можно видеть изменение скорости в реальном времени, а также наблюдать изменение нагрузки на CPU коммутаторов.



4. Проверка времени работы с момента последнего запуска или сбоя (Uptime)– показывает сколько времени назад было включено устройство. Для проверки, запомните время Uptime на одном из устройств, затем перезагрузите устройство (отключите и подключите питание к коммутатору), после загрузки устройства убедитесь, что счетчик Uptime обнулится и начался новый отсчет времени.

#### Отчет

- 1 Схема сети.
- 2 Схема сетевого уровня
- 3 Краткое пояснение настройки адресации сети в соответствии с диаграммой сетевого уровня.
- 4 Карта сети.
- 5 Выводы по мониторингу.

#### Критерии оценивания отчета:

|                                                 |                               |                         |                         |
|-------------------------------------------------|-------------------------------|-------------------------|-------------------------|
| Своевременность и успешность составления отчета | Качество усвоенного материала | Творческая составляющая | Перевод баллов в оценку |
|-------------------------------------------------|-------------------------------|-------------------------|-------------------------|

|                                             |                                                  |                                                           |                                                                                                     |
|---------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа- 10 баллов    | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов     | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие №7

**Тема:** Блокирование портов (2 часа)

### Цель занятия:

1. Научится блокировать вредоносные порты.

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows

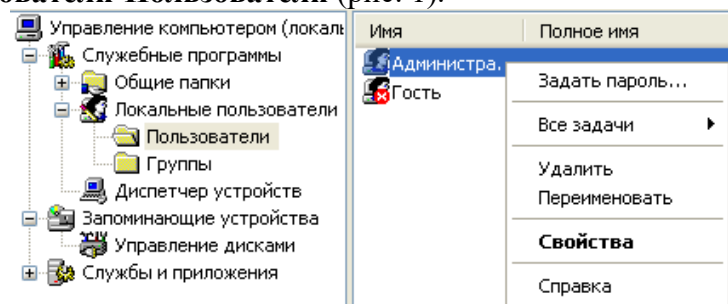
**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

Политику безопасности можно сравнить с пограничником, охраняющим границу страны. Ниже мы рассмотрим два способа улучшения безопасности работы нашей виртуальной сети за два приема.

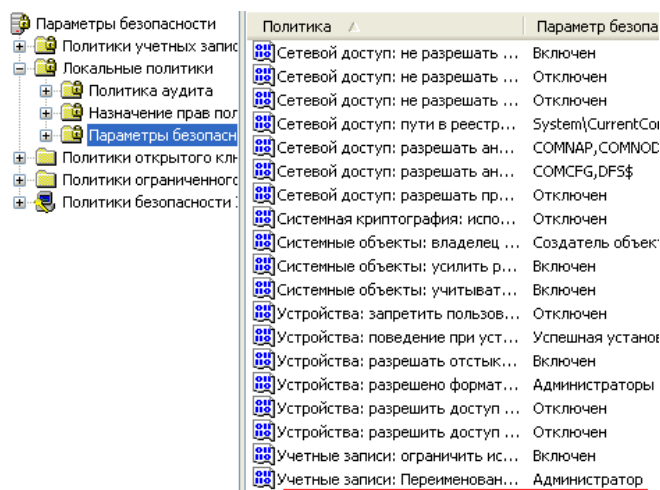
**Шаг 1.** Меняем учетную запись администратора (Пользователь Администратор с пустым паролем — это уязвимость)

Часто при установке Windows пароль администратора пустой и этим может воспользоваться злоумышленник. Иначе говоря, при установке Windows XP в автоматическом режиме с настройками по умолчанию мы имеем пользователя **Администратор** с пустым паролем и любой **User** может войти в такой ПК с правами администратора. Чтобы решить проблему выполним команду **Мой компьютер-Панель управления-Администрирование-Управление компьютером-Локальные пользователи-Пользователи** (рис. 1).

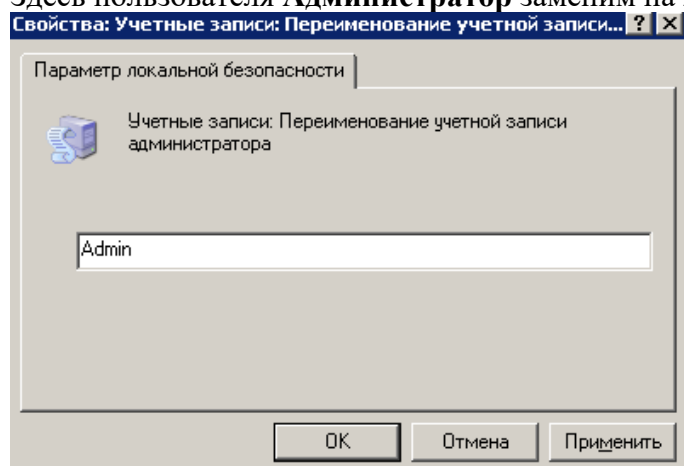


**Рис. 1.** Окно Управление компьютером

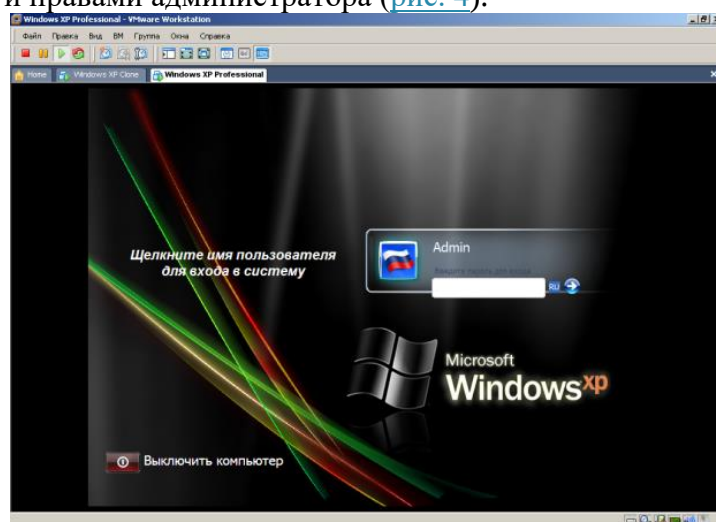
Здесь по щелчку правой кнопкой мыши на **Администраторы** зададим администратору пароль, например, 12345. Это плохой пароль, но лучше, чем ничего. Теперь в окне **Администрирование** зайдем в **Локальную политику безопасности**. Далее идем по веткам дерева: **Локальные политики-Параметры безопасности-Учетные записи: Переименование учетной записи Администратор** (рис..2).



**Рис. 2.** Находим в системном реестре запись Переименование учетной записи Администратор. Здесь пользователя **Администратор** заменим на **Admin** (рис. 3).



**Рис. 3.** Пользователю Администратор присваиваем новое имя. Перезагружаем ОС. После наших действий у нас получилась учетная запись Admin с паролем 12345 и правами администратора (рис. 4).



**Рис. 4.** Окно входа в ОС Windows XP

Все, теперь мы имеем пользователя **Администратор** с паролем, одна из уязвимостей системы устранена.

#### Примечание

Операцию по изменению имени пользователя и заданию пароля мы также могли бы выполнить без использования системного реестра, используя окно **Учетные записи пользователей**, что гораздо проще (рис. 5).

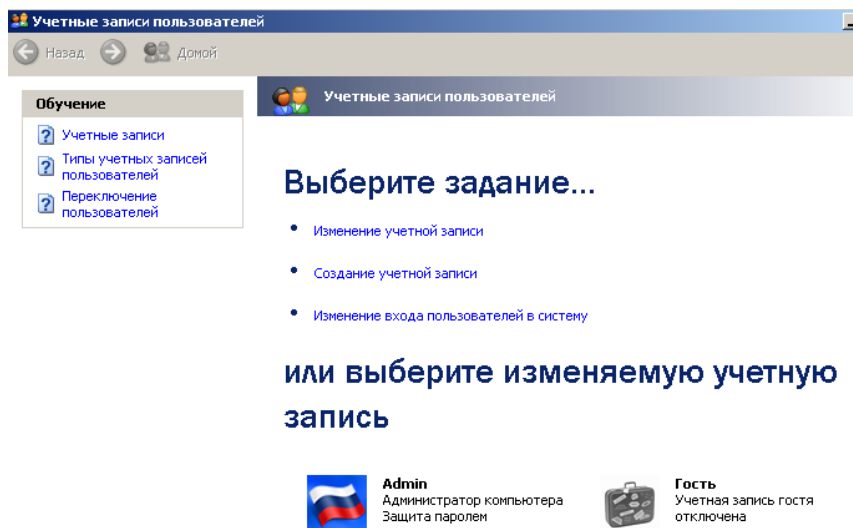


Рис. 5. Окно Учетные записи пользователей

### Примечание

Учетная запись Гость позволяет входить в ПК и работать на нем (например, в Интернет) без использования специально созданной учетной записи. Запись Гость не требует ввода пароля и по умолчанию заблокирована. Гость не может устанавливать или удалять программы. Эту учетную запись можно отключить, но нельзя удалить.

### Шаг 2. Делаем окно приветствия пустым (убираем уязвимость 2)

У нас окно входа в систему содержит подсказку Admin, давайте ее уберем, сделав окно пустым. Для начала в окне **Учетные записи пользователей** жмем на кнопку **Изменение входа пользователей в систему** и уберем флажок **Использовать страницу приветствия** (рис. 6 и рис. 7).

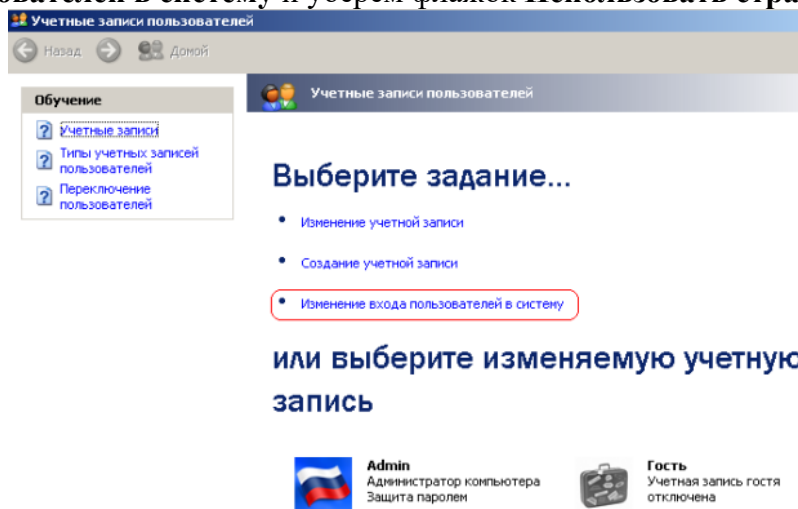


Рис. 6. Окно Учетные записи пользователей

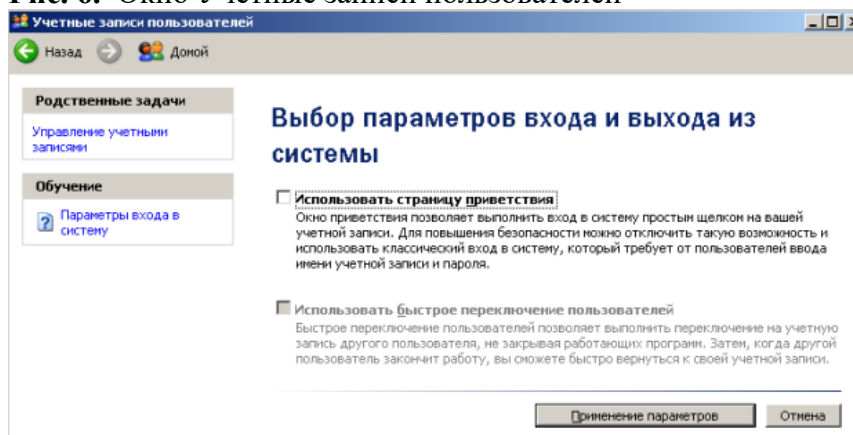


Рис. 7. Убираем флажок Использовать страницу приветствия

Но, это только половина дела. Теперь повысим безопасность сети еще на одну условную ступень, сделав оба поля окна приветствия пустыми (рис. 8).

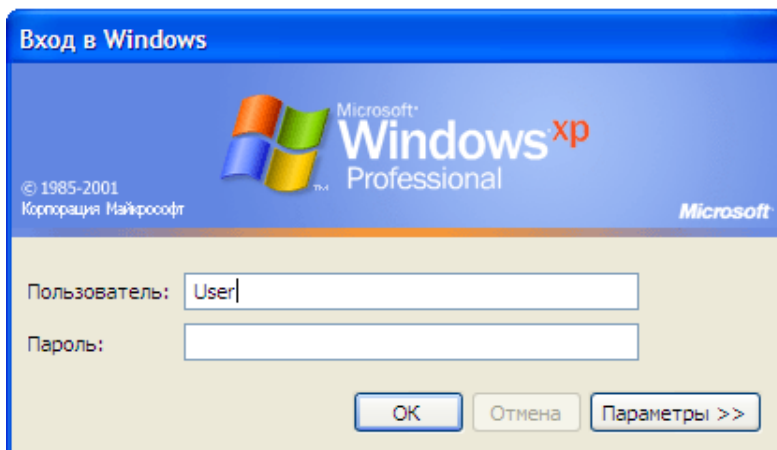


Рис. 8. Обе строки данного окна сделаем пустыми

Выполним команду **Панель управления-Администрирование – Локальные политики безопасности- Локальные политики-Параметры безопасности—Интерактивный вход: не отображать последнего имени пользователя**. Эту запись необходимо включить (рис. 9).

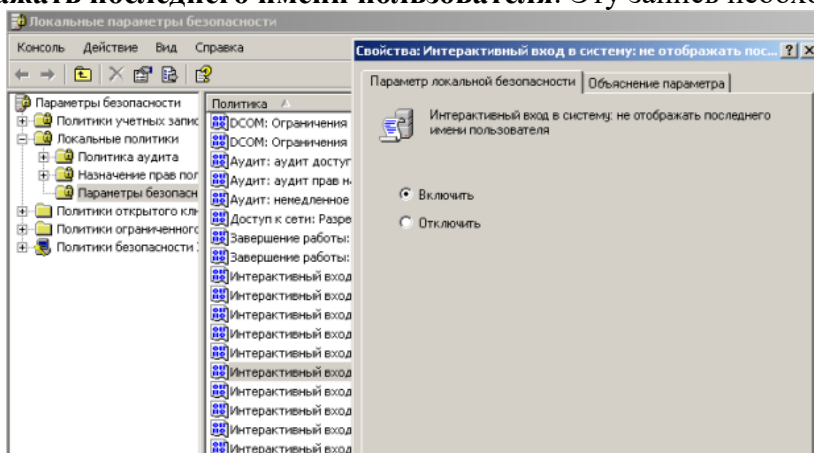


Рис. 9. Активируем переключатель Включить

Теперь после завершения сеанса пользователь должен угадать не только пароль, но и имя пользователя (рис. 10).

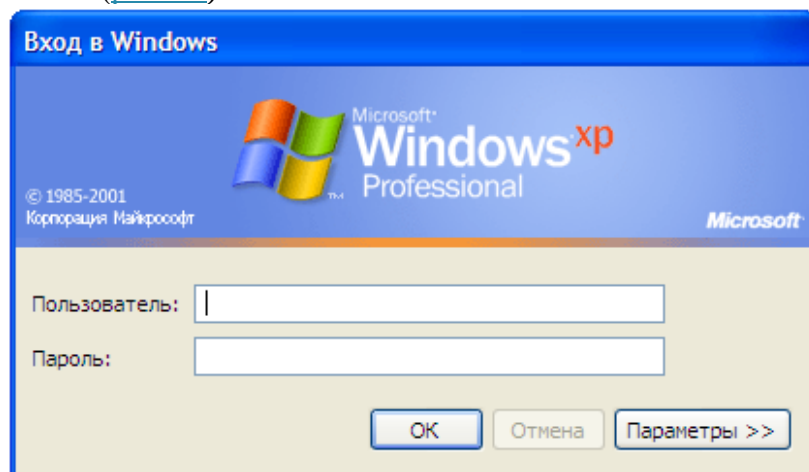


Рис. 10. Обе строки окна приветствия пусты

### Выявление сетевых уязвимостей сканированием портов ПК

Злоумышленники используют сканирование портов ПК для того, чтобы воспользоваться ресурсами чужого ПК в Сети. При этом необходимо указать IP адрес ПК и открытый port, к примеру, 195.34.34.30:23. После этого происходит соединение с удаленным ПК с некоторой вероятностью входа в этот ПК.

- TCP/IP port — это адрес определенного сервиса (программы), запущенного на данном компьютере в Internet. Каждый открытый порт — потенциальная лазейка для взломщиков сетей и ПК. Например, SMTP (отправка почты) — 25 порт, WWW — 80 порт, FTP — 21 порт.

- Хакеры сканируют порты для того, чтобы найти дырку (баг) в операционной системе. Пример ошибки, если администратор или пользователь ПК открыл полный доступ к сетевым ресурсам для всех или оставил пустой пароль на вход к компьютер.

Одна из функций администратора сети (сисадмина) — выявить недостатки в функционировании сети и устранить их. Для этого нужно просканировать сеть и закрыть (блокировать) все необязательные (открытые без необходимости) сетевые порты. Ниже, для примера, представлены службы TCP/IP, которые можно отключить:

- finger — получение информации о пользователях
- talk — возможность обмена данными по сети между пользователями
- bootp — предоставление клиентам информации о сети
- sysstat — получение информации о системе
- netstat — получение информации о сети, такой как текущие соединения
- rusersd — получение информации о пользователях, зарегистрированных в данный момент

### Просмотр активных подключений утилитой Netstat

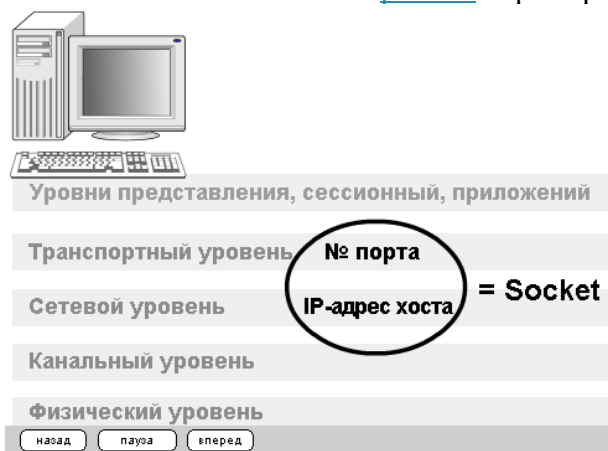
Команда **netstat** обладает набором ключей для отображения портов, находящихся в активном и/или пассивном состоянии. С ее помощью можно получить список серверных приложений, работающих на данном компьютере. Большинство серверов находится в режиме **LISTEN** — ожидание запроса на соединение. Состояние **CLOSE\_WAIT** означает, что соединение разорвано. **TIME\_WAIT** — соединение ожидает разрыва. Если соединение находится в состоянии **SYN\_SENT**, то это означает наличие процесса, который пытается установить соединение с сервером. **ESTABLISHED** — соединения установлены, т.е. сетевые службы работают (используются).

Итак, команда **netstat** показывает содержимое различных структур данных, связанных с сетью, в различных форматах в зависимости от указанных опций. Для сокетов (программных интерфейсов) TCP допустимы следующие значения состояния

- CLOSED — Закрыт. Сокет не используется.
- LISTEN — Ожидает входящих соединений.
- SYN\_SENT — Активно пытается установить соединение.
- SYN\_RECEIVED — Идет начальная синхронизация соединения.
- ESTABLISHED — Соединение установлено.
- CLOSE\_WAIT — Удаленная сторона отключилась; ожидание закрытия сокета.
- FIN\_WAIT\_1 — Сокет закрыт; отключение соединения.
- CLOSING — Сокет закрыт, затем удаленная сторона отключилась; ожидание подтверждения.
- LAST\_ACK — Удаленная сторона отключилась, затем сокет закрыт; ожидание подтверждения.
- FIN\_WAIT\_2 — Сокет закрыт; ожидание отключения удаленной стороны.
- TIME\_WAIT — Сокет закрыт, но ожидает пакеты, ещё находящиеся в сети для обработки

### Примечание

Что такое «сокет» поясняет [рис. 11](#). Пример сокета – 194.86.6..54:21



**Рис. 11.** Сокет это № порта + IP адрес хоста

### Практический пример. Обнаружение открытых на ПК портов утилитой Netstat

Для выполнения практического задания на компьютере необходимо выполнить команду **Пуск-Выполнить**. Откроется окно **Запуск программы**, в нем введите команду **cmd** ([рис. 12](#)).

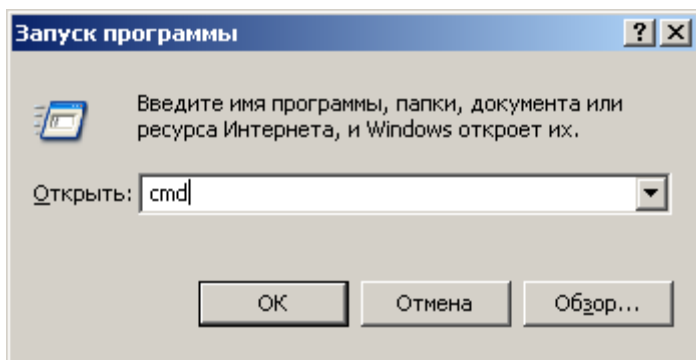


Рис. 12. Окно Запуск программы

Чтобы вывести все активные подключения TCP и прослушиваемые компьютером порты TCP/UDP введите команду **netstat** (рис. 13). Мы видим Локального адреса (это ваш ПК) прослушиваются 6 портов. Они нужны для поддержки сети. На двух портах мы видим режим **ESTABLISHED** — соединения установлены, т.е. сетевые службы работают (используются). Четыре порта используются в режиме **TIME\_WAIT** — соединение ожидает разрыва.

```

Активные подключения
Имя      Локальный адрес      Внешний адрес      Состояние
TCP      D:3086               localhost:3087      ESTABLISHED
TCP      D:3087               localhost:3086      ESTABLISHED
TCP      D:3414               localhost:1110      TIME_WAIT
TCP      D:3416               localhost:1110      TIME_WAIT
TCP      D:3415               OCSP.AMS1.VERISIGN.COM:http TIME_WAIT
TCP      D:3417               OCSP.AMS1.VERISIGN.COM:http TIME_WAIT
D:\Documents and Settings\110>

```

Рис. 13. Список активных подключений на тестируемом ПК

Запустите на вашем ПК Интернет и зайдите, например на **www.yandex.ru**. Снова выполните команду **netstat** (рис. 14). Как видим, добавилось несколько новых активных портов с их различными состояниями.

```

D:\Documents and Settings\110>netstat
Активные подключения
Имя      Локальный адрес      Внешний адрес      Состояние
TCP      D:1110               localhost:3433      TIME_WAIT
TCP      D:1110               localhost:3436      TIME_WAIT
TCP      D:1110               localhost:3441      TIME_WAIT
TCP      D:1110               localhost:3442      TIME_WAIT
TCP      D:1110               localhost:3443      TIME_WAIT
TCP      D:1110               localhost:3448      ESTABLISHED
TCP      D:1110               localhost:3452      TIME_WAIT
TCP      D:1110               localhost:3454      ESTABLISHED
TCP      D:1110               localhost:3456      TIME_WAIT
TCP      D:3430               localhost:3431      ESTABLISHED
TCP      D:3431               localhost:3430      ESTABLISHED
TCP      D:3432               localhost:1110      TIME_WAIT
TCP      D:3438               localhost:1110      TIME_WAIT
TCP      D:3440               localhost:1110      TIME_WAIT
TCP      D:3448               localhost:1110      ESTABLISHED
TCP      D:3450               localhost:1110      TIME_WAIT
TCP      D:3454               localhost:1110      ESTABLISHED
TCP      D:3458               localhost:1110      TIME_WAIT
TCP      D:3460               localhost:1110      TIME_WAIT
TCP      D:3461               localhost:1110      TIME_WAIT
TCP      D:3462               localhost:1110      TIME_WAIT
TCP      D:3434               addons-star.zlb.phx.mozilla.net:https TIME_WAIT
TCP      D:3445               static.yandex.net:http TIME_WAIT
TCP      D:3449               mc.yandex.ru:http   ESTABLISHED
TCP      D:3455               suggest.yandex.net:http ESTABLISHED
TCP      D:3463               suggest.yandex.net:http TIME_WAIT
TCP      D:3464               www.yandex.ru:http   TIME_WAIT
TCP      D:3465               yabs.yandex.ru:http  TIME_WAIT

```

Рис. 14. Активные подключения при работе ПК в Интернет

Команда **netstat** имеет следующие опции – табл. 1.

Таблица 1. Ключи для команды netstat

| Опция (ключ) | Назначение                                                                                                                                                                                    |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -a           | Показывать состояние всех сокетов; обычно сокет, используемый серверными процессами, не показывается.                                                                                         |
| -A           | Показывать адреса любых управляющих блоков протокола, связанных с сокетами; используется для отладки.                                                                                         |
| -i           | Показывать состояние автоматически сконфигурированных (auto-configured) интерфейсов. Интерфейсы, статически сконфигурированные в системе, но не найденные во время загрузки, не показываются. |
| -n           | Показывать сетевые адреса как числа. netstat обычно показывает адреса как символы. Эту опцию можно использовать с любым форматом показа.                                                      |

|                         |                                                                                                                                                                                                                                                                                                                                                       |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -t                      | Показать таблицы маршрутизации. При использовании с опцией -s, показывает статистику маршрутизации.                                                                                                                                                                                                                                                   |
| -s                      | Показать статистическую информацию по протоколам. При использовании с опцией -t, показывает статистику маршрутизации.                                                                                                                                                                                                                                 |
| -f<br>семейство_адресов | Ограничить показ статистики или адресов управляющих блоков только указанным семейством_адресов, в качестве которого можно указывать: <b>inet</b> Для семейства адресов <b>AF_INET</b> , или <b>unix</b> Для семейства адресов <b>AF_UNIX</b> .                                                                                                        |
| -I интерфейс            | Выделить информацию об указанном интерфейсе в отдельный столбец; по умолчанию (для третьей формы команды) используется интерфейс с наибольшим объемом переданной информации с момента последней перезагрузки системы. В качестве интерфейса можно указывать любой из интерфейсов, перечисленных в файле конфигурации системы, например, emd1 или lo0. |
| -p                      | Отобразить идентификатор/название процесса создавшего сокет (-p, —programs display PID/Program name for sockets)                                                                                                                                                                                                                                      |

### Программа NetStat Agent

Представьте ситуацию: ваше Интернет-соединение стало работать медленно, компьютер постоянно что-то качает из Сети. Вам поможет программа NetStat Agent. С ее помощью вы сможете найти причину проблемы и заблокировать ее. Иначе говоря, [NetStat Agent](#) — полезный набор инструментов для мониторинга Интернет соединений и диагностики сети. Программа позволяет отслеживать TCP и UDP соединения на ПК, закрывать нежелательные соединения, завершать процессы, обновлять и освобождать DHCP настройки адаптера, просматривать сетевую статистику для адаптеров и TCP/IP протоколов, а также строить графики для команд **Ping** и **TraceRoute** (рис. 15).

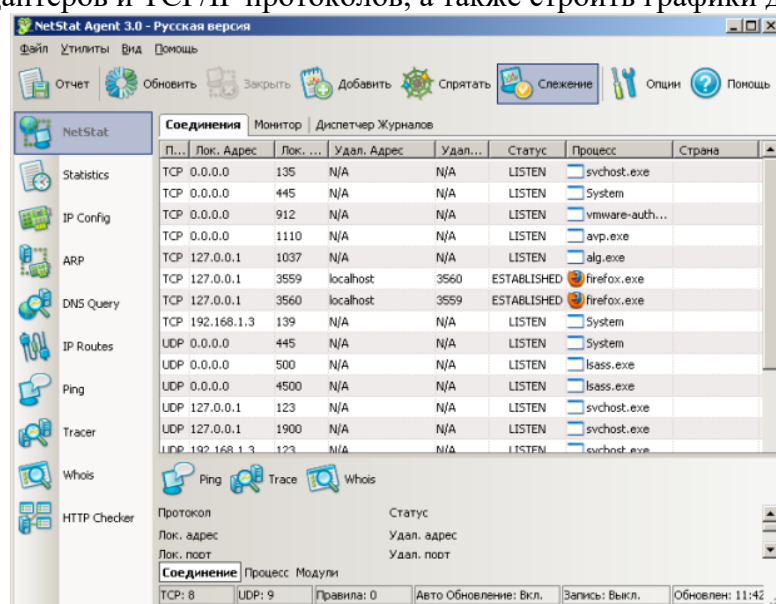


Рис. 15. Главное окно программы NetStat Agent

В состав программы NetStat Agent вошли следующие утилиты:

- **NetStat** — отслеживает TCP и UDP соединения ПК (при этом отображается географическое местоположение удаленного сервера и имя хоста).
- **IPConfig** — отображает свойства сетевых адаптеров и конфигурацию сети.
- **Ping** — позволяет проверить доступность хоста в сети.
- **TraceRoute** — определяет маршрут между вашим компьютером и конечным хостом, сообщая все IP-адреса маршрутизаторов.
- **DNS Query** — подключается к DNS серверу и находит всю информацию о домене (IP адрес сервера, MX-записи (Mail Exchange) и др.).
- **Route** — отображает и позволяет изменять IP маршруты на ПК.
- **ARP** — отслеживает ARP изменения в локальной таблице.
- **Whois** — позволяет получить всю доступную информацию об IP-адресе или домене.
- **HTTP Checker** — помогает проверить, доступны ли Ваши веб-сайты.

- **Statistics** — показывает статистику сетевых интерфейсов и TCP/IP протоколов.

### Сканер портов Nmap (Zenmap)

**Nmap** — популярный сканер портов, который обследует сеть и проводит аудит защиты. Использовался в фильме «Матрица: Перезагрузка» при взломе компьютера. Наша задача не взломать, а защитить ПК, поскольку одно и то же оружие можно использовать как для защиты, так и для нападения. Иначе говоря, сканером портов **nmap** можно определить открытые порты компьютера, а для безопасности сети пользователям рекомендуется закрыть доступ к этим портам с помощью брандмауэра (рис. 16).

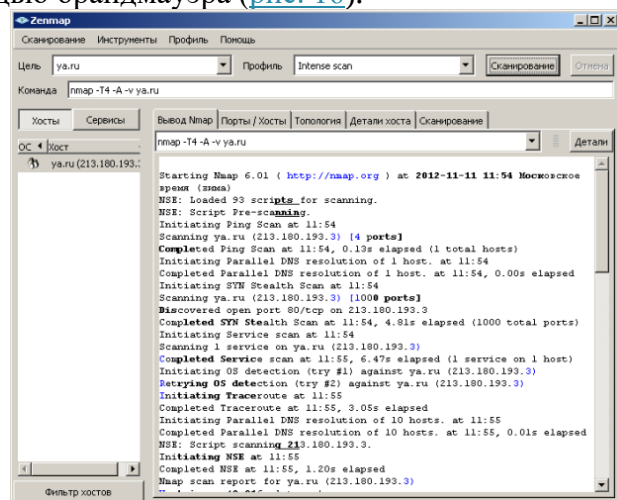


Рис. 16. Интерфейс программы Nmap

Обычно для того, чтобы просканировать все порты какого-либо компьютера в сети вводится команда **nmap -p1-65535 IP-адрес\_компьютера** или **nmap -sV IP-адрес компьютера**, а для сканирования сайта — командой **nmap -sS -sV -O -P0 адрес сайта**.

### Монитор портов TCPView

**TCPView** — показывает все процессы, использующие Интернет-соединения. Запустив **TCPView**, можно узнать, какой порт открыт и какое приложение его использует, а при необходимости и немедленно разорвать соединение — рис. 17.

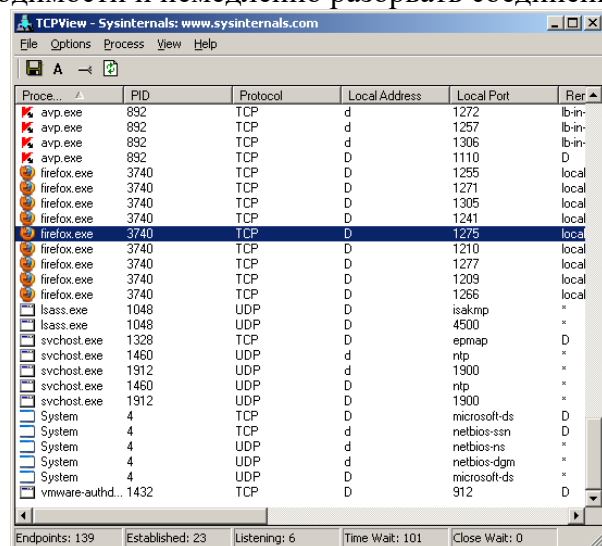


Рис. 17. Главное окно программы TCPView

Просмотрите активные сетевые подключения локального ПК с помощью монитора портов **tcpview**. Определите потенциально возможные угрозы (какие порты открыты, и какие приложения их используют). При необходимости можно закрыть установленное приложением TCP-соединение или процесс правой кнопкой мыши.

### Контрольные вопросы:

1. Какие уязвимости ОС Windows были устранены в данной лабораторной работе и какими путями?
2. Для чего используется утилита Netstat?

3. Перечислите какие утилиты вошли в состав программы NetStat Agent? Для чего используется каждая из утилит?
4. Для чего используется программа Nmap? TCPView?

### Отчет

1. Тема и цель работы
2. Экранные копии выполнения работы
3. Ответы на контрольные вопросы.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа - 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Тема 2.2 Сертификация информационных систем

### Лабораторное занятие № 8

**Тема:** Проверка наличия и сроков действия сертификатов (2 часа)

### Цель занятия:

1. Научится выполнять проверку наличия и срока действия сертификата безопасности.

### ИНСТРУКЦИОННАЯ КАРТА

### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

### Практические задания

Сертификат соответствия - документ, удостоверяющий соответствие объекта требованиям технических регламентов, положениям стандартов или условиям договоров.

Основанием для выдачи сертификата соответствия ГОСТ Р является протокол, составленный по результатам испытаний в аккредитованной Ростехрегулированием испытательной лаборатории. В некоторых регламентированных случаях для выдачи сертификата соответствия ГОСТ Р требуется предоставление в орган по сертификации дополнительных документов: санитарно-эпидемиологическое заключение, сертификат пожарной безопасности, технические условия и др. В этом случае названия и номера этих документов вносятся в сертификат соответствия ГОСТ Р.

Госстандартом РФ (Ростехрегулирование) принята «Номенклатура продукции, в отношении которой законодательными актами РФ предусмотрена обязательная сертификация». Номенклатура представляет собой перечень продукции, упорядоченный по общероссийскому классификатору ОК 005-93, и перечень соответствующих нормативных документов ГОСТ, СанПиН и др. Реализация потребителю продукции, входящей в состав этой номенклатуры, а также таможенная очистка невозможна без сертификата соответствия ГОСТ Р.

Федеральная таможенная служба совместно с Ростехрегулированием утвердили "Список товаров, для которых требуется подтверждение проведения обязательной сертификации при выпуске

на таможенную территорию Российской Федерации". В Списке продукция упорядочена в соответствии с Товарной номенклатурой внешнеэкономической деятельности (ТН ВЭД). В тех случаях, если сертификация продукции является обязательной, выдаваемый обязательный сертификат соответствия ГОСТ Р печатается на желтом бланке. Существуют 3 типа обязательных сертификатов соответствия ГОСТ Р:

1. Обязательный сертификат соответствия ГОСТ Р на поставку (контракт) партии товаров (единичная поставка). В этом случае импортер - российская фирма, получает обязательный сертификат соответствия ГОСТ Р на единичную поставку продукции. Обязательный сертификат соответствия ГОСТ Р выдается этой фирме-импортеру органом по сертификации в соответствие с контрактом, инвойсом и другими документами. Для другой поставки или другой фирмы-импортера этот сертификат уже не действует;

2. Обязательный сертификат соответствия ГОСТ Р для серийного производства сроком на 1 год. В этом случае не нужен конкретный импортер в России, Обязательный сертификат соответствия ГОСТ Р выдается фирме-производителю (изготовителю продукции) сроком на 1 год. Это означает, что фирма-производитель может в течение всего года поставлять свою продукцию любым фирмам и в любые регионы России;

3. Обязательный сертификат соответствия ГОСТ Р для серийного производства сроком на 3 года. В этом случае также не нужен конкретный импортер в России, Обязательный сертификат соответствия ГОСТ Р выдается фирме-производителю (изготовителю продукции) на 3 года. Обязательным условием получения этого сертификата является инспекционная поездка российских экспертов и проверка условий производства продукции на месте.

Продукция, указанная в обязательном сертификате соответствия ГОСТ Р, может доставляться в Россию в течение 3 лет. В соответствии с российским законодательством на фирмах, имеющих обязательный сертификат ГОСТ Р на серийное производство продукции в течение 3 лет, со стороны органа по сертификации должен осуществляться ежегодный инспекционный контроль.

Сертификат соответствия ГОСТ Р добровольной сертификации. Добровольная сертификация проводится по инициативе заявителей (изготовителей, продавцов, исполнителей) в целях подтверждения соответствия продукции требованиям стандартов, технических условий, рецептур и других документов, определяемых заявителем, и не может заменить обе формы обязательного подтверждения соответствия - обязательную сертификацию и декларирование соответствия. Добровольный сертификат соответствия ГОСТ Р, как и обязательный оформляется сроком на 1 или 3 года на фирму производителя (изготовителя продукции). Если у Вас нет необходимости в получении добровольного сертификата соответствия ГОСТ Р, Вы можете оформить Отказное письмо для торговли или для таможни. Зачем нужна добровольная сертификация? Сертифицированные товары или услуги Вашей компании покажут покупателям и партнерам, что Вы уверены в качестве, надежности и эффективности своей продукции. Добровольная сертификация повышает конкурентоспособность продукции и услуг, ускоряет процесс товарооборота. Таким образом, добровольная сертификация выступает как эффективный рыночный инструмент, в котором заинтересован как потребитель, так и изготовитель. Если сертификация продукции является добровольной - добровольный сертификат соответствия ГОСТ Р печатается на синем бланке.

Любой сертификат соответствия ГОСТ Р обязательно содержит следующие сведения:

1 регистрационный номер сертификата.

В структуре регистрационного номера можно выделить пять элементов:

РОСС XX XXXX X XXXXXX

(1) (2) (3) (4) (5)

1 – РОСС - знак регистрации в реестре Госстандарта;

2 – код страны расположения организации-изготовителя данной продукции (оказывающей данную услугу) в виде буквенного кода из двух символов (по ОК 025-95) латинского [алфавита](#) (например, Россия – RU, Индия – IN, Нидерланды – NL);

3 – код органа по сертификации (используются четыре последних знака регистрационного номера органа);

4 (одна или две буквы) – код типа объекта сертификации.

“А” – партия (единичное изделие), сертифицированная на соответствие обязательным требованиям;

“В” – серийно выпускаемая продукция, сертифицированная на соответствие обязательным требованиям;

5 – номер объекта регистрации (пятиразрядный цифровой код).

## 2 Срок действия сертификата.

Даты записываются следующим образом: число и месяц – двумя арабскими цифрами, разделенными точкой, год – четырьмя арабскими цифрами. Первую дату проставляют по дате регистрации сертификата в государственном реестре. Дата окончания срока действия сертификата, выданного на партию товара (смотри предыдущий пункт), не указывается.

## 3 Регистрационный номер органа по сертификации.

Приводится по государственному реестру, его наименование указывается в соответствии с аттестатом аккредитации (прописными буквами), адрес (строчными буквами), телефон и факс.

В структуре регистрационного номера аккредитованного органа по сертификации также имеется пять элементов:

РОСС XX XXXX XX XXXXXX

1 – аббревиатура РОСС – принадлежность к Российской Федерации;

2 – местонахождение ОС (в виде двухсимвольного буквенного кода латинского алфавита);

3 – код национального органа, принявшего решение о внесении в Госреестр (“0001” – код Госстандарта России);

4 – категория ОС в зависимости от области аккредитации (например: “10” – ОС продукции и услуг, сертификационный центр; “11” – ОС продукции; “12” – ОС услуг; “13” – ОС систем качества; “14” – ОС производства);

5 – буквенно-цифровой код конкретного ОС, определенный объектом сертификации и порядковым номером данного ОС среди органов по сертификации конкретных объектов, внесенных в реестр.

## 4 Наименование и описание продукции:

“серийный выпуск”, “партия” или “единичное изделие”. Для партии и единичного изделия указывается номер и размер партии или номер изделия, номер и дата выдачи накладной, договора (контракта), документа о качестве и т. п. Если сертификат соответствия выдается на серийное производство, указывается: серийный выпуск. Здесь же дается ссылка на приложение “см. приложение” (если оно есть).

## 5 Код ОКП продукции. (шесть разрядов с пробелом после первых двух).

Код ОКП указывается на конкретную продукцию в соответствии с Общероссийским классификатором продукции.

## 6 Обозначение нормативных документов.

Указываются документы на соответствие которым проводится сертификация продукции (ГОСТ, ОСТ, ТУ, СанПиН т. д.) с указанием разделов или пунктов, на соответствие обязательным требованиям которых проведена сертификация.

## 7 Код ТНВЭД продукции.

Код продукции по Товарной номенклатуре внешнеэкономической деятельности (ТНВЭД) Российской Федерации – десятиразрядный код продукции (обязателен для импортируемой и экспортируемой продукции).

## 8 Изготовитель.

Указывается наименование изготовителя, его адрес, страна происхождения.

## 9 Сертификат выдан

Указывается наименование, реквизиты (адрес, телефон, факс) и ИНН юридического лица, которому выдан сертификат соответствия.

## 10 На основании.

Указываются документы, на основании которых выдан сертификат:

- протокол испытаний с указанием номера и даты выдачи, наименования и регистрационного номера аккредитованной лаборатории в государственном реестре;

- документы (санитарно-эпидемиологическое заключение, ветеринарное свидетельство, сертификат пожарной безопасности и др.), выданные органами и службами федеральных органов исполнительной власти, с указанием наименования органа или службы, адреса, наименования вида документа, номера, даты выдачи и срока действия;

- документы других органов по сертификации и испытательных лабораторий с указанием наименования, адреса, вида документа, номера, даты выдачи и срока действия; декларация о соответствии с указанием номера и даты принятия.

#### 11 Дополнительная информация.

Указывается дополнительная информация, приводимая при необходимости, определяемой органом по сертификации. К ней могут относиться условия действия сертификата (при хранении, реализации); вид тары и упаковки; информация о маркировке; место нанесения знака соответствия; номер схемы сертификации; дата изготовления, срок годности, условия хранения и т. п.

12 Подписи, инициалы, фамилии руководителя органа, выдавшего сертификат, и эксперта, проводившего сертификацию, печать органа по сертификации (левом нижнем углу).

Приложение к сертификату оформляется в соответствии с правилами заполнения аналогичных реквизитов в сертификате.

Сертификат и приложение к нему заполняют машинописным способом.

Исправления, подчистки и поправки не допускаются.

Цвет бланка сертификата соответствия

- при обязательной сертификации – желтый.

- при добровольной сертификации – голубой.

#### **Сертификат на электрические машины**

Законодательством Российской Федерации предусмотрено оформление сертификата на электродвигатель для следующих видов моделей:

электродвигатели тяговые для электропоездов и электровозов; электродвигатели главного привода и тягового оборудования для тепловозов.

Также в соответствии с Постановлением Правительства № 000, где утверждены основные Перечни товаров, требующих обязательной оценки соответствия, для некоторых видов электродвигателей необходимо получить декларацию качества, к таким категориям изделий относятся:

электродвигатели коллекторные и асинхронные мощностью до 1 кВт; электродвигатели асинхронные переменного тока мощностью до 1 кВт; асинхронные электродвигатели мощностью от 1 до 100 кВт.

В том случае, если выпускаемые или реализуемые электродвигатели не требуют обязательной оценки качества, а производитель или поставщик желают официально подтвердить их качество и безопасность, можно получить сертификат на электродвигатель в системе добровольной сертификации, позволяющей изделиям с успехом конкурировать на потребительском рынке страны.

Трансформаторы являются одним из видов энергетического оборудования. Они подлежат обязательной сертификации трансформаторов в системе ГОСТ Р, с последующим оформлением результатов проведённых испытаний декларацией соответствия. Основание: указанная продукция включена в перечень № 2, который правительство РФ утвердило 01.12.09 своим постановлением с регистрационным № 000. Последний раз указанный перечень корректировался 18.06.12. Обязательную декларацию соответствия, согласно этому документу, необходимо оформлять на трансформаторы, имеющие по ОК-0095 коды 3413, 3412 и 3411. А именно:

- на безопасные разделительные, разделительные;
- на трансформаторные подстанции, относимые к комплексным;
- на силовые однофазные трансформаторы, мощность которых превышает 4кВ А, на силовые трёхфазные, мощность которых составляет 6,3 кВ А и более.

Обязательная декларация соответствия на вышепоименованную продукцию может быть принята только при наличии у будущего декларанта результатов измерений, испытаний и исследований, полученных в уполномоченной на подобные испытания лаборатории, либо сертификата ИСО 9001 (сертификата СМК), выданного органом по сертификации, имеющим необходимую для этого аккредитацию.

Чаще всего производители для такого важного оборудования, как трансформаторы оформляют не только обязательные разрешительные документы, но и добровольные сертификаты на трансформаторы.

С 15 февраля 2013 года на территории стран Таможенного Союза действует

Технический регламент ТС о безопасности оборудования, которое относят к низковольтному. Такой регламент имеет код ТР ТС 004/2011 в соответствии с Программой разработки ТР. Принят он Решением КТС № 000 от 16 августа 2011 г., но далее был сдвинут срок ввода в действие данного регламента.

Сертификация трансформаторов происходит на основе ПП РФ № 000. С февраля 2014 года будут распространяться требования ТР ТС 004/2011.

Оформление сертификатов на трансформаторы в форме декларации соответствия ГОСТ Р будет завершено. Только обязательное подтверждение соответствия требованиям рассматриваемого регламента и оформление соответствующего сертификата на трансформатор станет официальным разрешением на его оборот на территории ТС.

В соответствии с данным ТР ТС на устройства, которые подходят к перечню, указанному в Приложении к данному регламентирующему документу, будет полагаться оформление обязательного сертификата на трансформатор ТР ТС. Для всех других электрических устройств, которые регулируются данным регламентом, будет полагаться оформление декларации соответствия по форме единой декларации.

Формы разрешительных сертификационных документов Таможенного Союза утверждены Решением КТС г. Точно также Решением КТС № 000 от 01.01.01г. введен на территории Таможенного Союза знак маркировки сертифицированной продукции (по требованиям Технических регламентов Таможенного Союза) – знак ЕАС.

Таким знаком разрешено маркировать только продукцию, которая прошла процедуры подтверждения соответствия по правилам и схемам сертификации или декларирования, прописанных в Решении КТС № 000 от 7 апреля 2011 года на подтверждение исполнения требований того действующего ТР ТС, которому подчиняется конкретная номенклатура изделий.

При этом сертификаты на трансформаторы, оформленные до даты опубликования рассматриваемого ТР ТС, будут действовать до срока, указанного в самом сертификационном документе в качестве срока окончания действия.

До 1 июля 2014 года будет разрешен еще выпуск продукции, относящейся к низковольтному оборудованию, на основе национальных требований и сертификатов, оформленных ранее в соответствии с национальными стандартами. Но маркировать знаком ЕАС такую продукцию запрещено. Действие таких национальных сертификатов распространяется только на территорию самого государства, которое выдало соответствующий разрешительный документ.

#### Задание

Проанализировать заданный сертификат соответствия и написать вывод о его годности

#### Ход работы

Получить у преподавателя вариант сертификата соответствия (СС) Проанализировать все позиции СС и записать информацию по плану: указать в какой системе выдан сертификат; привести знак (логотип) системы сертификации назвать орган по сертификации, выдавший сертификат соответствия; указать срок действия СС; указать на какую продукцию выдан сертификат; назвать изготовителя продукции; указать каким нормативным документам соответствуют данная продукция; указать на основании каких документов выдан СС; указать характер системы сертификации; указать какую цель преследует данный сертификат;

3 На основании анализа позиции данного СС написать вывод его годности.

#### Контрольные вопросы

Укажите какие признаки СС характеризуют его подлинность (действительность). Укажите какие признаки в СС указывают на его недействительность. Назовите какой характер может иметь система сертификации. Поясните какую цель преследует обязательная сертификация. Поясните какую цель преследует добровольная сертификация. Укажите какая из отечественных систем сертификации является основополагающей Укажите какой признак на упаковке товара указывает на то, что продукция прошла сертификационные испытания Укажите что необходимо иметь производителю для маркировки товара знаком соответствия Укажите какой МЗ на упаковке товара информирует покупателя о том, что товар имеет СС Поясните процесс сертификации, принимают участие третья сторона. Что это такое Назовите кто оплачивает сертификационные испытания Укажите каким внешним признаком отличаются системы сертификации Поясните, существует ли срок действия СС

## Отчет

1. Тема и цель работы
2. Экранные копии выполнения работы
3. Ответы на контрольные вопросы.

### Критерии оценивания отчета:

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие № 9

**Тема:** Разработка политики безопасности корпоративной сети (2 часа)

### Цель занятия:

1. изучить основные механизмы обеспечения безопасности сервера под управлением MS Windows Server, научиться настраивать параметры безопасности сервера, научиться использовать и настраивать средства аудита безопасности сервера

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows 7, FoxPro

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

#### ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

В операционной системе Windows Server2008 для уменьшения количества небезопасных точек взаимодействия последней с агрессивной средой применяется особый инструмент – **Мастер настройки безопасности**, который позволяет тонко настраивать параметры безопасности каждой из ролей, служб и компонентов. С помощью Мастера настройки безопасности можно сравнивать текущие параметры безопасности сервера с требуемой политикой безопасности, тем самым, получая сведения о слабых местах конфигурации системы.

Мастер настройки безопасности предоставляет администратору пошаговые инструкции по созданию, изменению, применению или откату политики безопасности. **Политики безопасности**, создаваемые с помощью Мастера настройки безопасности, представляют собой **XML-файлы**, которые при применении настраивают службы, безопасность сети, определенные параметры реестра и политику аудита.

В Windows Server 2008 роли, роли служб и компоненты, развернутые через соответствующие мастера, являются **защищенными по умолчанию** т.е. параметры сервера уже настроены так, чтобы обеспечить необходимую защиту. Но все же **после начальной** установки роли **рекомендуется** воспользоваться Мастером настройки безопасности, чтобы убедиться в адекватности защиты сервера, проверив тем самым наличие слабых мест в конфигурации, которые могут

измениться в процессе установки. В случае же **установки роли сервера без использования** соответствующего мастера, последующий запуск Мастера настройки безопасности просто **обязателен**.

Мастер настройки безопасности **устанавливается** по умолчанию **вместе** с Windows Server 2008, поэтому не требует дополнительной подготовки или установки. В состав мастера настройки безопасности входит программа управления через командную строку **Scwcmd.exe**, которую можно использовать для выполнения следующих задач:

- **настройки** одного или нескольких серверов с использованием политики, созданной с помощью мастера настройки безопасности;
- **анализа** одного или нескольких серверов для сравнения с политикой, созданной с помощью мастера настройки безопасности;
- **просмотра результатов** анализа в формате HTML;
- **отката** политик мастера настройки безопасности;
- **преобразования** политики, созданной с помощью мастера настройки безопасности, в файлы, которые поддерживаются групповой политикой;
- **регистрации** расширений базы данных настройки безопасности с помощью мастера настройки безопасности.

Информационной средой функционирования Мастера настройки безопасности является **База данных настроек безопасности**, представляющая собой набор XML-документов, в которых перечисляются службы и порты, необходимые для каждой роли сервера, поддерживаемой мастером настройки безопасности. Эти файлы размещаются в папке **%Systemroot%\Security\Msscw\Kbs**. Базу данных настройки безопасности можно **расширить** путем создания определений пользовательских ролей **для приложений сторонних разработчиков**.

При запуске Мастера настройки безопасности для заданного сервера выполняется **поиск** по базе данных с целью определения следующих сведений:

- какие роли установлены на данном сервере;
- какие роли вероятно выполняются данным сервером;
- какие службы установлены, но не входят в базу данных настройки безопасности;
- какие IP-адреса и подсети настроены для данного сервера.

**Найденные** сведения Мастер настройки безопасности объединяет в **единый** XML-файл с именем **Main.xml**, который можно затем просмотреть либо через нажатие кнопки с окне Мастера настройки безопасности, либо через командную строку **Scwcmd.exe**.

## **ПРАКТИЧЕСКАЯ ЧАСТЬ**

### **Задание 1**

**Создать прототип (шаблон) политики безопасности для каждого сервера в соответствии с исполняемой им ролью.**

Указания к выполнению

1. Запустите Мастер настройки безопасности (меню Пуск последовательно выберите пункты Администрирование и Мастер настройки безопасности). Прочтите страницу приветствия и нажмите кнопку Далее.

2. На странице Действие настройки нажмите кнопку Создать новую политику безопасности, а затем кнопку Далее.

3. На странице Выбор сервера введите имя прототипа сервера и нажмите кнопку Далее.

4. На странице Обработка базы данных настройки безопасности после завершения обработки нажмите кнопку Далее (база данных настройки безопасности настраивается с учетом ролей указанного сервера).

5. На странице Настройка служб на основе ролей нажмите кнопку Далее. На странице Выбор ролей сервера убедитесь, что мастер обнаружил и выбрал все роли сервера, а затем нажмите кнопку Далее.

6. На страницах выбора клиентских возможностей, других параметров и служб убедитесь, что мастер обнаружил и выбрал все установленные компоненты, параметры и службы, а затем нажмите кнопку Далее.

7. На странице Подтверждение изменений для служб проверьте изменения режима службы, которые мастер настройки безопасности включит в созданную политику безопасности, а затем нажмите кнопку Далее.

8. На странице Сетевая безопасность нажмите кнопку Далее.

9. На странице Правила сетевой безопасности убедитесь, что мастер настройки безопасности обнаружил соответствующие порты и приложения, которые будут использоваться для настройки брандмауэра Windows с улучшенной защитой, а затем нажмите кнопку Далее.

¥ Внесите в отчет скриншот с отчетом мастера о текущей настройке сети

10. На странице Параметры реестра установите флажок Пропустить этот раздел и нажмите кнопку Далее.

11. На странице Политика аудита установите флажок Пропустить этот раздел и нажмите кнопку Далее.

12. На странице Сохранение политики безопасности нажмите кнопку Далее.

13. На странице Имя файла политики безопасности введите имя прототипа политики и нажмите кнопку Далее.

14. На странице Применение политики безопасности нажмите кнопку Применить позже, а затем кнопку Далее.

15. На странице Завершение мастера настройки безопасности нажмите кнопку Готово. Политика создана и сохранена, но еще не применена на компьютере.

## **Задание 2**

### **Применить созданную политику безопасности к серверу.**

Указания к выполнению

1. Запустите Мастер настройки безопасности.

2. На странице Действие настройки нажмите кнопку Применить существующую политику безопасности и кнопку Далее (если создано несколько политик, нажмите кнопку Обзор, а затем дважды щелкните имя требуемой политики).

3. На странице Выбор сервера введите имя сервера, к которому будет применена политика, и нажмите кнопку Далее.

4. На странице Применение политики безопасности нажмите кнопку Далее (для предварительной проверки параметров политики можно нажать кнопку Просмотр политики безопасности). Дождитесь завершения процесса и нажмите кнопку Далее.

¥ Внесите в отчет скриншот с результатами предварительной проверки создаваемой политики

5. На странице Завершение мастера настройки безопасности нажмите кнопку Готово.

## **Задание 3**

### **Проанализировать и просмотреть политики безопасности для каждого сервера безопасности с помощью программы командной строки Scwcmd.**

Указания к выполнению

1. В командной строке выполните следующую команду:  
scwcmd analyze /m: *Имя\_компьютера* /p: *Имя\_политики.xml* /o: *Папка\_сохранения\_результатов*

2. После завершения анализа выполните следующую команду:  
scwcmd view /x:

C:/Windows/security/msscw/TransformFiles/ *Файл\_результатов.xml* /s:SCWAnalysis.xsl (Scwanalysis.xsl – это установленный с мастером настройки безопасности файл, который форматирует результаты анализа для отображения. Оба файла должны располагаться в одном и том же каталоге).

¥ Внесите в отчет скриншот распечатку файла результатов анализа сервера на предмет соответствия безопасности.

## **Задание 4**

### **Сохранить созданную для каждого сервера политику безопасности в виде объекта групповой политики, применяемой к объектам Active Directory.**

Указания к выполнению

1. Введите в командную строку команду: **scwcmd transform /p: *Имя\_политики.xml* /g: *Имя\_объекта\_групповой\_политики***. (После завершения команды объект групповой политики будет

создан в доменных службах Active Directory, но политика, которая в нем содержится, не будет применяться до тех пор, пока объект не будет связан с сайтом, доменом или подразделением).

¥ Внесите в отчет скриншот, отображающий наличие созданного объекта в структуре Active Directory.

### **Задание 5**

**Создать расширенную политику аудита в домене, обслуживающем факультет информатики: отслеживание входа в домен, отслеживание причин доступа к объектам и отслеживание доступа к объектам.**

Указания к выполнению

1. Запустите оснастку управления групповой политикой (Пуск, Администрирование, Управление групповой политикой).

2. В дереве консоли выберите домен inf.edu, перейдите к элементу Политика домена по умолчанию, а затем выберите пункт Изменить.

3. Через пункты Конфигурация компьютера, Политики, Конфигурация Windows, Параметры безопасности, Конфигурация расширенной политики аудита перейдите к элементу Политики аудита системы.

4. Выберите пункт Вход/выход, а затем дважды щелкните пункт Вход.

5. Установите флажок Настроить следующие события аудита, поставьте флажки в полях Успех и Отказ, а затем нажмите кнопку ОК.

6. Вернитесь в окно Политики аудита системы, выберите пункт Доступ к объектам, щелкните правой кнопкой мыши пункт Работа с дескриптором, затем нажмите кнопку Свойства.

7. Установите флажок Настроить следующие события аудита, поставьте флажки в полях Успех и Отказ, а затем нажмите кнопку ОК.

8. Вернитесь в окно Политики аудита системы, дважды щелкните пункты Доступ к объектам и Файловая система.

9. Установите флажок Настроить следующие события аудита и затем установите флажки Успех, Отказ или оба флажка – Успех и Отказ, а затем нажмите кнопку ОК.

¥ Внесите в отчет скриншоты, отображающие произведенные вами настройки политики аудита.

### **Задание 6**

**Проверить правильность применения расширенной политики аудита.**

Указания к выполнению

1. Войдите на корневой контроллер домена с учетной записью администратора.

2. Запустите программу командной строки от имени администратора.

3. При появлении диалогового окна Контроль учетных записей пользователей подтвердите действие и нажмите кнопку Да.

4. Введите в командной строке команду auditpol.exe /get /category:\* и подтвердите ввод.

5. Убедитесь, что элементы Успех, Отказ или Успех и отказ отображаются справа от элемента Вход.

¥ Внесите в отчет скриншот с отчетом о результатах проверки применения политики расширенного аудита.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Авторизация

2. Мастер настройки безопасности,

3. Для выполнения каких задач используют программу управления через командную строку Scwcmd.exe.

### **Отчет**

Сохраните в своей папке под названием Лабораторная работа 9.

**Критерии оценивания отчета:**

|                                                 |                               |                         |                         |
|-------------------------------------------------|-------------------------------|-------------------------|-------------------------|
| Своевременность и успешность составления отчета | Качество усвоенного материала | Творческая составляющая | Перевод баллов в оценку |
|-------------------------------------------------|-------------------------------|-------------------------|-------------------------|

|                                             |                                                  |                                                           |                                                                                                     |
|---------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов   | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов     | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

## Лабораторное занятие № 10

**Тема:** Получение сертификата (2 часа)

### Цель занятия:

1. изучение видов сертификации объектов информатизации..

### ИНСТРУКЦИОННАЯ КАРТА

#### Материально - техническое оснащение

**Оборудование:** компьютер, операционная система Windows

**Характер выполнения работы:** обучающиеся выполняют работу индивидуально.

#### Практические задания

**Цифровой сертификат** – электронный документ, выданный и заверенный Удостоверяющим центром.

Цифровой сертификат - это небольшой файл, содержащий в себе следующую информацию:

1. имя и идентификатор владельца сертификата;
2. открытый ключ подписи (шифрования);
3. имя, идентификатор и цифровую подпись Удостоверяющего центра;
4. серийный номер, версию и срок действия сертификата.

**Инфраструктура открытых ключей** (англ. *PKI - Public Key Infrastructure*) — набор средств (технических, материальных, людских и т. д.), распределенных служб и компонентов, в совокупности используемых для поддержки криптозадач на основе закрытого и открытого ключей.

В основе PKI лежат несколько основных принципов:

1. закрытый ключ известен только его владельцу;
2. удостоверяющий центр создает сертификат открытого ключа, таким образом удостоверяя этот ключ;
3. никто не доверяет друг другу, но все доверяют удостоверяющему центру;
4. удостоверяющий центр подтверждает или опровергает принадлежность открытого ключа заданному лицу, которое владеет соответствующим закрытым ключом.

#### Ход работы

**Упражнение №1** – знакомство со структурой сертификата, самоподписанные сертификаты.

С помощью консоли работы с личными сертификатами пользователя можно познакомиться с возможностями манипулирования уже полученными сертификатами.

1. Попытались зарегистрироваться на компьютере ClientStand от имени локального пользователя LStudent. Но на этом компьютере не оказалось пользователя LStudent, поэтому мы зарегистрировались от имени пользователя User1.
2. В консоль управления (mmc.exe) добавили оснастку работы с сертификатами для текущего пользователя.

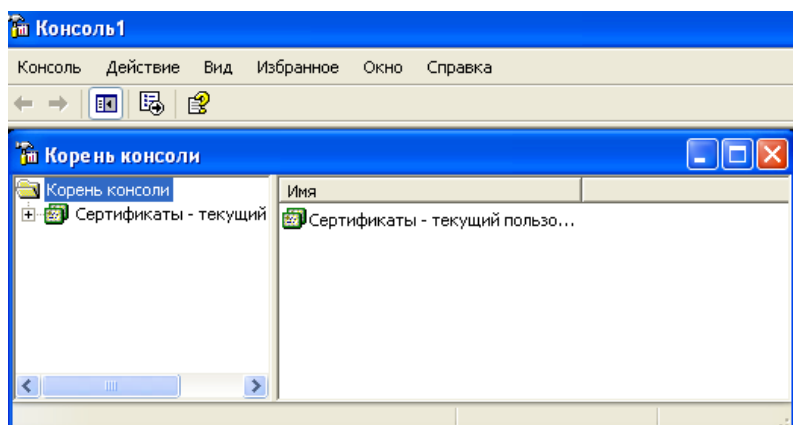


Рисунок 1-1. Добавление оснастки работы с сертификатами текущего пользователя

3. Проверка наличия или отсутствия сертификатов в разделе персональных сертификатов.

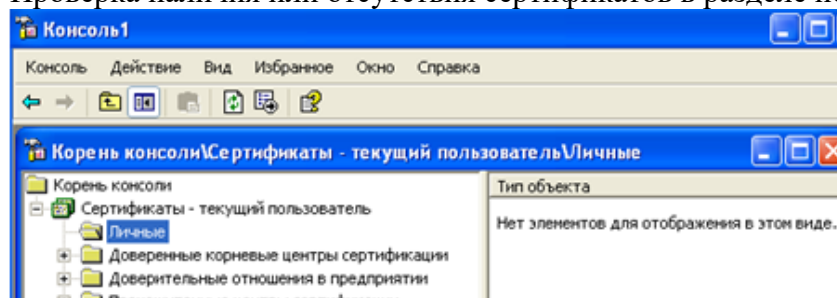


Рисунок 1-2. Личные сертификаты

Видно, что личных (персональных) сертификатов у данного пользователя нет.

4. С помощью файлового менеджера на устройстве в каталоге «Encrypted» зашифровали файл Encr1.txt с помощью правой кнопки мыши.

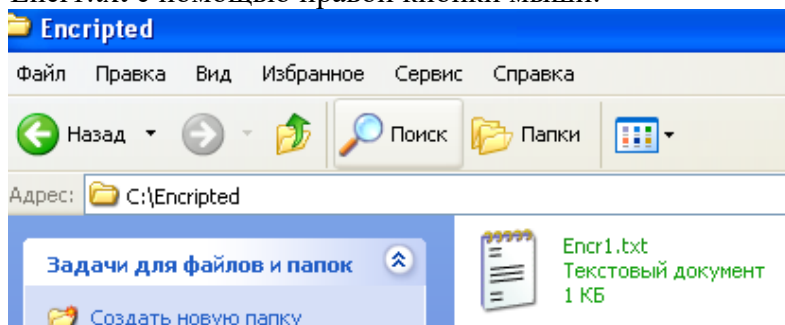


Рисунок 1-3. Зашифрованный с помощью свойств данного объекта файл

5. Проверка наличия сертификатов в разделе персональных сертификатов в консоли работы с сертификатами.

Так как мы использовали шифрование данных, то в разделе личные сертификаты появился сертификат шифрования пользователя.

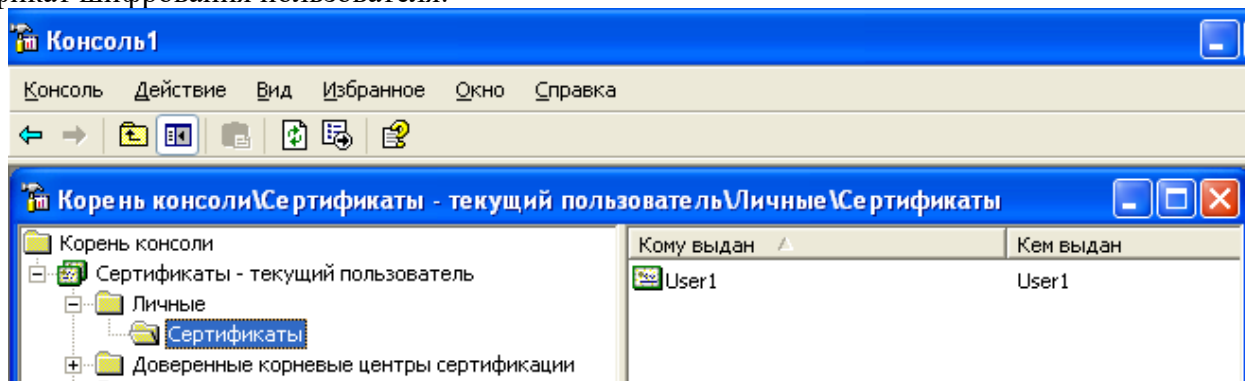


Рисунок 1-4. Новый сертификат

6. Выбираем сертификат для знакомства с его свойствами и составом.

К свойствам сертификата относятся:

- статус доверия (сертификат может являться надежным, подразумевать доверие и являться ненадежным);

- срок сертификата (срок окончания действия сертификата);
- тип сертификата (отображает формат сертификата);
- отпечаток (отображает цифровую подпись)

Далее также можно просмотреть состав данного сертификата

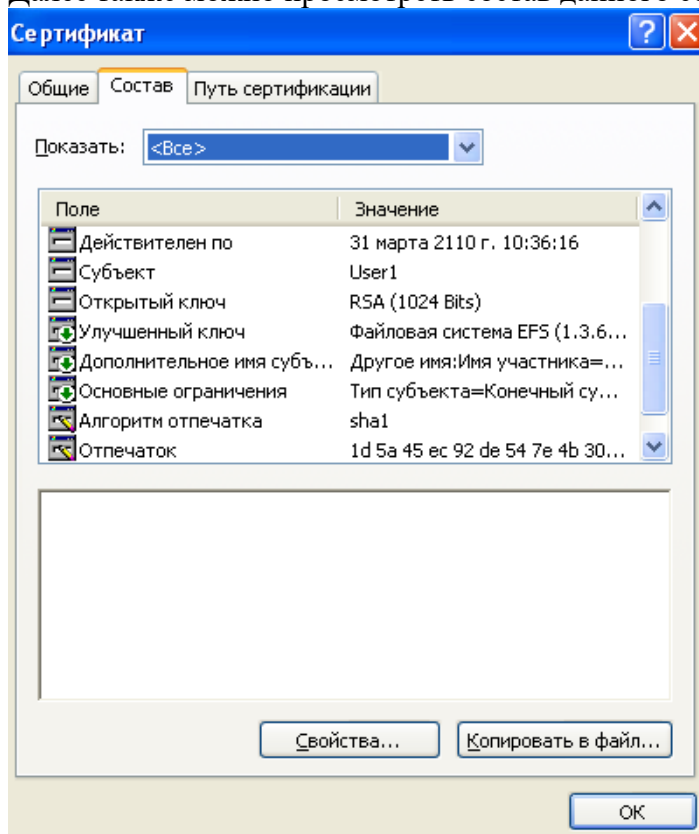


Рисунок 1-4. Состав нашего сертификата

*Упражнение №2* – управление центром выдачи сертификатов (Certificate Authority – CA), автономный и корпоративный CA

Задача упражнения познакомиться с возможностями получения сертификатов через Web от автономного центра сертификации, а также с методами работы с корпоративным центром выдачи сертификатов.

1. Установим центр сертификации на этом компьютере, при этом надо указывать имя текущего компьютера – SRV1.

Для начала необходимо через «Панель управления» -> «Установка и удаление программ» установить программу, такую как служба сертификатов.

Затем подключаем центр сертификации.

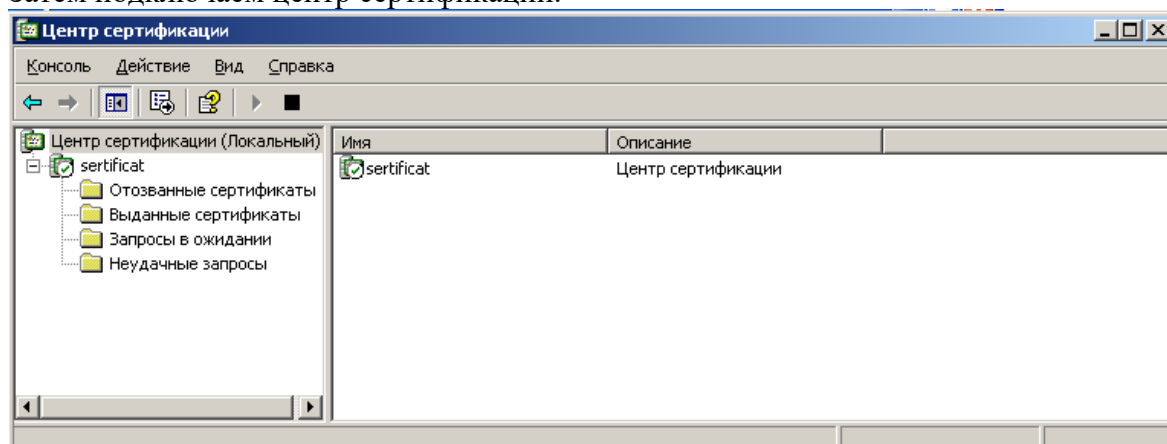


Рисунок 2-1. Центр сертификации

2. В консоль управления (mmc.exe) добавим оснастку работы с центром сертификации
3. В панели работы с центром сертификации посмотрим были ли уже выданы какие-то сертификаты.

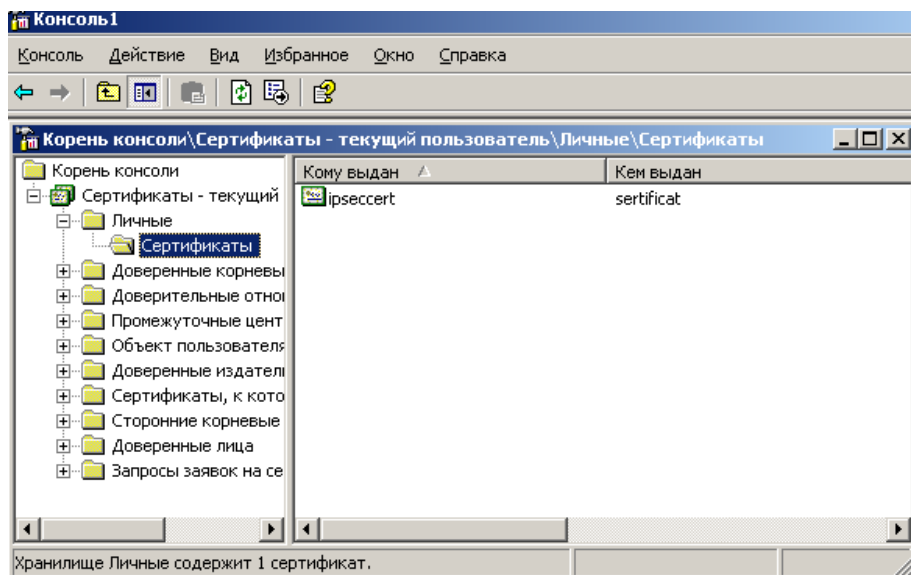


Рисунок 2-2. Сертификат

Ранее был выдан сертификат ipsecert через Web, и именно его видим в личных сертификатах.

- Открываем Обозреватель Интернета. Идём по адресу: <http://SRV1/certsrv>. Появляется открытая стартовая страница центра выдачи сертификатов.

Стартовая страница находится по адресу: <http://localhost/certsrv/>, где имеется возможность создать новый сертификат через Web.

- Через ссылку «запрос сертификата», а затем «расширенный запрос на сертификат» запрашиваем сертификат аутентификации клиента.

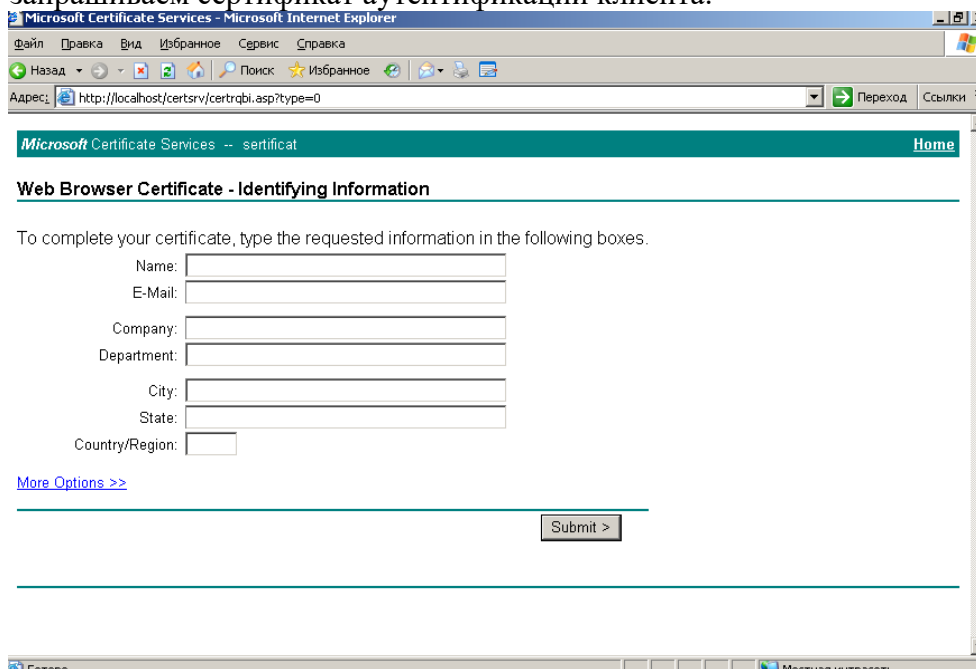


Рисунок 2-3

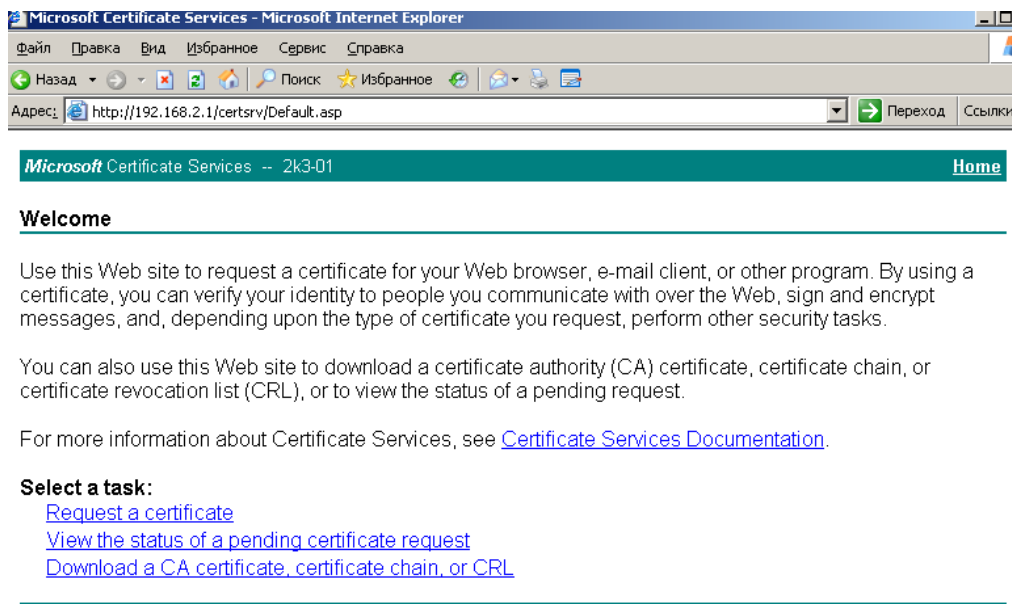


Рисунок 2-4. Окно обозревателя

6. В панели работы с центром сертификации MMC в пункте «запросы в ожидании» выдадим запрашиваемый сертификат.

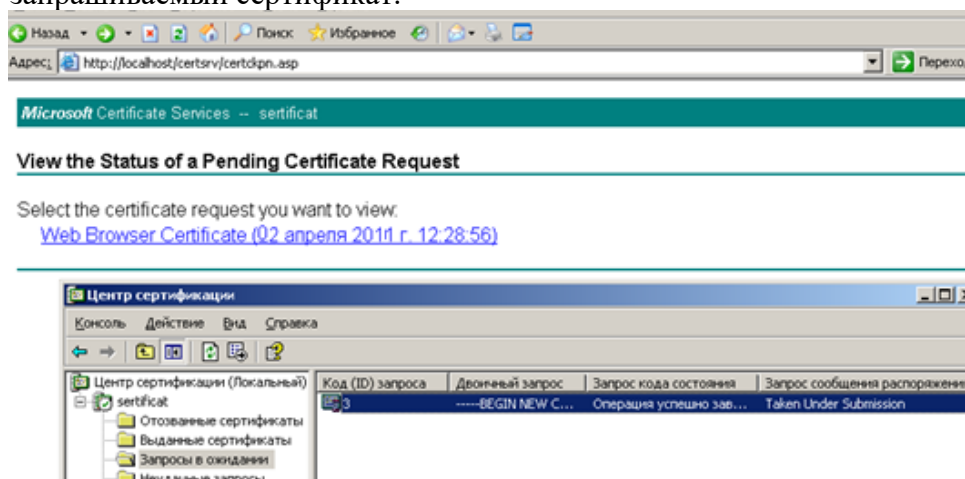


Рисунок 2-5. Запросы в ожидании

7. Идём обозреватель Интернета. Вернувшись к стартовому меню, установили выданный сертификат.
8. В панели работы с сертификатами клиента видим, что сертификат получен.

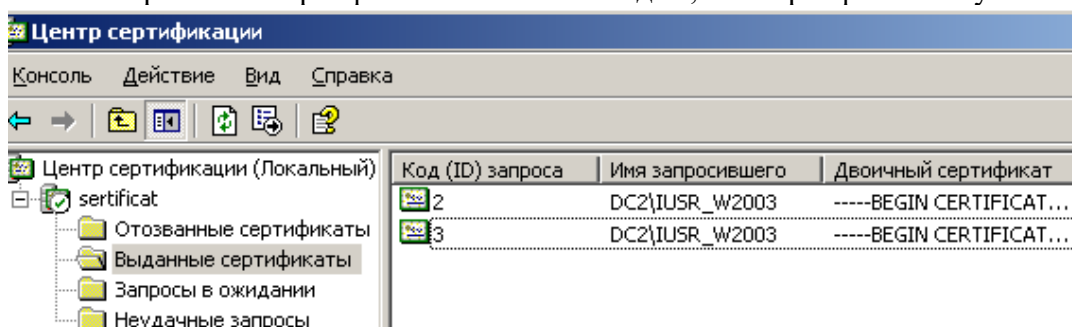


Рисунок 2-6. Новый выданный сертификат

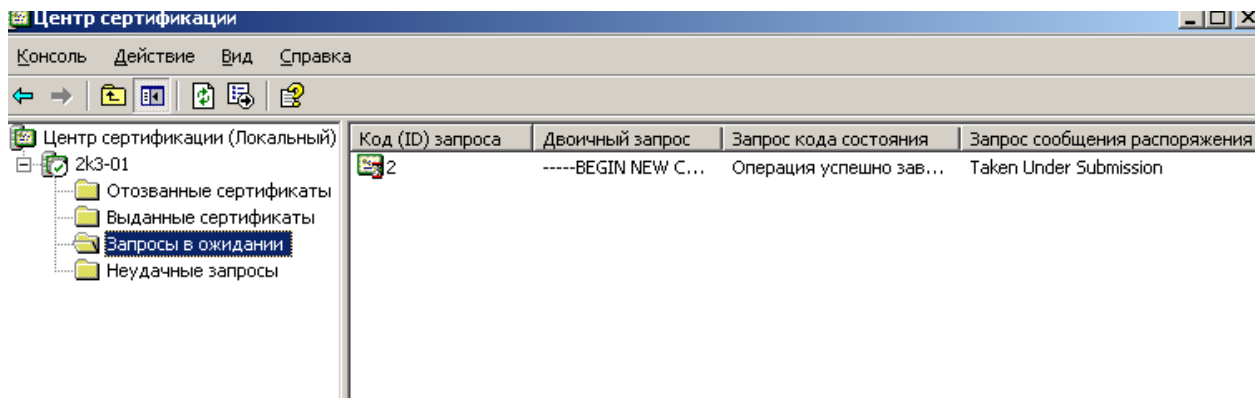


Рисунок 2-7. Проверка запросов в ожидании

Значит, мы убедились в том, что сертификат выдан, т.к. он отсутствует в ожиданиях и уже присутствует в разделе выданные сертификаты.

*Упражнение 3 – запрос и получение сертификатов с СА.*

В случае использования службы Active Directory можно запрашивать сертификаты через панель работы с сертификатами MMC или вообще запрашивать и получать сертификаты автоматически, без участия пользователя. При этом и «одобрения» от администратора центра выдачи сертификатов не потребуется – все проверки выполнит Active Directory.

1. Запустили MMC.
2. Добавление оснастки работы с сертификатами пользователя.  
Вошли в домен Sarg под именем Администратор, добавили оснастку.
3. Запрашиваем сертификат для шифрованной файловой системы из панели управления сертификатами пользователя.

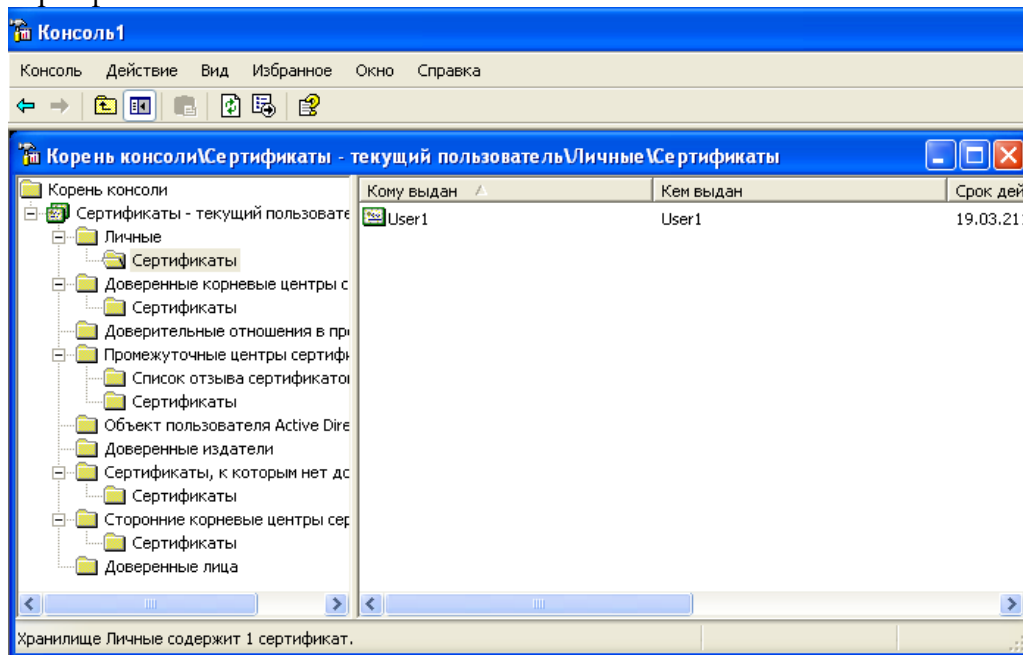


Рисунок 3. Запрос сертификата для шифрованной системы

Попытка сделать запрос сертификата для шифрования файловой системы была неудачна. Видимо невозможно получить такой сертификат, так как нет возможности опознать центр сертификатов на данном компьютере.

Такой тип сертификата можно получить, например, с помощью Web, если там поддерживается данный тип сертификата.

**Выводы:** в данной лабораторной работе мы ознакомились с сертификатами. Выяснили возможности проверки и выдачи сертификатов с помощью служб сертификации.

В ходе выполнения упражнений были предложены различные способы для предоставления тех или иных сертификатов. Использовались такие методы, как:

- выдача локального сертификата,
- выдача сертификатов через Web

Первый способ позволил нам выдать сертификат самим себе при шифровке файла.

Вопросы:

1. Как правильно выдать сертификат посредством запроса?
2. Как узнать, что данный ключ принадлежит данному пользователю?
3. Можно ли создать сертификаты на основе какого-то другого сертификата, т.е. возможность создания дочерних сертификатов. И как?

**Отчет**

1. Тема и цель работы
2. Экранные копии выполнения работы
3. Ответы на контрольные вопросы.

**Критерии оценивания отчета:**

| Своевременность и успешность составления отчета | Качество усвоенного материала                    | Творческая составляющая                                   | Перевод баллов в оценку                                                                             |
|-------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Своевременно выполнена работа – 10 баллов       | Точность выполнения работы – 10 баллов           | Защита работы аудиторно – 20 баллов                       | 70 б – оценка «5»<br>40-60 б – оценка «4»<br>30 баллов – оценка «3»<br>Менее 20 баллов – оценка «2» |
| Досрочно выполненная работа – 20 баллов         | Эстетичность выполнения работы – 10 баллов       | Защита своей работы персонально – 10 баллов               |                                                                                                     |
| Не своевременно выполнена работа – 0 баллов     | Полное содержание выполненной работы – 10 баллов | Оформление своей работы несколькими способами – 20 баллов |                                                                                                     |

### Информационные источники

#### Печатные издания:

1. Мартишин, С. А. Базы данных практическое применение СУБД SQL и NOSQL-типа для проектирования информационных систем: учеб. пособие.-М.: ИД "ФОРУМ": ИНФРА-М, 2020.

#### Электронные издания (электронные ресурсы):

1. ЭБС «Научная электронная библиотека eLIBRARY» [Электронный ресурс]. – Режим доступа: <http://elibrary.ru/defaultx.asp>
2. ЭБС «Издательства Лань» [Электронный ресурс]. – Режим доступа: <http://e.lanbook.com/>
3. ЭБС «Университетская библиотека online» [Электронный ресурс]. – Режим доступа: [www.biblioclub.ru](http://www.biblioclub.ru)
4. ЭБС «Библиокомплектатор» [Электронный ресурс]. – Режим доступа: (<http://www.bibliocomplectator.ru/>)
5. ЭБС «ЮРАЙТ» [Электронный ресурс]. – Режим доступа: (<http://biblio-online.ru>)
6. Свободный каталог периодики библиотек России [Электронный ресурс]. – Режим доступа: <http://ucpr.arbicon.ru/>

#### Дополнительные источники:

1. Сoadминистрирование баз данных и серверов/ Перлова О.Н. 2020
2. Сопровождение и обслуживание программного обеспечения компьютерных систем/ Зверева В.П. 2020
3. Федорова Г.Н. Основы проектирования баз данных. –М.: ОИЦ «Академия» 2020
4. Сергеев, А. Г. Стандартизация и сертификация : учебник и практикум для среднего профессионального образования / А. Г. Сергеев, В. В. Терегеря. — Москва : Издательство Юрайт, 2019. — 323 с
5. Разработка, администрирование и защита баз данных/ Федорова Г.Н. 2020