Задание

Повторить теоретический материал, выполнить защиту информации в базе данных своего курсового проекта.

С уважением, Хвастова Светлана Ивановна

!!! Если возникнут вопросы обращаться по телефону 0721389311.
Электронная почта: xvsviv@rambler.ru

КП. Обеспечение безопасности БД. Аспекты информационной безопасности базы данных

Содержание

Введение

Одной из важнейших характеристик качества любой информационной системы является уровень обеспечения ее информационной безопасности[1].

Проблема обеспечения безопасности информационных систем может быть определена как решение трех взаимосвязанных задач по реализации требуемого уровня [1]:

- *конфиденциальности*, т.е. обеспечения пользователям доступа только к данным, для которых пользователь имеет явное или неявное разрешение на доступ;
- *целостности*, т.е. обеспечения защиты от преднамеренного или непреднамеренного изменения информации или процессов ее обработки;
- *доступности*, т.е. обеспечения возможности авторизованным в системе пользователям доступа к информации в соответствии с принятой технологией.

Поскольку информация (данные) в информационных системах, как правило, хранится в базах данных, то имеет смысл говорить о безопасности баз данных или системы баз данных, имея в виду обеспечение безопасности как при хранении, так и при обработке данных.

Для создания баз данных и управления данными в них используются системы управления базами данных (СУБД). Современные СУБД имеют встроенные средства обеспечения безопасности данных. Поэтому в работе будут рассмотрены эти средства для некоторых наиболее распространенных СУБД.

Управляемая система баз данных, как правило, является распределенной, т.е. физически может быть размещена на нескольких носителях, а возможно, и в нескольких узлах, взаимодействие между которыми осуществляется по протоколам транспортного уровня. Поэтому анализ информационной безопасности СУБД должен быть проведен по двум направлениям [1]:

- Безопасность архитектурных решений и их программных реализаций собственно в СУБД, которая включает исследование следующих проблем:

> идентификация и аутентификация субъектов системы;

> технологии реализации дискреционной, мандатной и ролевой модели доступа к данным;

> реализация аудита действий пользователя.

- Безопасность взаимодействия с внешними по отношению к СУБД программными и аппаратными компонентами, которая включает исследование следующих проблем:

> сопряжение с элементами операционной системы (анализ информационных потоков вниз), т.е. изучение возможностей пользователей несанкционированно осуществлять чтение и запись в файлы операционной системы, включая возможность модификации записей аудита;

> сопряжение с программным обеспечением промежуточного уровня (анализ управляющих потоков вверх), т.е. выявление портов взаимодействия с внешним программным обеспечением, устойчивость к перегрузкам каналов шумовым трафиком и вставкам ложных пакетов, к перенастройкам параметров протоколов, а также анализ алгоритмов и технологий линейного шифрования трафика межсерверного обмена и взаимодействия клиентского программного обеспечения с серверами баз данных.

В работе в основном ограничимся рассмотрением вопросов, связанных с первым направлением.

Новым направлением для СУБД промышленного уровня является наличие встроенных механизмов шифрования в основном на базе алгоритмов DES и AES.

1 Средства защиты баз данных MS Access 2003

СУБД Microsoft Access (MS Access) – это единственный в своем роде продукт, который поставляется вместе с Microsoft Office и используется миллионами людей, как разработчиками, так и простыми пользователями [2]. Эта СУБД появилась на рынке баз данных в октябре 1992 года. СУБД MS Access может использоваться как для создания локальных баз данных и пользовательских приложений, так и для создания распределенных баз данных и многопользовательских приложений.

В этом разделе будут рассмотрены вопросы защиты приложения с помощью встроенных механизмов обеспечения защиты, начиная с версии MS Access 97 [2] . Эти механизмы включают в себя защиту:

- при помощи пароля, который нужно указывать при каждом открытии базы данных;
- на уровне пользователей;
- с помощью MDE-файла;
- программным путем, используя встроенный объектно-ориентированный язык программирования для приложений VBA (Visual Basic for Applications).

Прежде чем устанавливать защиту базы данных при помощи пароля или на уровне пользователя рекомендуется всегда делать резервные копии базы данных и файла рабочей группы (System.mdw (см. раздел 1.2.1)) и копировать эти резервные копии в специально отведенное для этого место.

1.1. Защита при помощи пароля

Ядро Jet (Join Engine Technology) версии 3.5 и более поздние версии ядра СУБД MS Access предоставляют возможность установить пароль на базу данных, который нужно будет вводить при каждом открытии базы данных. Следует отметить, что защита базы данных при помощи пароля и защита на уровне пользователя независимы друг от друга. Это означает, что даже если пользователь знает пароль, ему все равно нужно иметь разрешения на работу с объектами базы. Если пользователь забывает пароль, то не существует способа удалить этот пароль или открыть базу данных. Поэтому пользоваться паролем нужно предельно осторожно. Если пароль не забыт, то его можно удалить. Для этого нужно открыть базу данных в монопольном режиме, имея права администратора или владельца базы. Рассмотрим, как это можно сделать.

Пусть создана база данных (mdb-файл) по имени **Моя БД**, которую необходимо защитить паролем. Для этого нужно открыть базу данных в монопольном режиме:

- запустить MS Access;
- выполнить команду $\Rightarrow$ *Файл/Открыть* (или по кнопке );
- в открывшемся окне указать имя базы данных (**Моя БД**);
- в этом же окне открыть список **Открыть**, в котором выбрать **монопольно**.

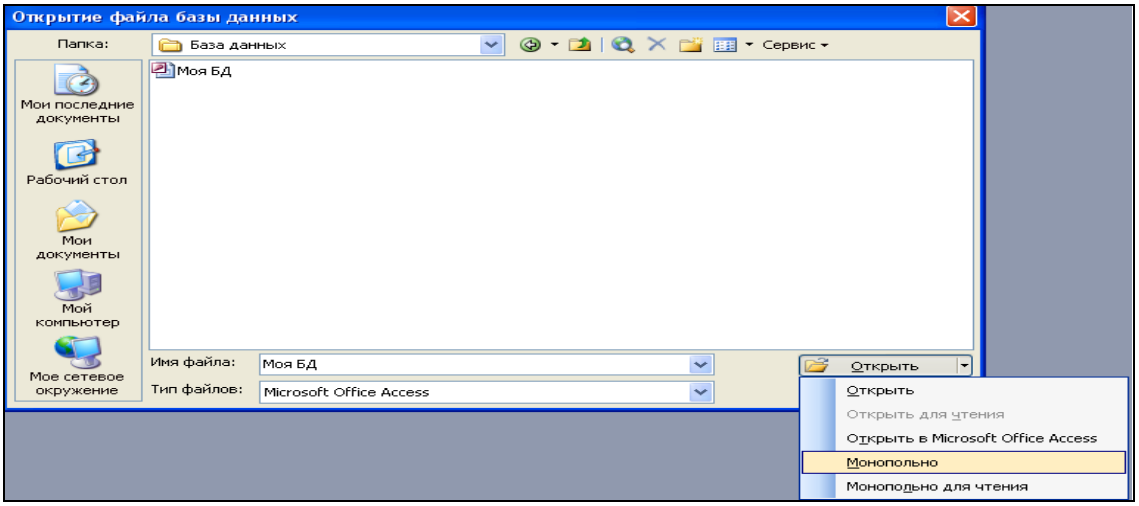

Рисунок 1 – Открытие базы данных в монопольном режиме

Откроется база данных в монопольном режиме. Чтобы задать пароль, нужно выполнить команду системного меню

⇒ *Сервис/Защита/Задать пароль базы данных*

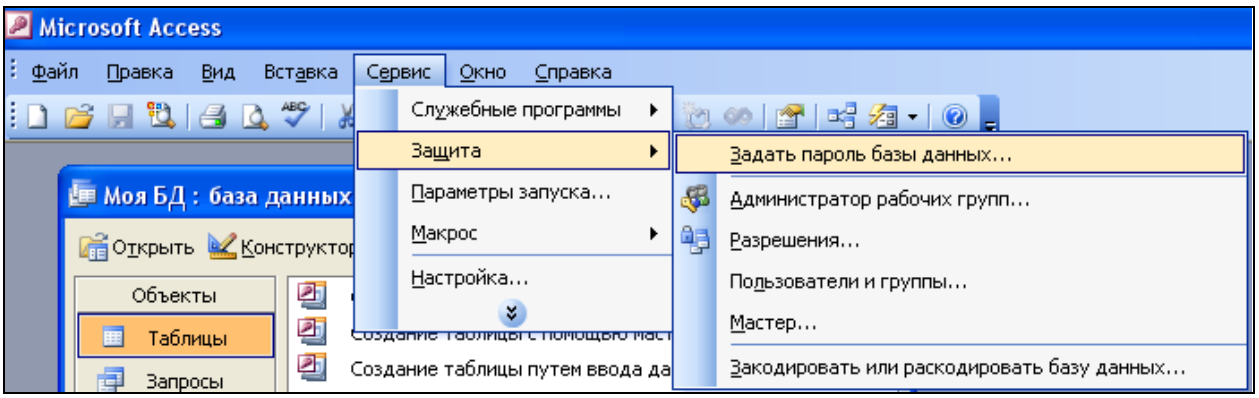

Рисунок 2 – Выбор команды для задания пароля на открытие базы данных

Откроется окно для задания пароля:

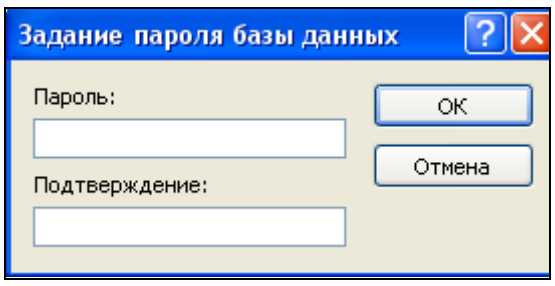

Рисунок 3 – Задание пароля

Удалить пароль можно так:

- открыть базу данных **Моя БД** в монопольном режиме, как было сказано выше, введя пароль;
- выполнить команду системного меню

⇒ *Сервис/Защита/Удалить пароль базы данных*

Чтобы изменить пароль, надо открыть базу данных в монопольном режиме, удалить старый пароль и задать новый.

Задать, изменить и удалить пароль можно программным путем, используя язык программирования VBA, например, следующим образом:

- открыть базу данных **Моя БД** в монопольном режиме;
- открыть редактор VBA командой

⇒ *Сервис/Макрос/Редактор Visual Basic*

- создать стандартный модуль (все программы на языке VBA оформляются в виде процедур, а процедуры размещаются в модулях) командой:

⇒ *Insert/Module*

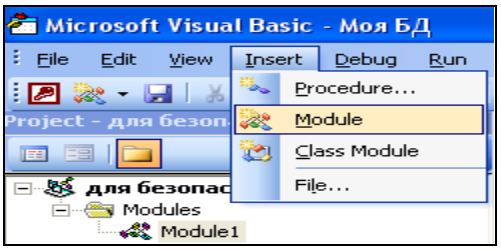

Рисунок 4 – Создание стандартного модуля

Откроется окно модуля, в котором нужно создать три процедуры (для создания пароля, изменения пароля и удаления пароля).

- создать процедуру для задания пароля командой

⇒ *Insert/Procedure*

Откроется окно для создания процедуры, в котором надо указать имя процедуры (пусть это будет **создать**), тип процедуры (пусть это будет подпрограмма Sub) и область видимости процедуры (можно оставить по умолчанию Public):

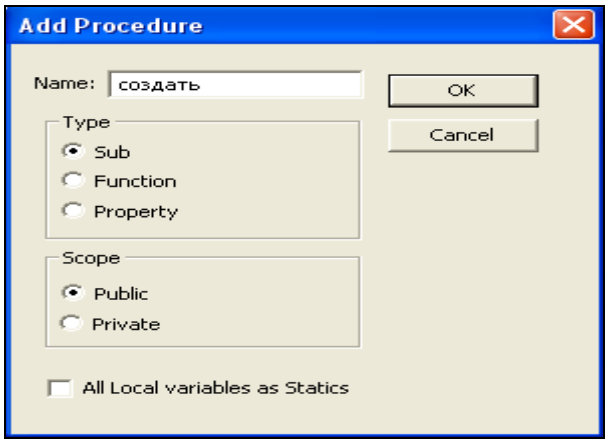

Рисунок 5 – Добавление в модуль процедуры для создания пароля

По кнопке ОК в модуле будет создан пустой шаблон процедуры, в котором отображается заголовок процедуры (Public Sub создать() и завершающий процедуру оператор (End Sub) :

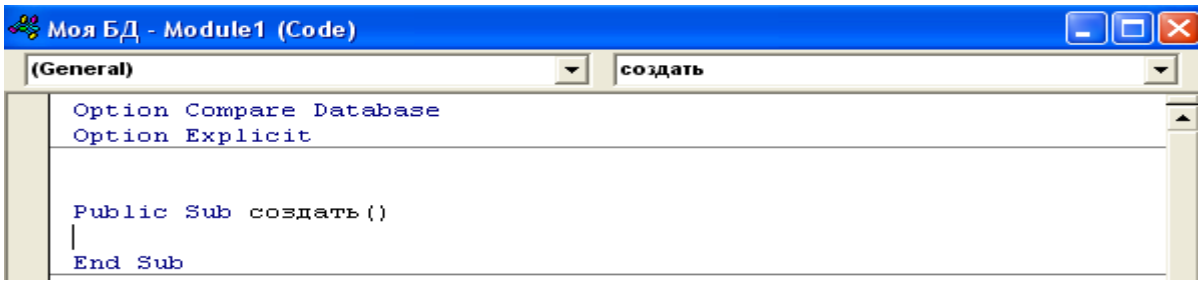

Рисунок 6 – Пустой шаблон процедуры

• заполнить тело процедуры операторами (набрать с помощью клавиатуры, комментарии можно не набирать):

```
Public Sub создать()
'--- описание объектной переменной типа базы данных
Dim MyDB As Database
'--- присваивание ей значения текущей (открытой) базы данных
Set MyDB = CurrentDb()
'--- задание пароля (123) с помощью метода NewPassword
MyDB.NewPassword "", "123"
End Sub
```

Здесь кавычки "" означают, что у базы данных пароля не было, а "123" – что создан пароль 123 (как строка символов, строки символов заключаются в двойные кавычки).

Чтобы выполнить эту процедуру, надо поместить курсор в любое место внутри этой процедуры и выполнить команду (см. рис. 7):

⇒ *Run/Run Sub/UserForm* (или кнопкой ▶)

• проверить, действительно ли установлена защита с помощью пароля. Для этого надо закрыть базу данных, затем ее снова открыть и ввести пароль.

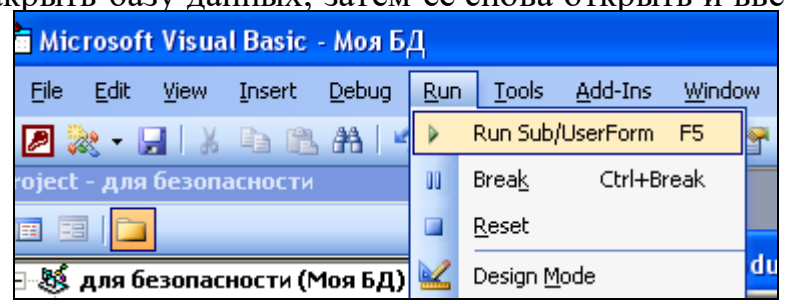

Рисунок 7 – Запуск процедуры на выполнение

• аналогично добавить в модуль еще две процедуры на изменение и удаление пароля:

```
Public Sub создать()
'--- описание объектной переменной типа базы данных
Dim MyDB As Database
'--- присваивание ей значения текущей (открытой) базы данных
Set MyDB = CurrentDb()
'--- задание пароля (123) с помощью метода NewPassword
MyDB.NewPassword "", "123"
End Sub

Public Sub изменить()
Dim MyDB As Database
Set MyDB = CurrentDb()
MyDB.NewPassword "123", "1234"
End Sub

Public Sub удалить()
Dim MyDB As Database
Set MyDB = CurrentDb()
MyDB.NewPassword "1234", ""
End Sub
```

Здесь "123" – старый, а "1234" – новый пароли, а "" – означает, что пароль не установлен.

1.2. Защита на уровне пользователя

Каждая база данных, которая создается или используется в среде MS Access, содержит информацию обо всех пользователях и их правах доступа к объектам базы [2]. Многие не пользуются защитой, поэтому по умолчанию MS Access регистрирует вошедшего пользователя под именем администратора (Admin), который имеет полный доступ ко всем объектам базы. Однако мы можем воспользоваться защитой базы данных на уровне пользователя. В этом случае пользователь должен при открытии базы данных ввести имя пользователя и пароль. Только после этого пользователь получит доступ к тем объектам, разрешения на использование которых были выданы ему администратором базы. Таким образом, разрешения на доступ к объектам и регистрацию пользователя, которому выданы соответствующие разрешения, осуществляет только администратор базы или пользователь, обладающий правами администратора. Это можно сделать как через интерфейс MS Access, так и программным путем. Рассмотрим оба варианта.

1.2.1. Обеспечение защиты через интерфейс Access

В MS Access защита на уровне пользователя задействована всегда [2]. Многие не пользуются защитой, поэтому по умолчанию MS Access регистрирует вошедшего пользователя как пользователя Admin без пароля.

На рисунке 8 представлена схема организации защиты на уровне пользователя. Следует отметить "слабость" такой защиты. Тем не менее рассмотрим ее подробнее.

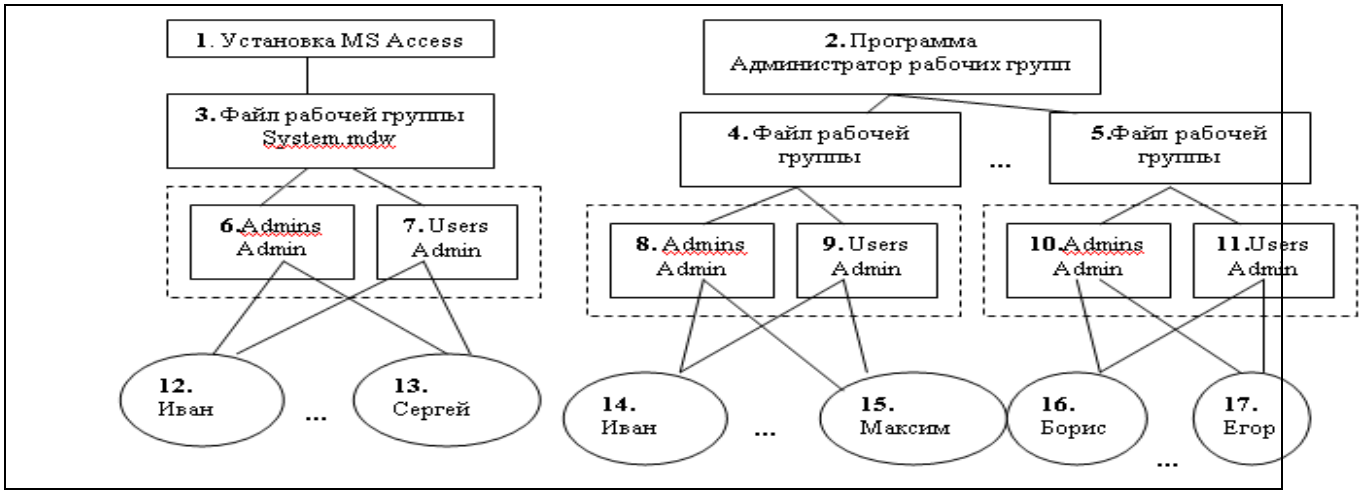

Рисунок 8 – Схема защиты базы данных на уровне пользователя

После установки MS Access (блок 1 на рисунке 8) автоматически создается файл рабочей группы System.mdw (блок 3). При запуске MS Access информация о пользователях и группах пользователей получается из этого файла. Файл System.mdw - системная база данных относится к категории скрытых системных файлов. Полное имя этого файла можно узнать, открыв базу данных и выполнив команду

⇒ *Сервис/Защита/Администратор рабочих групп*

Чтобы пройти по указанному пути, нужно, чтобы были доступны скрытые файлы и папки. Для этого нужно запустить проводник и выполнить команду

⇒ *Сервис/Свойства папки/вкладка* ***Вид****/включить переключатель* ***Показывать скрытые файлы и папки*** в папке **Скрытые файлы и папки**

Далее, используя проводник, можно открыть эту базу данных (файл System.mdw), которая содержит скрытые системные таблицы и запросы:

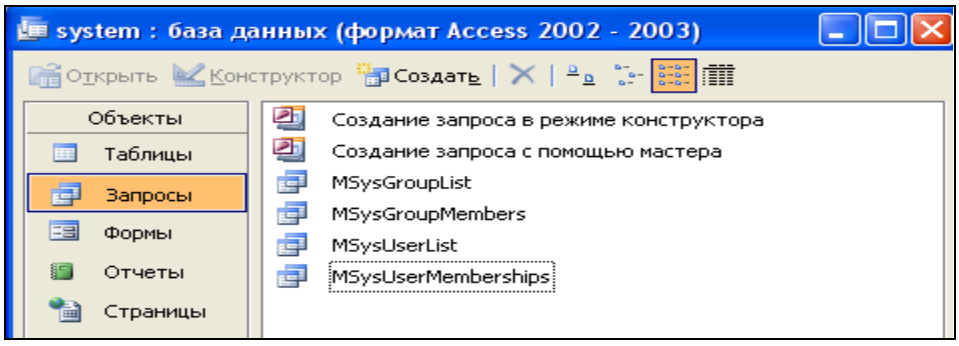

Рисунок 9 – База данных **System.mdw**

Запрос MSysGroupList позволяет получить список групп пользователей. Первоначально это две группы Admins и Users (блоки 6 и 7 на рисунке 8).

Запрос MSysGroupMembers – запрос с параметром, позволяет получить список членов введенной в диалоге группы пользователей. Первоначально обе группы Admins и Users содержат только пользователя Admin (блоки 6 и 7 на рисунке 8).

Запрос MSysUserList позволяет получить список всех пользователей, содержащихся во всех группах. Первоначально такими пользователями являются Admin, Creator и Engine). Последние два пользователя устанавливаются автоматически и поддерживаются механизмом ядра Jet.

Запрос MSysUserMemberships – запрос с параметром, позволяет указать группу, в которую входит введенный в диалоге пользователь.

Можно также принудительно создавать файлы рабочих групп (блоки 4 и 5) с помощью программы (Администратор рабочих групп – блок 2).

В файлах рабочих групп можно создавать новые учетные записи отдельных пользователей и групп пользователей. Каждой группе администратор может выдать разрешения на доступ к объектам базы данных. В группу могут входить несколько пользователей, которым предоставляются одинаковые разрешения для работы с объектами базы данных, а именно такие, какие были выданы группе. Такие разрешения называют неявными (все пользователи группы наследуют разрешения, данные группе). Таким образом, использование групп позволяет упростить процедуру предоставления разрешений для пользователей, включенных в группу.

Однако каждому пользователю администратор может выдать дополнительные разрешения (явные разрешения), которые могут отсутствовать у группы, в которую входит пользователь. В этом случае такой пользователь наследует все разрешения своей группы, а также и дополнительные разрешения. Таким образом,но при этом в любом случае будут использоваться разрешения, которые обеспечивают ему максимальный доступ.

По умолчанию в файле рабочей группы создаются две группы: Admins и Users (блоки 6 – 11). Пользователь Admin является членом обеих групп Admins и Users (как показано в блоках 6 – 11). Пользователь Admin не может быть исключен из группы Admins. Пользователь Admin может добавлять новых пользователей (блоки 12 – 17) и давать им разрешения. Добавленный пользователь автоматически становится членом группы Users. По умолчанию члены этой группы могут выполнять следующие действия:

- создавать новые базы данных;
- изменять системные установки;
- восстанавливать базы данных;
- сжимать базы данных.

1.2.1.1. Создание файлов рабочих групп

Для создания файлов рабочих групп (блоки 4, 5 на рисунке 8) нужно выполнить следующие действия:

- открыть базу данных **Моя БД** (если при этом не было предложено ввести имя пользователя и пароль, то это означает, что регистрация произошла под именем Admin с пустым паролем);
- запустить программу администратора рабочих групп командой системного меню (рисунок 10)

⇒ *Сервис/Защита/Администратор рабочих групп*

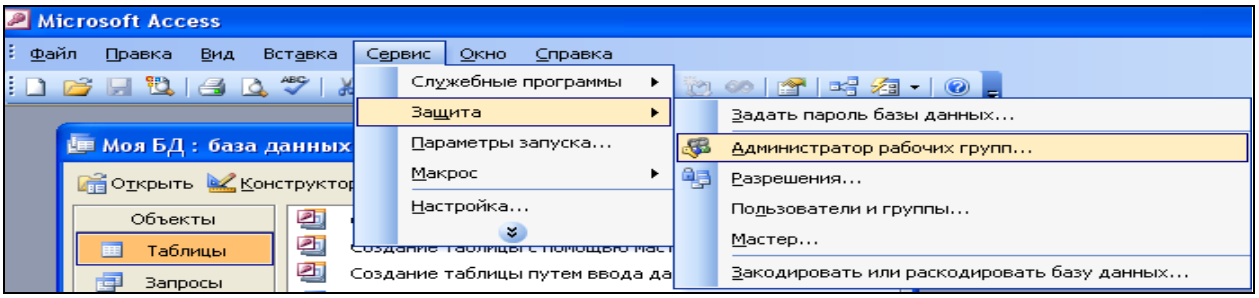

Рисунок 10 – Запуск программы администратора рабочих групп

Откроется окно Администратора рабочих групп, в котором нужно выбрать кнопку **Создать** для создания новой рабочей группы или кнопку **Связь** для связи с нужной рабочей группой (если создано несколько групп). В этом же окне прописан путь к файлу System.mdw.

- Для создания новой рабочей группы выбираем кнопку **Создать**. Откроется окно для задания сведений о новой рабочей группе. Имя, организация и код группы введены произвольно (рисунок 11). Однако их следует записать, если вы будете создавать еще рабочие группы.

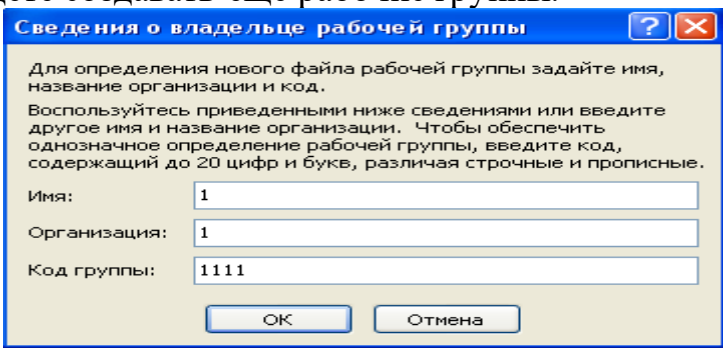

Рисунок 11 – Задание сведений о владельце новой рабочей группы

Будет создан файл **System1.mdw**, содержащий информацию о пользователях созданной группы.

Аналогично можно создавать еще файлы рабочих групп.

Таким образом, имеем два файла: **System.mdw** и **System1.mdw**, в которых пока имеется по две группы: **Admins** и **Users**.

1.2.1.2. Добавление новых пользователей и групп в файлы рабочей группы и задание им разрешений

По материалам предыдущего раздела имеем два файла рабочих групп: **System.mdw** и **System1.mdw**. Здесь рассмотрим вопросы добавления новых групп пользователей и отдельных пользователей в группы, а также вопросы, связанные с заданием им разрешений на доступ к объектам базы данных **Моя БД**. В качестве объектов могут использоваться таблицы, формы, запросы, отчеты, макросы и сама база данных, но не отдельные поля таблицы и не отдельные элементы управления. Как выдать разрешение на столбцы таблицы или на элементы управления на форме будет рассмотрено в разделе 1.2.3.

Прежде, чем приводить пример, рассмотрим общую схему организации защиты через интерфейс MS Access.

Итак, при открытии вновь созданной базы данных (например, **Моя БД**) всегда будет существовать база данных – файл **System.mdw**, с которой автоматически поддерживается связь на уровне администратора рабочих групп. Используя команды меню

⇒ *Сервис/Защита...*

Можно создать несколько групп пользователей с разными разрешениями на доступ к объектам базы **Моя БД**. В свою очередь, каждая группа может содержать несколько пользователей, которые автоматически будут иметь разрешения, предоставленные группе, в которую эти пользователи включены. Такие разрешения называются *неявными*. Можно создать отдельных пользователей, не включенных ни в какую группу, и этим пользователям выдать разрешения. Такие разрешения называются *явными*.

Пользователю, включенному в группу, могут быть выданы более сильные разрешения, чем разрешения для группы. В этом случае такой пользователь наследует разрешения группы, в которую он входит, но при этом в любом случае будут использоваться для него максимальные разрешения.

Используя администратор рабочих групп, можно создать еще файлы рабочих групп **System1.mdw**, **System2.mdw**, … со своими группами пользователей, которым также можно выдать соответствующие разрешения на доступ к объектам базы данных **Моя БД**.

Рассмотрим последовательность действий для одной группы из файла **System.mdw**, в которую входят несколько пользователей, которым выданы только неявные разрешения.

- При открытой базе **Моя БД** создать связь с файлом **System.mdw**, используя команду меню: ⇒ *Сервис/Защита/Администратор рабочих групп*
- Выдать группе разрешения на открытие базы данных и разрешения на доступ к объектам базы. Тогда и всем пользователям, входящим в эту группу, будут выданы неявно такие же разрешения.
- Удалить все разрешения (если они есть) из группы Users, оставив разрешение на открытие базы данных.
- Сменить владельца объектов на владельца группы командой:

⇒ *Сервис/Защита/Разрешения/*вкладка ***Смена владельца***

- Удалить разрешения для пользователя Admin, оставив ему разрешение только на открытие базы данных.
- Открыть базу данных от имени одного из пользователей, входящих в группу. По умолчанию база данных для нового пользователя открывается с пустым паролем.
- Изменить пароль для этого пользователя, используя команду меню:

⇒ *Сервис/Защита/Пользователи и группы/*вкладка ***Изменение пароля***

• Открыть базу данных **Моя БД** от имени пользователя группы и убедится в том, что этот пользователь будет иметь разрешения предоставленные группе, в которую он включен.

Пример 1. Добавим в файл **System.mdw** две новые группы:

Группа 1 с кодом **1111** и **Группа 2** с кодом **2222**.

В первую группу включим двух пользователей: **Ивана** с кодом **иии** и **Сергея** с кодом **сссс**.

Во вторую группу включим **Машу** с кодом **ммм** и **Юлю** с кодом **ююю**.

В файл **System1.mdw** добавим пользователя **Кирилла** с кодом **кккк**.

Пользователям группы 1 (**Ивану** и **Сергею**) разрешим удалять данные из таблицы **поставки** и добавлять данные в таблицу **поставщики**, пользователям группы 2 (**Маше** и **Юле**) – читать и обновлять данные таблицы **детали**, **Кириллу** – добавлять данные в таблицу **поставки** (рисунок 12):

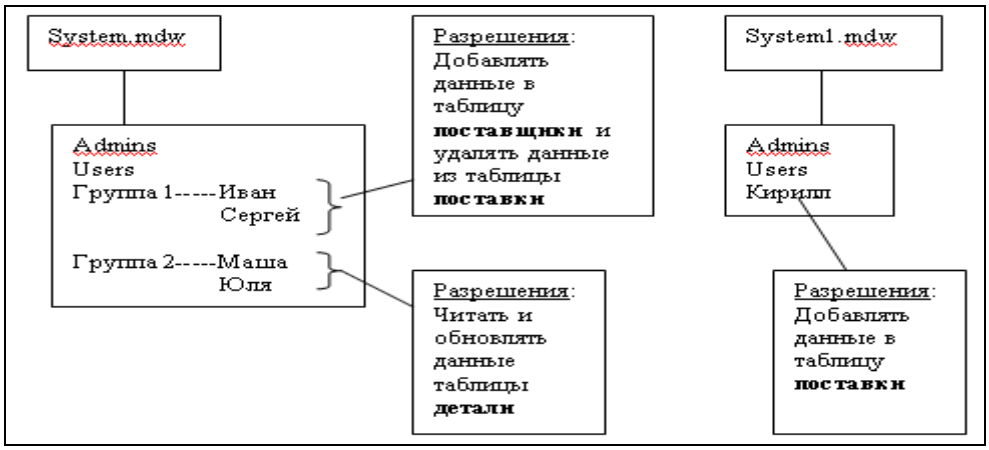

Рисунок 12 - Разрешения

Это можно сделать с помощью следующих действий (не забывайте, что сейчас мы по умолчанию имеем права администратора с пустым паролем):

• откроем базу **Моя БД**;

• свяжем базу данных с файлом **System.mdw** командой:

⇒ *Сервис/Защита/Администратор рабочих групп*

• создадим две группы пользователей (**Группа 1** с кодом 1111 и **Группа 2** с кодом 2222), выполнив команду ⇒ *Сервис/Защита/Пользователи и группы*;

> в открывшемся окне активизируем вкладку **Группы**;

> активизируем кнопку **Создать**;

> открывается окно для добавления новой группы, в котором набираем с клавиатуры информацию о группе, как показано на рисунке 13:

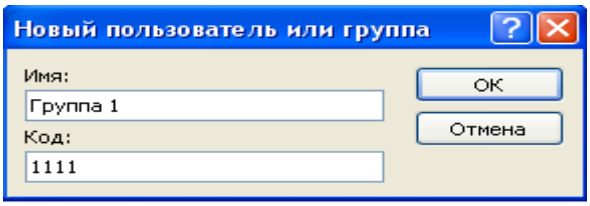

Рисунок 13 – Добавление новой группы

Аналогично добавляем вторую группу (**Группа 2** с кодом **2222**);

- для проверки откроем базу данных – файл **System.mdw**;
- выполняя запрос MSysGroupList, получим список групп в файле рабочей группы **System.mdw**:

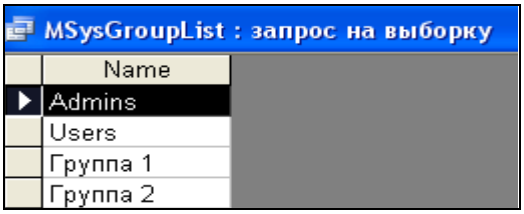

Рисунок 13 – Список групп

- добавим пользователя **Ивана** с кодом **иии** в группу 1 командой

⇒ *Сервис/Защита/Пользователи и группы*;

> в открывшемся окне активизируем вкладку **Пользователи** и кнопку **Создать**;

> набираем имя и код пользователя Ивана (рисунок 14):

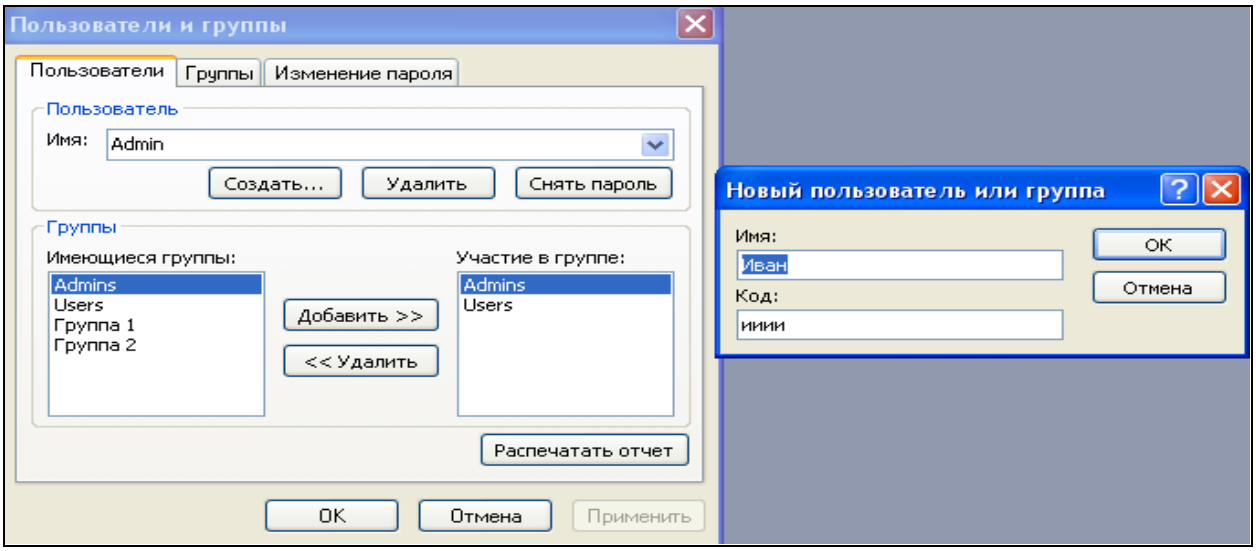

Рисунок 14 – Добавление нового пользователя

Следует помнить о том, что имена пользователей и коды зависят от регистра.

Пользователь или группа будут распознаваться по имени. Имя пользователя или группы (ID) и соответствующий код (PID) будут использоваться Microsoft Jet для автоматического создания системного кода (system ID, SID). Код пользователя или группы (PID) должен состоять из любой комбинации символов и цифр и иметь размер от 4-х до 20 символов.

После закрытия окна **Новый пользователь или группа** в окне **Пользователи и группы** появится в строке *имя* **Иван** вместо **Admin**, далее кнопкой **Добавить** надо добавить **Группу 1** из списка имеющихся групп в список **Участие в группе** (так как Иван является участником в группе 1). В результате окно **Пользователи и группы** примет вид:

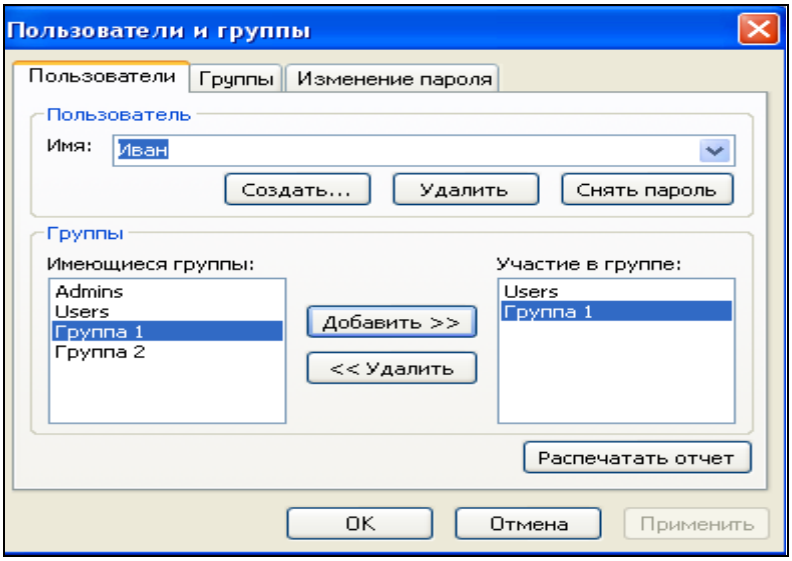

Рисунок 15 – Результат добавления пользователя Ивана в Группу 1

- аналогично добавляем в Группу 1 пользователя **Сергея** с кодом **cccc**, а в Группу 2 пользователей **Машу** с кодом **мммм** и **Юлю** с кодом **ююююю**;
- откроем базу данных – файл **System.mdw,** и в результате выполнения запроса MSysUserList, получим:

Рисунок 16 – Пользователи базы данных **Моя БД**

Как было сказано выше, кроме добавленных нами пользователей (Иван, Маша, Сергей и Юля) пользователями базы данных являются admin,Creator и Engine. Последние два пользователя устанавливаются автоматически и поддерживаются механизмом ядра Jet.

- выполнив соединение с файлом рабочей группы **System1.mdw**, добавим в эту рабочую группу пользователя **Кирилла** с кодом **кккк**. Следует учесть, что новые пользователи по умолчанию всегда добавляется в группу Users (если принудительно этот пользователь не добавляется в группу Admins). Таким образом, **Кирилл** будет добавлен в группу **Users** файла рабочей группы **System1.mdw**, а пользователи Иван, Маша, Сергей и Юля – в группу **Users** файла рабочей группы **System.mdw**.

• дадим разрешение **Группе 1** на удаление данных из таблицы **поставки** и добавление данных в таблицу **поставщики**:

> сначала создадим связь с файлом **System.mdw**, воспользовавшись администратором рабочих групп:

⇒ *Сервис/Защита/Администратор рабочих групп*;

> откроем окно **Разрешения**:

⇒ *Сервис/Защита/Разрешения*;

> в окне **Разрешения** сделаем соответствующие установки:

включить переключатель **группы**, выделим группу **Группа 1**, тип объекта **Таблица**, выделить таблицу **поставки**, установить флажок **удаление данных** (при этом автоматически будут установлены флажки **чтение макета** и **чтение данных**) и кнопкой **Применить** дадим разрешение на таблицу **поставки**.

Далее для выделенной группы **Группа 1** выделим таблицу **поставщики** и установим флажок **вставка данных** (при этом автоматически будут установлены флажки **чтение данных** и **чтение макета**) и кнопку **Применить**.

После активации кнопок **Применить** и **ОК** группе 1, а, следовательно, и всем ее членам будут даны установленные разрешения.

• аналогично даются разрешения для группы **Группа 2** и для пользователя **Кирилл** (в файле рабочей группы **System1.mdw**.

Теперь нужно убедиться, что данные разрешения установлены. Разрешения, выданные группе, устанавливаются автоматически для всех членов группы. Однако какому-либо члену группы можно выдать и более сильные разрешения.

Проверим разрешения для пользователя Ивана, который входит в первую группу. Поскольку ему никаких дополнительных разрешений не было установлено, то он имеет разрешения, выданные группе 1, т.е. Иван может удалять данные из таблицы **поставки** и добавлять данные в таблицу **поставщики**.

Для этого нужно выполнить следующую последовательность действий:

• сменить владельца объектов базы данных, например, на владельца группы 1, как показано на рисунке 17:

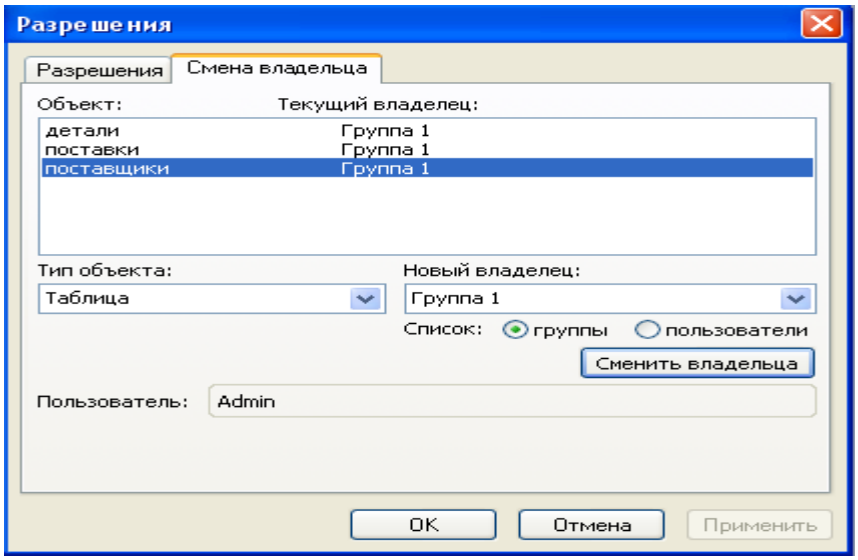

Рисунок 17 – Смена владельца на группу 1

- **удалить разрешения у группы Users** (так как по умолчанию группе Users даются все права) и у пользователя Admin, оставив у них только разрешение на открытие базы данных и монопольный доступ;
- сменить пароль пользователя Admin (пустой пароль) на непустой пароль как показано на рисунке 18 (здесь звездочками набран пароль **аааа**):

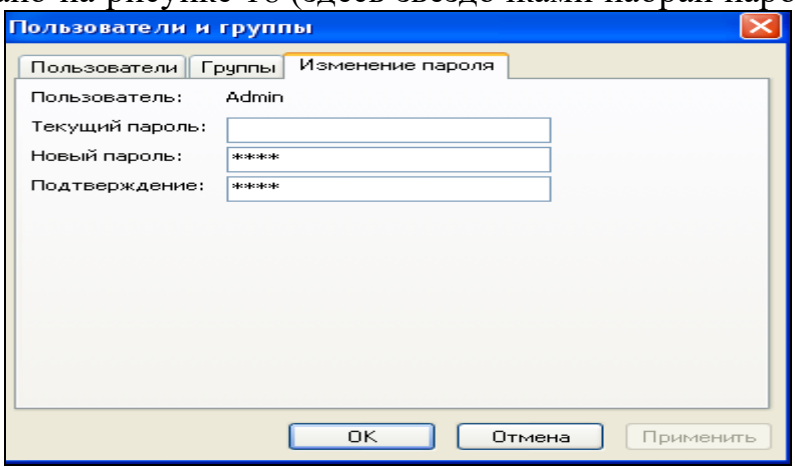

Рисунок 18 – Изменение пароля администратора

- открыть базу данных Моя БД от имени Ивана (без пароля, так как открытие базы данных для нового пользователя осуществляется по умолчанию без пароля)– члена группы 1:

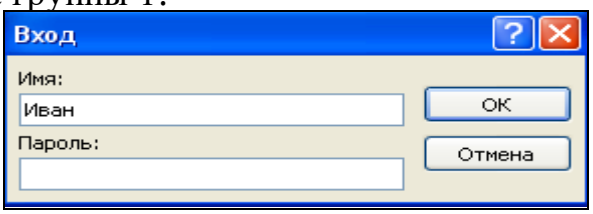

Рисунок 19 – Открытие базы данных от имени Ивана с пустым паролем

- теперь можно задать пароль **иии** для **Ивана** командой

⇒ *Сервис/Защита/Пользователи и группы*/вкладка ***Смена пароля***

• чтобы перейти к другому пользователю этой же группы, например, к Сергею, надо открыть базу данных, изменить имя пользователя на Сергея, а затем задать пароль **cccc** Сергею. Теперь можно открывать базу данных для Сергея, который неявно будет иметь разрешения, данные группе 1.

• Чтобы перейти к пользователям группы 2, надо пользователю Сергею дать права администратора для объектов базы данных. Далее надо открыть базу данных от имени Сергея и сменить владельца на группу 2:

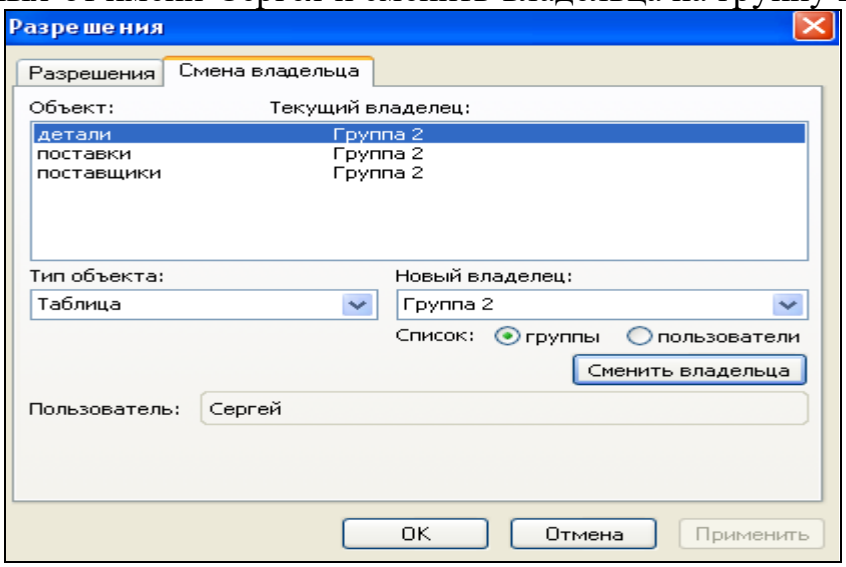

Рисунок 20 – Смена владельца на группу 2

• Снять пароль с Сергея, открыть базу данных, заменив в окне входа Сергея на пользователя из группы 2, например на Машу (сначала с пустым паролем), а затем дать Маше пароль **мммм**.

• Открыть базу данных **Моя БД** от имени Маши и убедиться в том, что Маша получила все права, прописанные группе 2.

Аналогичные действия можно выполнить для Юли – пользователя из группы 2, а также для Кирилла из файла рабочей группы **System1.mdw**.

Слабость рассмотренной защиты состоит в том, что администрирование может быть разрешено пользователям, которые не обладают правами администратора, однако возможно их наделить такими правами. Это касается только объектов базы данных, но не самой базы данных. Так, если владельцем указана Группа 1, то для любого пользователя этой группы могут быть даны права администратора, который может для этой группы или для любого пользователя этой группы изменить разрешения на объекты базы данных. Но этот пользователь не может создавать новые группы или новых пользователей базы данных. Также он не может удалять группы или отдельных пользователей. Это может делать только пользователь, включенный в группу Admins. Аналогичные рассуждения можно провести и для пользователей Группы 2.

Если рассмотренный способ установки защиты данных покажется сложным, можно попытаться установить защиту с помощью мастера, как описано в следующем разделе.

1.2.2. Обеспечение защиты с помощью мастера

Программа-мастер позволяет автоматически установить защиту на уровне пользователя. Прежде чем запустить мастера необходимо выполнить следующие действия:

- создать и присоединить новый файл рабочей группы командой

⇒ *Сервис/Защита/Администратор рабочих групп/*кнопка ***Создать*** и затем кнопкой **Связь**;

- задать пароль для пользователя Admin:

⇒ *Сервис/Защита/Пользователи и группы/*вкладка ***Смена пароля***;

- создать нового пользователя, например, **админ**, которого добавить в группу Admins:

> дать пользователю админ права администратора на базу данных и на все объекты базы;

> сменить владельца с Admin на **админ**;

> отобрать у пользователя Admin все разрешения;

> удалить Admin из группы Admins; пользователь Admin остался в группе Users.

- открыть базу данных Моя БД, которую надо защитить.

Теперь можно запускать мастера защиты базы данных:

- запустить мастера защиты командой

⇒ *Сервис/Защита/Мастер…*

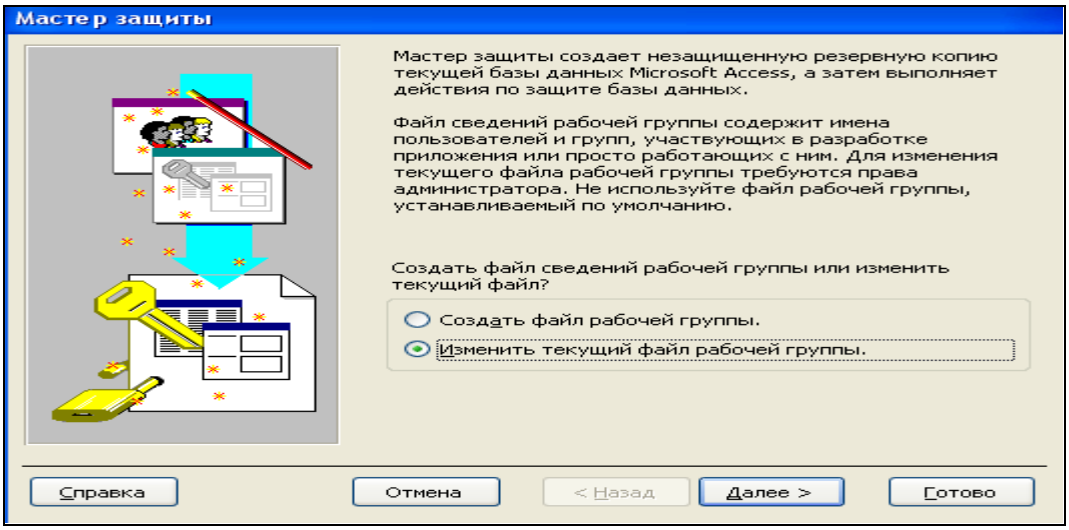

Рисунок 21 – Первый шаг работы мастера защиты

- оставить включенным переключатель **Изменить текущий файл рабочей группы** и кнопкой **Далее** перейти ко второму шагу.
- на втором шаге работы мастера отметить флажками объекты базы данных, которые необходимо защитить (пусть это будут указанные на рисунке 22 таблицы)

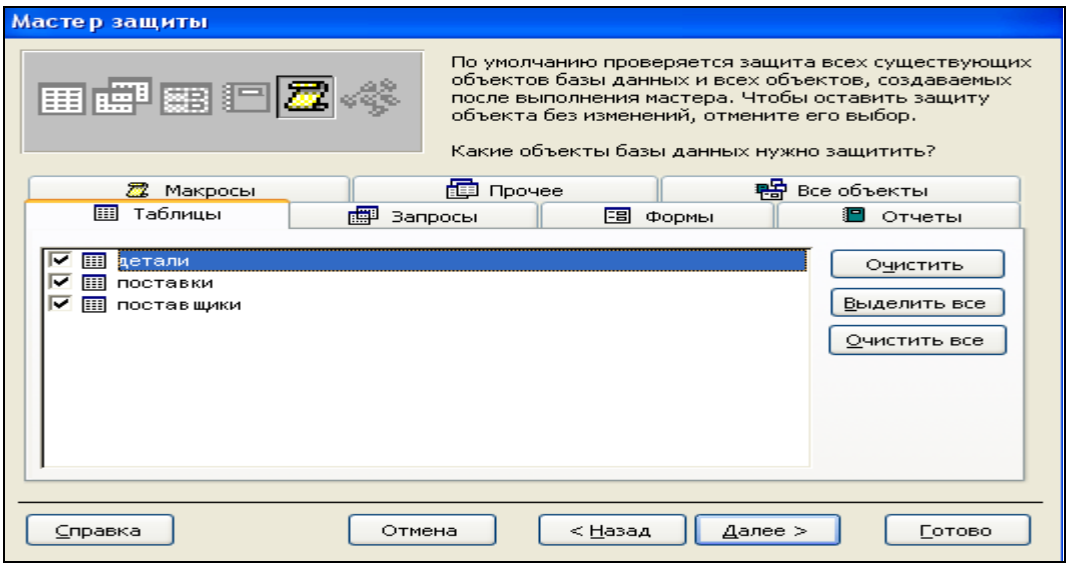

Рисунок 22 – Второй шаг мастера защиты

- на третьем шаге работы мастера нужно указать группы, которые нужно включить в фай рабочей группы (пусть это будут группы, отмеченные флажками на рисунке 23; код группы можно оставить такой, какой предлагает мастер, а можно изменить на более короткий, как это сделано для группы **Новые данные**)

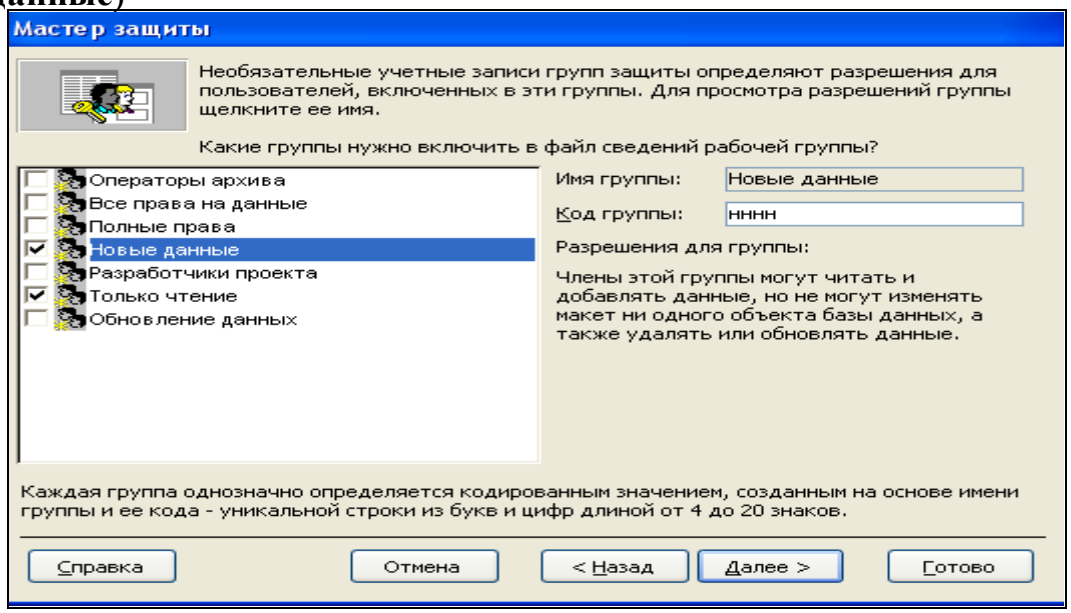

Рисунок 23 – Третий шаг мастера защиты

- на следующем шаге работы мастера не дадим пользователям группы Users никаких разрешений, как предлагает мастер;
- на следующем шаге работы мастера добавим пользователей с указанием паролей для них (Ивана с паролем **иииии**, Сергея с паролем **сссс**, Машу с паролем **ммммм** и Юлю с паролем **ююююю**); результат показан на рисунке 24. Пользователь **админ** из группы Admins добавлен автоматически.

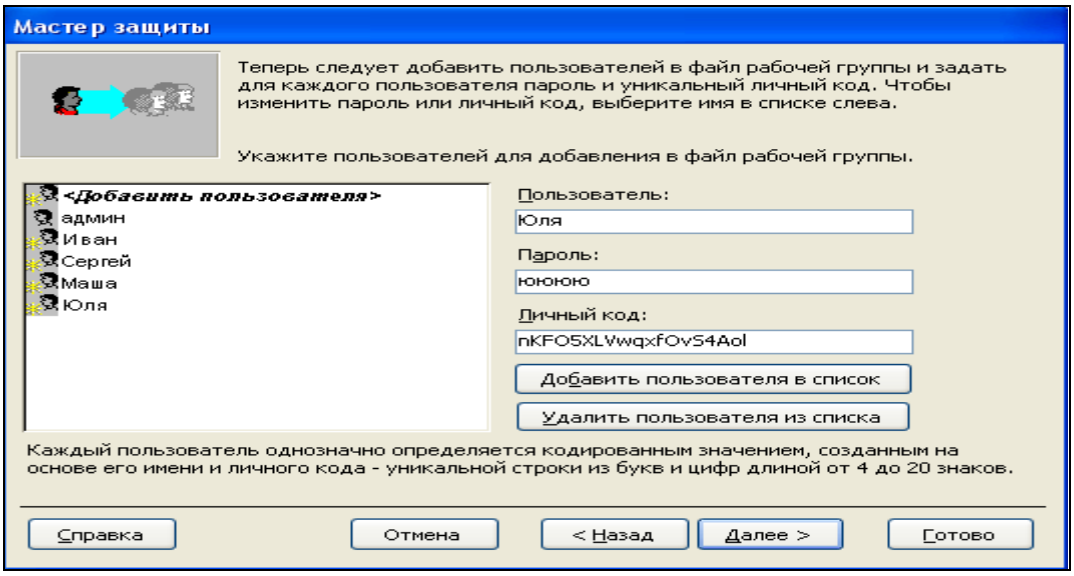

Рисунок 24 – Добавление пользователей в список

- на следующем шаге работы мастера выбираем каждого пользователя из списка **Пользователь или группа** и с помощью флажка включаем его в нужную группу (например, Ивана и Сергея – в группу **Новые данные** (могут читать и добавлять в таблицы данные), а Машу и Юлю – в группу **Только чтение**, пользователь admin автоматически включен в группу Admins).
- на последнем шаге мастер создает назащищенную копию базы данных – файл Моя БД.bak. Это делается для того, чтобы можно было восстановить администратору первоначальные установки или снять защиту;
- зададим пароль пользователю admin командой

⇒ *Сервис/Защита/Пользователи и группы/*вкладка ***Смена пароля***.

В результате работы мастера база данных оказывается защищенной. Открыть ее могут только пользователи из заданного мастеру списка с указанием имени и пароля пользователя, а также пользователь **admin**, который имеет права администратора. Так как пользователям из группы Users не давалось никаких разрешений, то никакой другой пользователь не сможет открыть базу данных.

1.2.3. Выдача разрешений на столбцы таблицы

Выдать разрешения на отдельные столбцы таблицы можно, используя запросы. При этом можно предоставить пользователю возможность добавлять или удалять записи, даже если он не имеет прав на чтение данных из таблицы. Например, возможна ситуация, когда в базе данных сотрудников нужно от пользователей базы скрыть информацию о доходах сотрудников. Как это можно делать, рассмотрим на примере.

Пример 2. Предположим, что в защищенной базе данных **Моя БД** (см. пример1) содержится таблица **детали**, в которой указана цена каждой детали. Пусть пользователю **Маша** надо запретить доступ к полю **цена**, а к остальным полям этой таблицы разрешить доступ. Для решения этой задачи нужно выполнить следующие действия:

- открыть базу данных **Моя БД** с правами администратора;

- создать запрос, который будет выбирать все поля из таблицы детали, кроме поля с ценой детали

SELECT детали.номерд, детали.имяд

FROM детали;

- отобрать права у Маши на любые действия с таблицей детали;
- оставить у Маши права на открытие и монопольный доступ к базе данных, а также на другие таблицы;
- дать права Маше читать или полные права на только что созданный запрос;
- открыть запрос в режиме конструктора;
- открыть окно свойств запроса и установить значение "владельца" свойству "При запуске предоставляются права" и закрыть запрос;
- сметить владельца. Сделать владельцем таблиц Машу, а владельцем запроса - пользователя Admin;
- **не забыть удалить все права у группы Users**, поскольку по умолчанию этой группе даны все права;
- если база данных открывалась для администратора Admin без пароля, то установить для Admin пароль, как было сказано в примере 1;
- закрыть базу данных, затем снова открыть ее от имени Маши и убедиться, что Маша не может читать данные из таблицы детали, но может просматривать данные без цены из этой таблицы через запрос;
- установить для Маши непустой пароль.

1.2.4. Снятие защиты

В некоторых случаях бывает необходимо снять защиту базы данных. Если перед установкой защиты была создана резервная копия базы данных, это делается легко. В простейшем случае для снятия защиты нужно выполнить следующую последовательность действий:

- зарегистрируйтесь в системе как член группы Admins (например, можно использовать учетную запись пользователя **админ**, созданного в предыдущем разделе);
- дайте пользователю Admin права администратора, включив его обратно в группу Admins;
- выдайте группе Users полные права доступа ко все объектам базы данных;
- сделайте владельцем базы пользователя Admin;
- закройте и заново запустите MS Access;
- зарегистрируйтесь в системе как пользователь Admin (в окне входа замените **админ** на Admin);
- задайте пользователю Admin пустой пароль.

1.3. Создание MDE-файла

Прежде чем создавать файл с расширением .MDE, рекомендуется создать резервную копию базы данных, поскольку возврат от MDE-файла к MDB-файлу невозможен.

Чем же отличается MDE-файл от MDB-файла?

MDE-файл является базой данных, в которой все программы, созданные в MDB-файле, сохранены в скомпилированном виде. Поэтому просмотр и редактирование исходных кодов невозможен. Кроме того, изменение таких объектов базы данных, как формы, отчеты и модули, также невозможен. Режим конструктора для них оказывается недоступным. Можно вносить изменения только в таблицы и запросы.

Если же какие-то недоступные в MDE-файле объекты потребуют доработки или изменения, то следует вернуться к сохраненному MDB-файлу, сделать необходимые изменения, а затем заново создать MDE-файл. Создать базу данных в виде MDE-файла можно, имея права администратора, так:

- открыть исходную базу данных (MDB-файл);
- исполнить команду меню

⇒ Сервис/Служебные программы/Создать MDE-файл,

указав местоположение MDE-файла.

Файлы . MDE обычно создаются, когда разработчик хочет, чтобы пользователи не имели к исходному коду, а также к объектам интерфейса (формам и отчетам), и вмесе с тем разработчик не заинтересован в реализации полной системы обеспечения защиты на уровне пользователя. Хотя, если такая защита реализована в исходном MDB-файле, то она действует и в MDE-файле.

2 Защита данных на уровне пользователей в Access 2010 и Access 2007

Порядок выполнения работы

1. Получите у преподавателя индивидуальное задание.

2. Откройте свою БД и проверьте возможность выполнения команды

⇒ Файл/Сведения/Пользователи и разрешения для Access 2010

И ***Пользователи и разрешения*** для Access 2007

3. Если нет, то найдите местоположение системной БД System.mdw .

4. Создайте на рабочем столе ярлык для открытия системной базы данных System.mdw :

C:\DocumentsandSettings\...\ApplicationData\Microsoft\System.mdw

5. Откройте базу данных System.mdw (по умолчанию Вы имеете права администратора с пустым паролем).

6. Импортируйте в эту базу объекты своей БД (таблицы, формы,…),необязятельно все, а только те, на которые Вы предполагаете дать разрешения. Для этого надо на ленте активизировать вкладку **Внешние данные**, в которой выбрать **Access**. В открывшемся окне с помощью кнопки **Обзор** найти свою БД, в которой выбрать импортируемые объекты и завершить импорт кнопкой **ОК**.

7. Создайте группы пользователей и отдельных пользователей согласно индивидуального задания(***⇒ Файл/Сведения/Пользователи и разрешения***).

Дайте группам и отдельным пользователям разрешения на доступ к объектам в соответствии с заданием. Кроме того, дайте группе Users права на открытие БД и отберите все права на доступ к защищаемым объектам (**!!!!**).

8. Замените пустой пароль администратора на любой непустой (запомните его**!!!!**).

9. Закройте базу данных.

10. Снова откройте базу System.mdw. В открывшемся окне замените слово **Admin** на имя пользователя (он может входить в какую-либо группу, и тогда он наследует разрешения, данные группе; а может не входить ни в какую группу, тогда Вы ему дали индивидуальные разрешения). При этом по умолчанию откроется база данных от имени этого пользователя с пустым паролем.

11. Замените пустой пароль пользователя на непустой (запомните его**!!!!**).

12. Закройте базу.

13. Снова откройте базу System.mdw от имени пользователя с непустым паролем и проверьтедействия выданных разрешений. Покажите результаты преподавателю и закройте базу.

14. Верните системную базу данных в исходное состояние.Для этого надо открыть базу данных System.mdw с импортированными объектами от имени **Admin**. Далее выполните команду ⇒ ***Файл/Сведения/Пользователи и разрешения*** и удалить сначала всех созданных Вами пользователей, затем удалить созданные Вами группы. Далее следует изменить пароль для администратора на пустой пароль и удалить импортированные объекты. Показать результат преподавателю.

15. Закрыть базу.

Порядок выполнения работы

1. Получить задание у преподавателя (дерево пользователей и разрешения).
2. Обеспечить защиту на уровне пользователей через интерфейс Access.
3. Обеспечить защиту на уровне пользователей с помощью мастера.
4. Создать MDE-файл и убедиться в его возможностях по защите данных.

Литература

1. Смирнов С.Н. Безопасность систем баз данных. – М.: Гелиос АРВ, 2007. – 352 с.

2. Т.О'Брайен, Подж С., Уайт Дж. Microsoft Access 97: разработка приложений; пер. с англ. – СПб.: БХВ – Санкт-Петербург, 1999. – 640 с.

3. Литвин П., Гетц К., Гунделой М. Разработка корпоративных приложений в Access 2002. Для профессионалов. – СПб.: Питер; Киев: BHV, 2003. – 848 с.