Администрирование и настройка политики безопасности сервера реляционной базы данных MySQL

Мошак Николай Николаевич

д.т.н., доцент, профессор Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича, Большевиков прю, 22, г. Санкт-Петербург, 193323, Россия, nnmoshak49@mail.ru

д.т.н., доцент, профессор Санкт-Петербургсккого государственного университета аэрокосмического приборостроения, ул. Большая Морская, 67, лит А, г. Санкт-Петербург, 190000, Россия, nnmoshak49@mail.ru

Рудинская Сабина Романовна

старший преподаватель, Учреждения образования «Белорусская государственная академия связи», Республика Беларусь, ул. Ф. Скорины, 8/2, г. Минск, 220114, Республика Беларусь, email: sabina.rudin@mail.ru

Груздев Алексей Андреевич

студент Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича, пр. Большевиков, д.22, корп. 1, г. Санкт-Петербург, 193232, Россия, gruzdev.a.a26@mail.ru

АННОТАЦИЯ

Введение. Предлагается методика администрирования и настройки политики безопасности сервера реляционной базы данных MySQL с учетом требований политики информационной безопасности организации и регулятора. **Цель исследования**: создать и апробировать способ администрирования, который может быть использован для реализации требований политики информационной безопасности организации и регулятора, а также в учебном процессе высших учебных заведений при подготовке студентов соответствующего направления. **Методика проведения исследования**: использован подход, предполагающий выполнение каждого этапа исследования пошагово, с детальным описанием каждого шага и представлением результатов в виде визуальных материалов. Это позволяет четко и наглядно представить промежуточные и конечные результаты исследования. **Полученные результаты:** предложена методика установки и администрирование SQL-сервера на примере сервера MySQL, а также настройки параметров безопасности сервера базы данных на примере требований информационной безопасности регулятора к автоматизированным системам класса защищенности 1Б. Рассмотрены все этапы взаимодействия с MySQL: установка, настройка, создание пользователя и базы данных, а также выполнение CRUD (Create, Read, Update, Delete) запросов к ней. Выполнены настройки параметров безопасности сервера базы данных с учетом указанных требований информационной безопасности регулятора. Показано, что настройки доступа непривилегированного пользователя к серверу с использованием утилиты «MySQL Workbench» значительно повышает безопасность системы, поскольку предполагает проверку его прав доступа через командную строку и аутентификацию по паролю в соответствии с требованиями информационной безопасности регулятора к автоматизированным системам класса защищенности 1Б, что устраняет угрозу несанкционированного доступа к базе данных.

КЛЮЧЕВЫЕ СЛОВА: база данных MySQL; политика безопасности MySQL; администрирование MySQL; информационная безопасность; создание и настройка сервера.

Введение

Защита информации – комплекс мероприятий, направленных на обеспечение информационной безопасности (ИБ) организации, основывающейся на законодательстве Российской Федерации, Доктрине информационной безопасности Российской Федерации, утв. Президентом РФ 09.09.2000 г., Федеральных законов РФ в области ИБ и других государственных нормативно-методических документах в области ИБ. 1 мая 2022 года Президентом РФ был подписан указ № 250, направленный на обеспечение информационной безопасности ряда ключевых организаций России. Основной целью безопасности информационной системы (ИС) организации – является понижение размера вероятного ущерба ценных активов до допустимых значений, а также на дежная и качественная эксплуатация ИС в условиях возникающих угроз. Защищенность ИС достигается проведением руководством организации соответствующей политики информационной безопасности (далее – Политика ИБ) и управления информационной безопасностью. Поэтому после определения официальной Политики ИБ следует определить конкретные защитные меры и средства, а также меры контроля, реализующие практические процедуры защиты, которые определяют, как именно выполнять и контролировать требования Политики на информационных системах. Под защищенностью понимается состояние информации, при котором становится невозможным или затруднено воздействия случайного или преднамеренного характера, влияющие на конфиденциальность, целостность и доступность информационных активов и инфраструктуры организации. Анализ уровня защищенности ИС – это анализ реализованных мер защиты информации, который позволит определить степень соответствия требованиям основных нормативно-правовых актов по ИБ, а также оценить реальный уровень защищенности ИС организации от возможных угроз [1-4].

Практическая реализация анализа защищенности ИС является неотъемлемым этапом проведения аудита безопасности ИС и базируется на полученных в процессе указанного аудита количественных и качественных оценках текущего состояния защищенности ИС в организации. ГОСТ Р ИСО 19011-2012 устанавливает концепцию проведения аудита систем менеджмента, включая аудит безопасности информации. Основные этапы и методы работ по проведению аудита компьютерной безопасности, описанные в [5-15]. Сравнительный анализ методик аудита ИБ ИС приведен в [16].

Анализ защищенности ИС должен проводиться: в процессе разработки Политики ИБ и в процессе эксплуатации ИС в рамках осуществления функций системы управления ИБ. При проведении анализа защищенности реализуются две стратегии. Первая – пассивная, реализуемая на уровне операционной системы, системы управления базой данных (СУБД) и приложений, при которой осуществляется анализ конфигурационных файлов и системного реестра на наличие неправильных параметров; файлов паролей на наличие легко угадываемых паролей, а также других системных объектов на нарушения политики безопасности. Вторая стратегия – активная, осуществляемая в большинстве случаев на сетевом уровне, позволяющая воспроизводить наиболее распространенные сценарии атак и анализировать реакции системы на эти сценарии. В настоящее время, видимо, не существует каких-либо стандартизированных методик анализа защищенности ИС. Поэтому в конкретных ситуациях алгоритмы действий аудиторов могут существенно различаться. Типовая методика состоит в следующем:

- изучение исходных данных по ИС;

- оценка рисков, связанных с осуществлением угроз безопасности в отношении ресурсов ИС;
- анализ механизмов безопасности организационного уровня, политики безопасности организации и организационно-распорядительной документации по обеспечению режима информационной безопасности и оценка их соответствия требованиям существующих нормативных документов, а также их адекватности существующим рискам;
- ручной анализ конфигурационных файлов маршрутизаторов, межсетевых экранов и прокси-серверов, осуществляющих управление межсетевыми взаимодействиями, почтовых и DNS серверов, а также других критических элементов сетевой инфраструктуры;
- сканирование внешних сетевых адресов «открытого» контуров локальной вычислительной сети (ЛВС) из сети Интернет;
- сканирование ресурсов ЛВС изнутри;
- анализ конфигурации серверов и автоматизированных рабочих мест (АРМ) «закрытого» и «открытого» контуров ЛВС при помощи специализированных программных агентов.

Защита АРМ и серверов ИС является неотъемлемой частью построения ее системы безопасности. В [17] предложен способ настройки изолированной программной среды АРМ пользователя средствами операционной системы Windows для защиты информации пользователя от несанкционированного доступа. В статье предлагается методика администрирования и настройки политики безопасности сервера реляционной базы данных MySQL «закрытого» контура ИС, а также настройки параметров безопасности сервера базы данных на примере требований ИБ ФСТЭК к автоматизированным системам класса защищенности 1Б. Рассмотрены все этапы взаимодействия с MySQL: установка, настройка, создание пользователя и базы данных, а также выполнение CRUD (Create, Read, Update, Delete) запросов к ней. Выполнены настройки параметров безопасности сервера базы данных с учетом указанных требований ИБ регулятора. Показано, что настройки доступа непривилегированного пользователя к серверу с использованием утилиты «MySQL Workbench» значительно повышает безопасность системы, поскольку предполагает проверку его прав доступа через командную строку и аутентификацию по паролю, что устраняет угрозу несанкционированного доступа к базе данных.

Установка *MySQL* 8.0.18 на платформу операционной системы Windows

Установка *MySQL* возможна на операционную систему (ОС) *Windows* 2000 и выше. Дистрибутив *MySQL* можно скачать с сайта (URL: https://www.mysql.com/). Онлайн-документация (на английском языке) расположена по адресу (URL: http://dev.mysql.com/doc/). Для установки необходимо иметь права администратора. Перед скачиванием необходимо установить *Microsoft. Net Framework* 4.0, использующийся в программах администрирования в среде *Microsoft Windows*. Для других операционных систем используются другие методы администрирования. Непосредственно процесс установки производится аналогично другим программам в *Windows*.

Перейдём на сайт *MySQL* и загрузим любой из *MSI*-установщиков, один автономный, другой – использует подключение к сети Интернет (рис. 1).

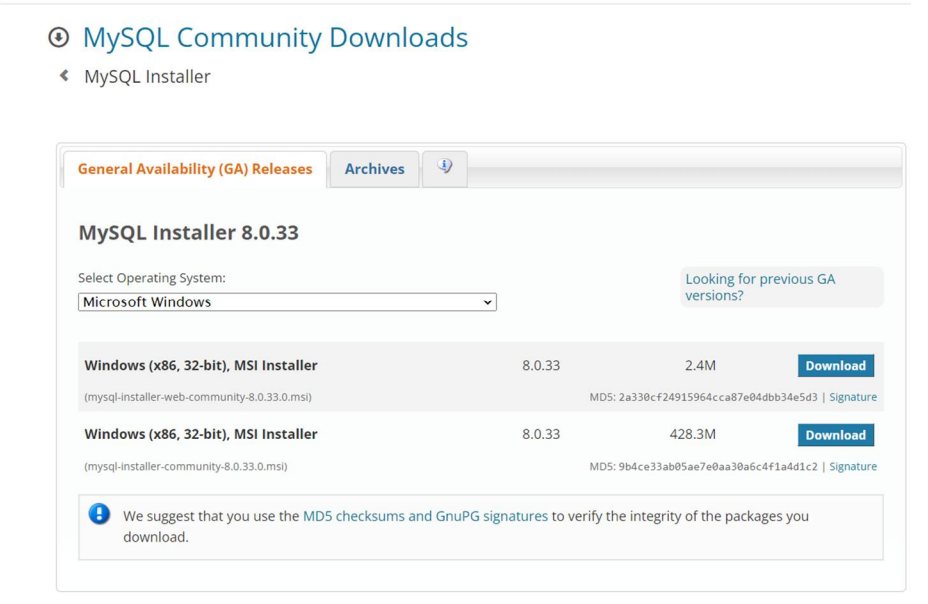

Рис. 1. Сайт загрузки *MySQL*

При выборе типа устанавливаемого сервера рекомендуется выбрать пользовательскую установку «*Custom*» (рис. 2).

Примечание 1. Допускается выбрать полную установку, но будут установлены неиспользуемые компоненты и потребуется дополнительно установить библиотеку *Visual C++ 2010 Runtime*, не входящую в дистрибутив.

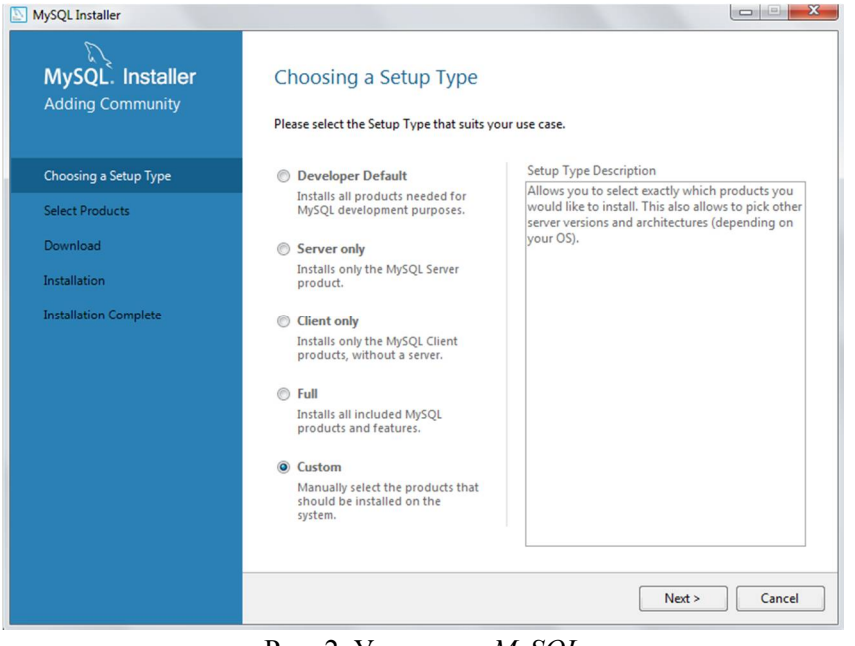

Рис. 2. Установка *MySQL*

При выборе типа установки «*Custom*» будут установлены только необходимые компоненты *MySQL* (рис. 3).

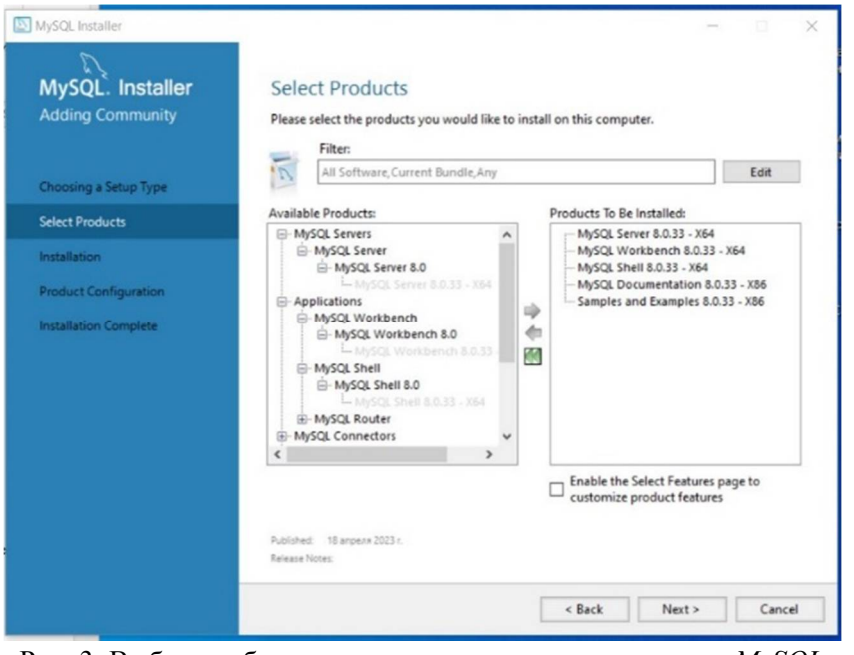

Рис. 3. Выбор необходимых компонентов для установки *MySQL*

Примечание 2. Обязательно должны быть установлены следующие компоненты:

- «*MySQL Server*» 8.0.33 – x64, в который входят *MySQL Server*, *Client Programs*;
- В группе «*MySQL Server* 5.6.11» – *MySQL Server*, *Client Programs*, *Server Data Files*;
- В группе *Applications* – *MySQL Notifier*;
- Группу *Documentation* – рекомендуется установить полностью.

После выбора типа установки кнопкой «*Next* >» запустим установку *MySQL* (рис. 4).

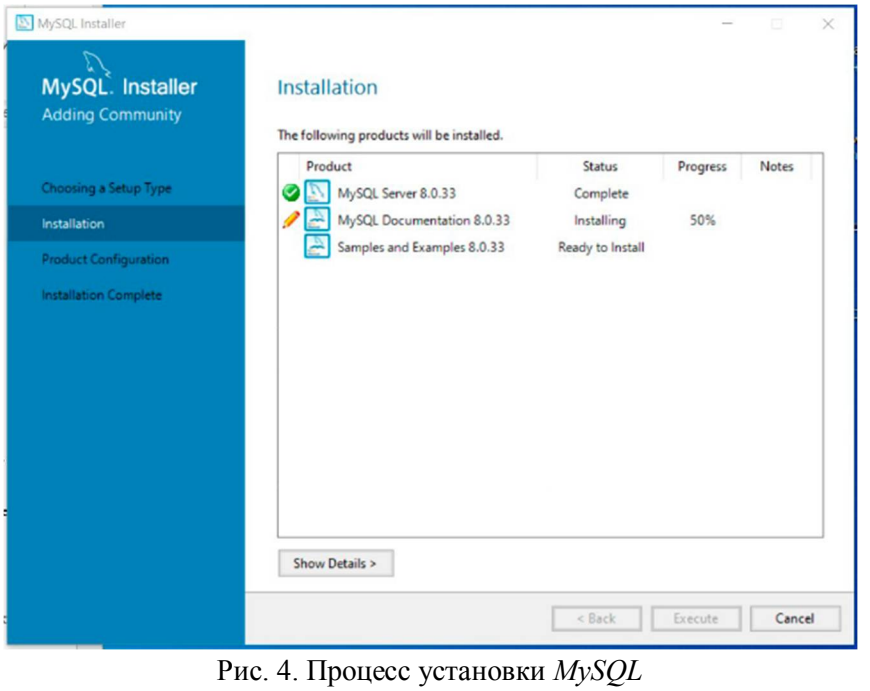

Рис. 4. Процесс установки *MySQL*

По окончании установки запускается мастер настроек *MySQL* (он также доступен пользователю и после инсталляции). В окне настройки серверной части рекомендуется выбрать конфигурацию «*Development Machine*» (рис. 5).

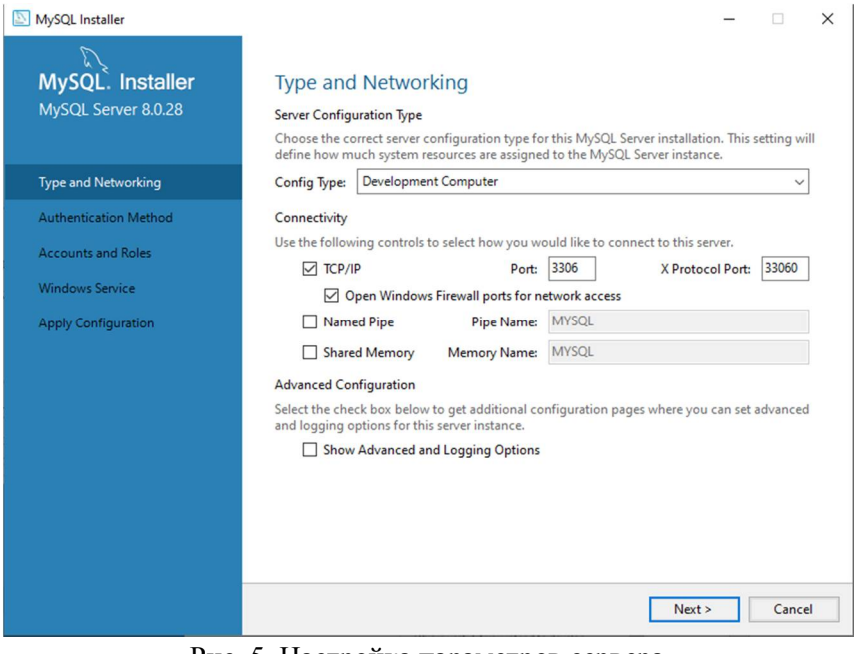

Рис. 5. Настройка параметров сервера

Далее выберем рекомендованный метод аутентификации (рис. 6) и переходим к настройке пароля главного администратора сервера (учетная запись – *root*) (рис. 7). Пользователь с правами доступа *root* имеют полный доступ ко всем базам данных и таблицам внутри этих баз.

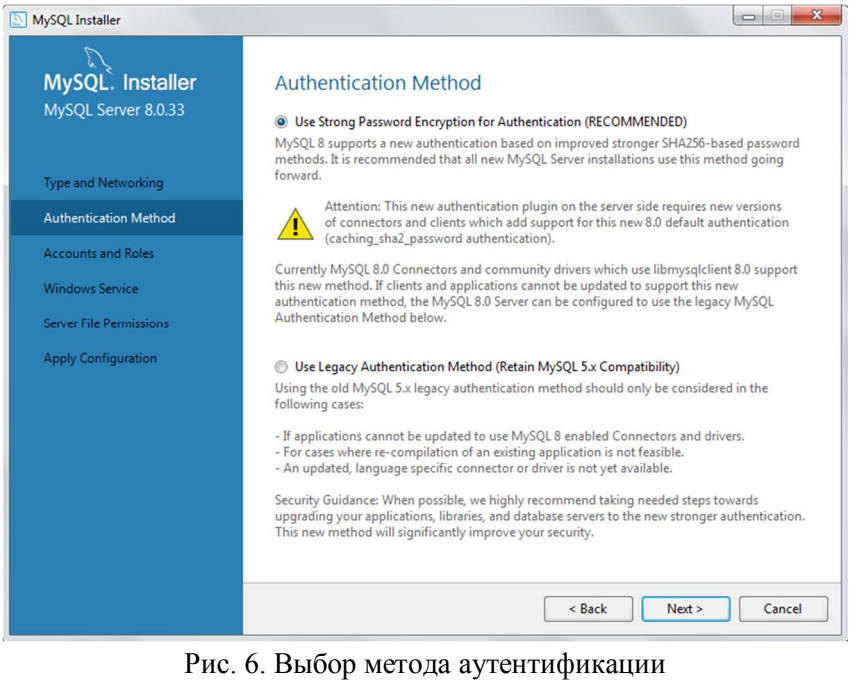

Рис. 6. Выбор метода аутентификации

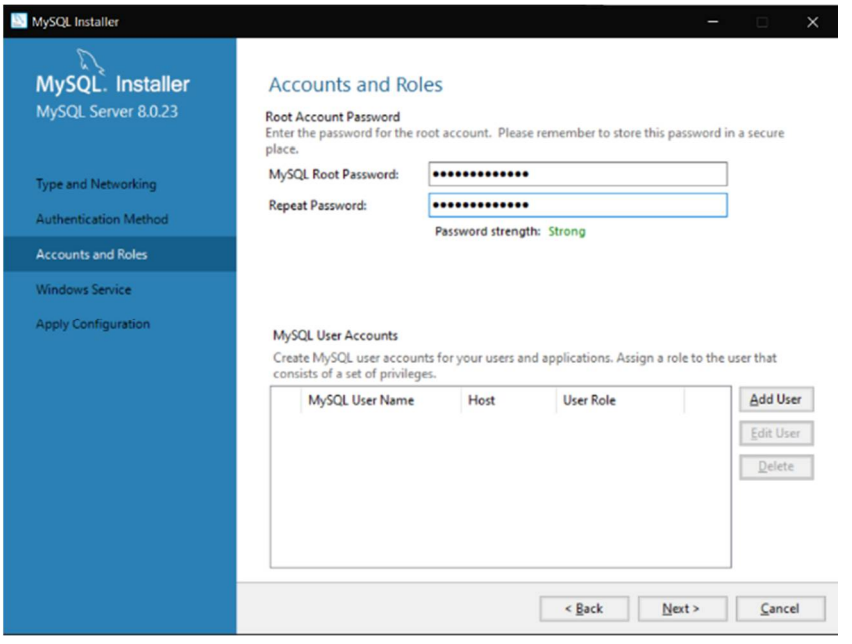

Рис. 7. Настройка пароля главного администратора

Так как сервер установлен в «закрытом» контуре ИС, то в зависимости от уровня конфиденциальности обрабатываемой информации, необходимо выбрать пароль администратора соответствующей сложности для защиты доступа к базам данных (рис. 6). В соответствии с требованиями информационной безопасности ФСТЭК, например, класса защищенности 1Б (Руководящий документ ФСТЭК «Автоматизированные системы. Защита от несанкционированного доступа к информации Классификация автоматизированных систем и требования по защите информации»):

- осуществляется идентификация и проверка подлинности субъектов доступа при входе в систему по паролю условно-постоянного действия, длиной не менее шести буквенно-цифровых символов;
- осуществляется регистрация входа (выхода) субъектов доступа в систему (из системы), либо регистрация загрузки и инициализации операционной системы и ее программного останова. Регистрация выхода из системы или останова не проводится в моменты аппаратурного отключения автоматизированной системы (АС).

При этом, пароль должен содержать определенное количество символов, в т. ч. и специальных символов, букв разного регистра и цифр. Установим пароль вида *P@ssword4321*.

Также имеется возможность добавить дополнительных пользователей, установив им необходимый уровень доступа, о чем речь пойдет ниже.

Окно настройка запуска сервиса можно оставить без изменений (рис. 8).

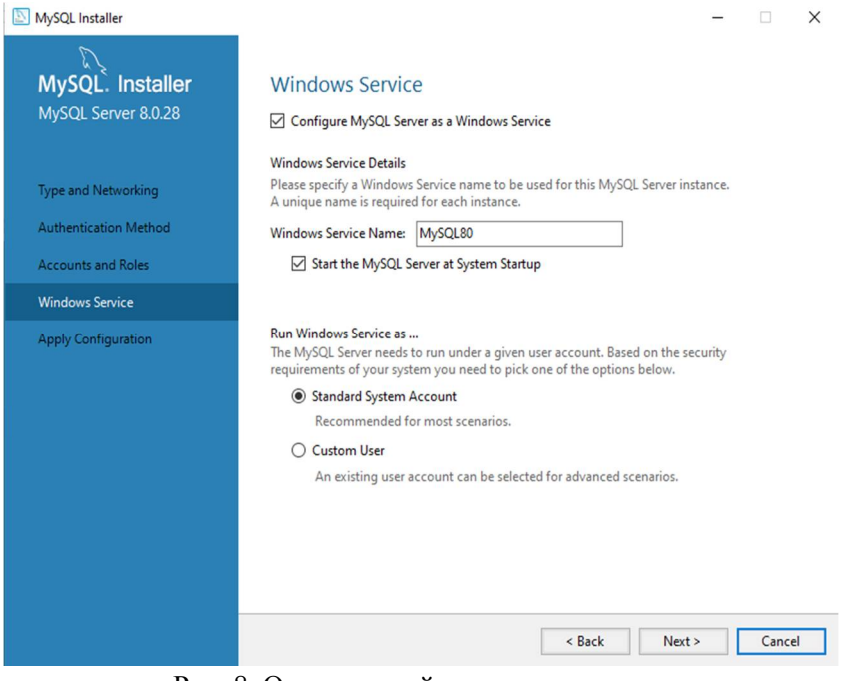

Рис. 8. Окно настройка запуска сервиса

На следующем этапе производится автоматическая настройка сервера в соответствии с заданной конфигурацией и его запуск. Окно автоматической настройки сервера в соответствии с заданной конфигурацией применяет все раннее указанные настройки на сервере. В результате выполнения отображаются этапы настроек и создается журнал их применения. Все настройки применены успешно. На этом установка и предварительная настройка сервера «*MySQL Server*» завершена (рис. 9).

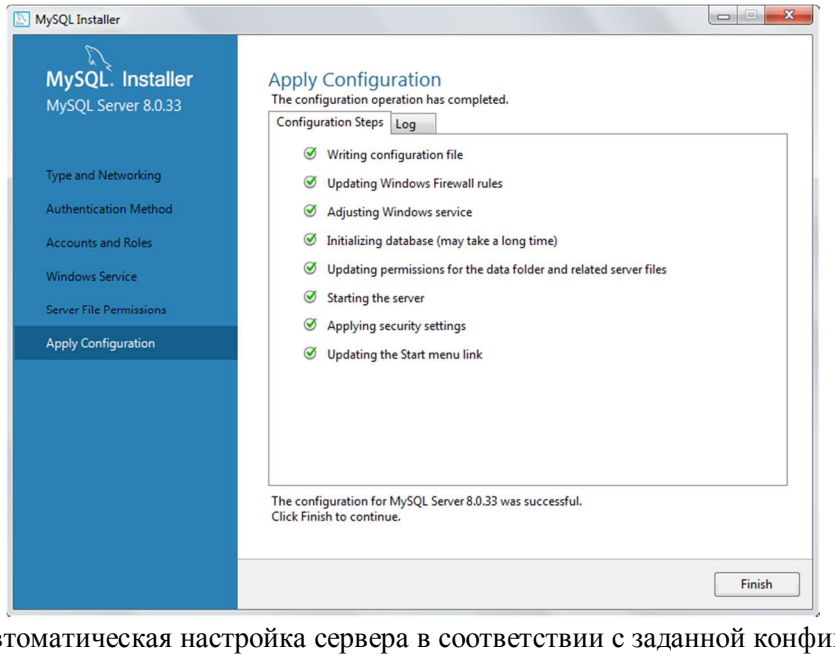

Рис. 1. Автоматическая настройка сервера в соответствии с заданной конфигурацией

Для дальнейшего управления сервером используется утилита «*MySQL Notifier*» (вызывается из меню программ). Утилита выводит иконку в панели задач, являющуюся индикатором состояния сервера базы данных, а также позволяющая управлять сервером (запуск и остановка сервера). На этом установка и предварительная настройка сервера «*MySQL Server*» завершена, и он готов к работе.

На (рис.10) показан результат проверки пароля при подключении к локальному серверу базы данных.

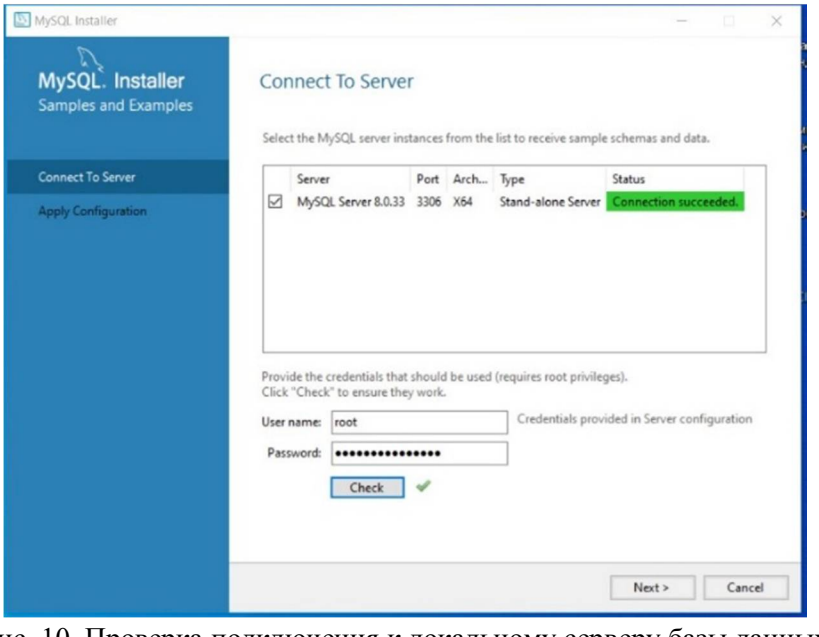

Рис. 10. Проверка подключения к локальному серверу базы данных

Создание новой базы данных

Итак, у нас имеется настроенный сервер «*MySQL Server*» и утилита для взаимодействия с ним «*MySQL Notifier*». Для того, чтобы можно было использовать возможности сервера, такие как создание и назначение прав пользователям на таблицы и базы данных, требуется, собственно, сначала создать пользовательскую базу данных и таблицу в ней. Имеющиеся по умолчанию базы данных можно посмотреть с использованием команды: *show databases;*

Для создания новой базы данных откроем утилиту *MySQL Command Line Client* и зайдем под *root*-пользователем, для этого запустим консоль (*win+r >>cmd>> mysql -uroot -p>> P@ssword4321*).

Создание базы данных выполняется с помощью команды *CREATE DATABASE based*. Синтаксис команды: *CREATE DATABASE database_name;* где *database_name* – Имя, которое будет присвоено создаваемой базе данных.

Создадим новую базу данных «*BD_Telephone_directory*» командой (рис. 11):

CREATE DATABASE BD_Telephone_directory;

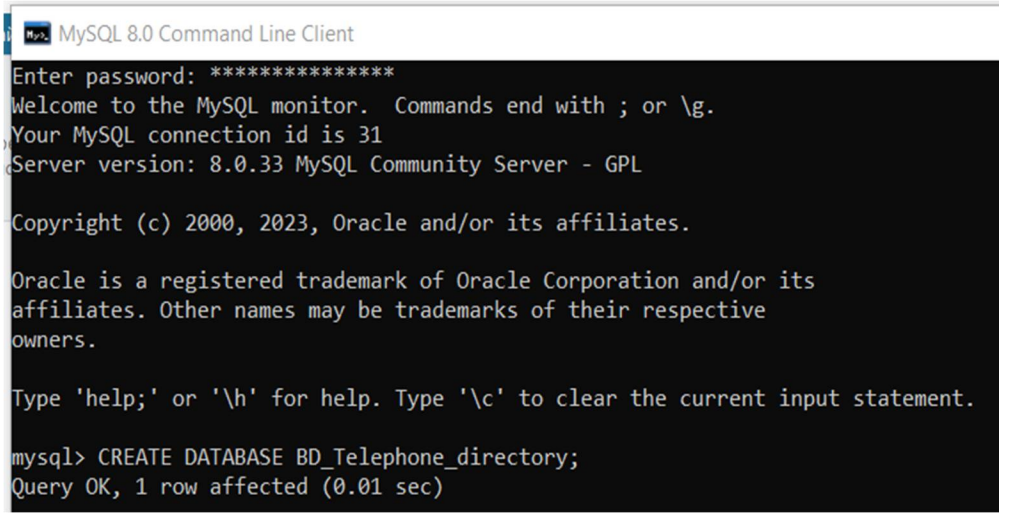

Рис. 11. Команда создания базы данных

Проверим, что база данных создалась командой: *show databases* (рис. 12).

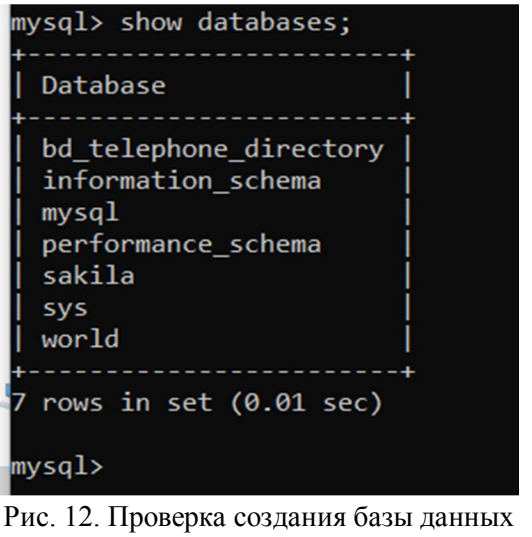

Рис. 12. Проверка создания базы данных

Для того чтобы начать работать с базой данных, необходимо её выбрать. Для этого необходимо прописать команду «*use database_name;*».

Выберем созданную базу данных и укажем серверу, что далее мы будем работать именно с ней, используя следующую команду (рис. 13):

use BD_Telephone_Directory;

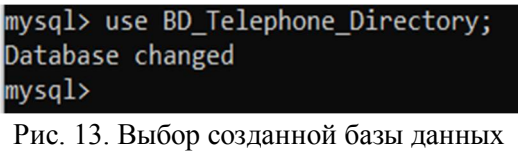

Рис. 13. Выбор созданной базы данных

Создание нового пользователя и настройка неограниченных прав его доступа в MySQL

Одним из основных этапов администрировании баз данных является настройка прав пользователя по доступу и работе с базой данных. От этого зависит целостность базы данных, связанных с ней таблиц и, соответственно, информации, что в них содержится. Для работы пользователя с базой данных ему должны быть назначены определенные права по доступу в соответствии с его должностными обязанностями и принятой Политикой. Основой системы безопасности является система привилегий (*privilege system*), позволяющая гибко управлять правами доступа как к управлению сервером, так и к отдельным базам, таблицам и полям таблиц, а также встроенным функциям и хранимым процедурам. Для изучения всей системы привилегий рекомендуется обратиться к документации на программу, здесь же рассмотрим основные моменты. Настройку параметров доступа можно производить из клиента базы данных – программы *mysql.exe* (находится в каталоге установки продукта в подкаталоге *bin*). При запуске программе можно указать много параметров, но для изучения достаточно использовать синтаксис:

- *mysql.exe -u* <имя пользователя> *-p* <база данных>;
- *-u* – флаг, за которым через пробел указывается имя пользователя;
- *-p* – флаг, указывающий на необходимость запроса пароля;
- <*база данных*> – имя базы, с таблицами которой будет проводиться работа. Этот параметр не является обязательным, т. к. из клиента в любой момент можно переключиться на работу с другой базой с использованием команды «*use* <база данных>».

Все команды, вводимые в клиенте, обязательно должны заканчиваться точкой с запятой.

Все параметры безопасности *MySQL* хранятся в виде таблиц системной базы данных «*mysql*», поэтому первый запуск клиента рекомендуется осуществить командой *mysql.exe –u root –p mysql* (пароль был установлен в процессе инсталляции).

Для создания нового пользователя следует выполнить следующие шаги:

1) Создать пользователя *MySQL* и предоставить неограниченные права доступа;
2) Назначить специальные права доступа для пользователя *MySQL*.

Таким образом, после первоначальной настройки сервера *MySQL*, будет предоставлено имя пользователя и пароль, причем эти начальные учётные данные дают привилегии «root-доступа», то есть привилегированного пользователя. Пользователь с правами доступа root имеют полный доступ ко всем базам данных и таблицам внутри этих баз – но на предприятиях обычно требуется предоставить доступ к базе данных для сотрудников, которым не требуется и не рекомендовано полное управление. Для создания нового пользователя в *MySQL* необходимо использовать следующую команду:

CREATE USER 'username'@'localhost' IDENTIFIED BY 'password';

Здесь *'username'* – это имя нового пользователя, которое вы выбираете, и *'password'* – это пароль для нового пользователя. В данном примере *localhost* означает, что пользователю будет доступен только локальный вход. Вы можете заменить *'localhost'* на другой хост, если хотите разрешить доступ к пользователю с других автоматизированных рабочих мест (АРМ). Можно предоставить пользователю возможность войти в свою учетную запись с любого АРМ. Однако, в этом случае требуется настраивать дополнительные политики безопасности на каждом АРМ, с которого можно осуществить работу. В нашем случае

достаточно будет настроить локальные политики безопасности и ограничить пользователю доступ в Интернет только одним *IP* адресом – тем, на котором установлен сервер *MySQL*.

Создадим нового пользователя «*New_User1*» выбранной базы данных с паролем *«qwertypass»* с учетом требований информационной безопасности ФСТЭК к АС класса защищенности 1Б [11], выполнив команду (рис. 14):

CREATE USER 'New_User1'@'localhost' IDENTIFIED BY 'qwertypass';

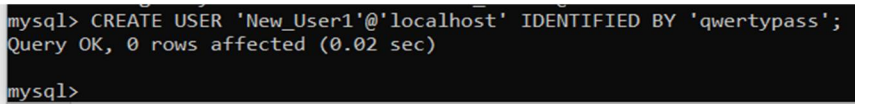

Рис. 14. Команда создания нового пользователя

Назначим вновь созданному пользователю неограниченные права доступа к созданной базе данных *BD_Telephone_Directory*, выполнив команду (рис. 15):

GRANT ALL PRIVILEGES ON BD_Telephone_Directory. TO 'New_User1'@'localhost';*

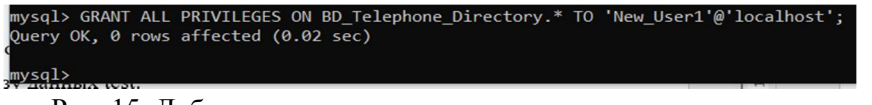

Рис. 15. Добавление прав доступа созданному пользователю

Проверим назначенные права доступа для пользователя командой (рис. 16):

SHOW GRANTS FOR 'New_User1'@'localhost';

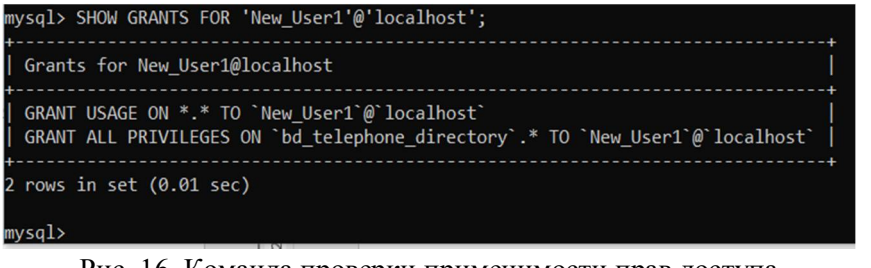

Рис. 16. Команда проверки применимости прав доступа

Создание таблицы в установленной базе данных

Создадим пользовательскую таблицу с названием «*Phone_Numbers*» телефонного справочника (рис. 20) в базе данных «*BD_Telephone_Directory*» с заданными параметрами: столбцы *UserName* – тип данных *Text*, *UserAddress* – тип данных *Text*, *UserPhone* – тип данных *Text*. Также добавим дополнительную колонку, в которой пропишем автоматический счетчик записей *auto_increment primary key* *muna INTEGER* – это значение будет увеличиваться с каждой новой записью и позволит более гибко оперировать содержимым таблицы и облегчит проведение операций с кортежами (строками) в таблице (рис. 17). Для этого используем команду:

CREATE TABLE Phone_Numbers (ID INTEGER auto_increment PRIMARY KEY, UserName TEXT NOT NULL, UserAddress TEXT NOT NULL, UserPhone TEXT NOT NULL);

```
mysql> CREATE TABLE Phone_Numbers (ID INTEGER auto_increment PRIMARY KEY, UserName TE
XT NOT NULL, UserAddress TEXT NOT NULL, UserPhone TEXT NOT NULL);
Query OK, 0 rows affected (0.03 sec)

mysql>
```

Рис. 17. Создание таблицы

Поскольку сейчас таблица пустая, заполним ее произвольными данными – в нашем примере это телефонная книга, поэтому добавим в нее записи, содержащие имя, адрес и телефонный номер абонентов с помощью команды *INSERT* (рис. 18).

```
MySQL 8.0 Command Line Client

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Иван','у
л. Ленина','268-35-63');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Максим',
'Петродворцовый проезд','183-52-62');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Наталья'
,'пл. Мужества','527-52-42');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Петр','у
л. Правды','295-47-37');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Ярослав'
,'Удельный переулок','274-66-45');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Тимофей'
,'ул. Канта','296-38-47');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Phone_Numbers (UserName, UserAddress, UserPhone) VALUES ('Ольга','
пр. Маршала Блюхера','276-86-35');
Query OK, 1 row affected (0.01 sec)

mysql>
```

Рис. 18. Вставка данных в таблицу

Проверим содержимое таблицы командой с использованием ключевого слова *select*: *select * from phoneNumbers;* (рис. 19).

```
mysql> SELECT * FROM Phone_Numbers;
+----+----------+----------------------+-----------+
| ID | UserName | UserAddress          | UserPhone |
+----+----------+----------------------+-----------+
|  1 | Алексей  | наб. р. Волковки     | 434-42-68 |
|  2 | Федор    | пр. Славы            | 289-26-56 |
|  3 | Иван     | ул. Ленина           | 268-35-63 |
|  4 | Максим   | Петродворцовый проезд | 183-52-62 |
|  5 | Наталья  | пл. Мужества         | 527-52-42 |
|  6 | Петр     | ул. Правды           | 295-47-37 |
|  7 | Ярослав  | Удельный переулок    | 274-66-45 |
|  8 | Тимофей  | ул. Канта            | 296-38-47 |
|  9 | Ольга    | пр. Маршала Блюхера  | 276-86-35 |
+----+----------+----------------------+-----------+
9 rows in set (0.00 sec)

mysql>
```

Рис. 19. Проверка введённых данных

Очевидно, что новая таблица *Phone_Numbers* с несколькими записями успешно создана.

Назначение специальных прав доступа для пользователя *MySQL*

Настройка параметров безопасности крайне важна, так как именно они будут определять доступ пользователя к базе данных и осуществлять его идентификацию и аутентификацию. Для начала отменим все ранее установленные неограниченные права пользователя *New_User1* с помощью команды *revoke:*

*REVOKE ALL PRIVILEGES on BD_Telephone_Directory.*FROM "New_User1"@"localhost";*

MySQL позволяет назначать права доступа с помощью команды:

GRANT [тип прав] ON [имя базы данных].[имя таблицы] TO 'имя пользователя'@'тип доступа на сервер';

Нужно заменить значение «тип прав» на тот вид прав доступа, который требуется предоставить новому пользователю. Также нужно указать базу данных и имена таблиц, доступ к которым предоставляется. В *MySQL* есть несколько типов прав доступа, некоторые из них описаны ниже:

- *CREATE* – позволяет пользователям создавать базы данных/таблицы;
- *SELECT* – позволяет пользователям делать выборку данных;
- *INSERT* – позволяет пользователям добавлять новые записи в таблицы;
- *UPDATE* – позволяет пользователям изменять существующие записи в таблицах;
- *DELETE* – позволяет пользователям удалять записи из таблиц;
- *DROP* – позволяет пользователям удалять записи в базе данных/таблицах.

Определим пользователю *New_User1* право использовать *SELECT* на все таблицы внутри баз данных, имеющихся в файловой системе *SQL*, а право на *INSERT* – право на добавление новых записей только в таблицу телефонного справочника *Phone_Numbers* и ограничения на доступ к отдельным полям таблицы устанавливать не нужно. Учитывая требования к информационной безопасности предъявляемое нормативными документами ФСТЭК [11] к информационной системе класса 1Б, права на создание, удаление таблиц, изменение и добавление данных должны быть только у администратора *root*, который отвечает за целостность и безопасность базы данных.

С помощью администратора *root* предоставим нашему непривилегированному пользователю доступ лишь к базовым командам управления базой данных. Проведем назначение прав следующей командой (рис. 20):

GRANT SELECT, INSERT ON BD_Telephone_Directory. TO 'My_User1'@'localhost';*

```
mysql> SHOW GRANTS FOR 'New_User1'@'localhost';
+-----------------------------------------------------------------------------------------------+
| Grants for New_User1@localhost                                                                |
+-----------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `New_User1`@`localhost`                                                 |
| GRANT SELECT, INSERT ON `bd_telephone_directory`.* TO `New_User1`@`localhost`                 |
| GRANT SELECT, INSERT ON `bd_telephone_directory`.`phone_numbers` TO `New_User1`@`localhost` |
+-----------------------------------------------------------------------------------------------+
3 rows in set (0.00 sec)
```

Рис. 20. Назначение определённых прав пользователю.

Проверить применимость прав можно командой:

show grants for 'New_User1'@'localhost';

Применение указанных настроек безопасности ограничивает права доступа пользователя в *MySQL* в соответствии с Политикой организации.

Организация доступа непривилегированного пользователя к серверу *MySQL*

Осуществим подключение к серверу *MySQL* с помощью утилиты «*MySQL Workbench*», предварительно установив ее с помощью той же программы, которой мы устанавливали «*MySQL Server*». Чтобы запустить командную строку под новым пользователем необходимо зайти во вкладку «*MySQL Workbench*» и создать новое подключение, вызвав *Manage Server Connections*, для которого настроить следующие параметры (остальные оставить по умолчанию) (рис. 21):

Connection Name: *New_User1;* Password: *qwertypass*.

Выбор пароля должен удовлетворять требованиям информационной безопасности ФСТЭК к АС класса защищенности 1Б [11].

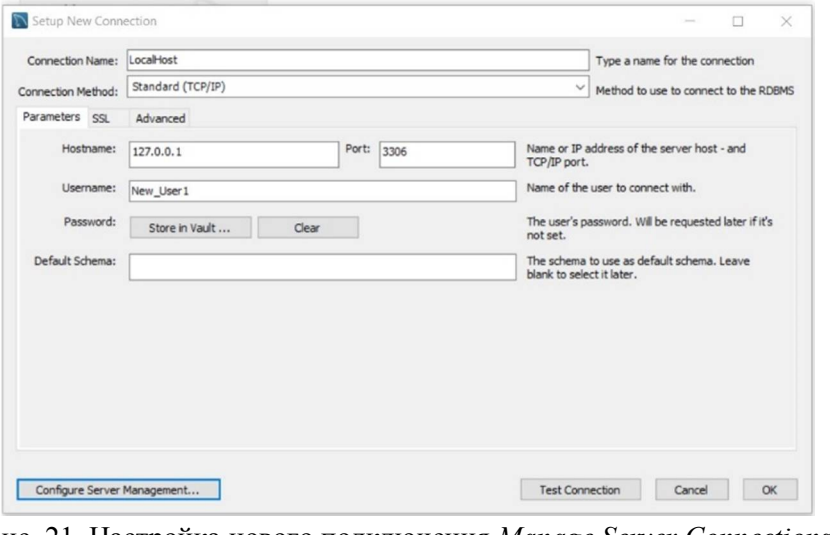

Рис. 21. Настройка нового подключения *Manage Server Connections*

После настройки *Manage Server Connections*, нажмем кнопку «*Test Connection*» и если все прошло успешно, то нажимаем «ОК». После чего настройка нового соединения будет успешно завершена, а в утилите «*MySQL Workbench*» появится новое доступное соединение для созданного не привилегированного пользователя (рис. 22):

Осуществим тестовое подключение к серверу с использованием нового соединения, выполнив следующие действия: *кликнуть на подключение New_User1 правой кнопкой мыши → Start Command Line Client и ввести пароль пользователя «qwertypass».*

Как видно (рис. 23), подключение к серверу успешно установлено и непривилегированный пользователь получил доступ к серверу с помощью утилиты «*MySQL Workbench*».

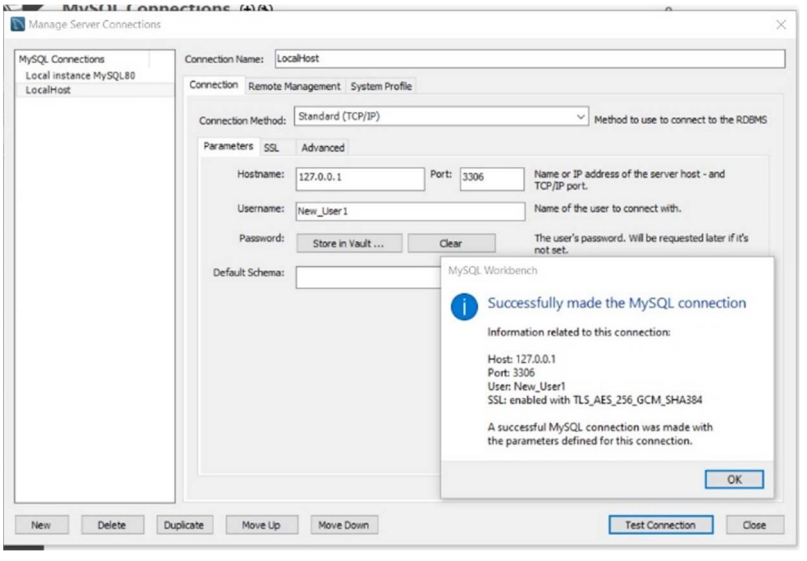

Рис. 22. Новое доступное соединение для непривилегированного пользователя

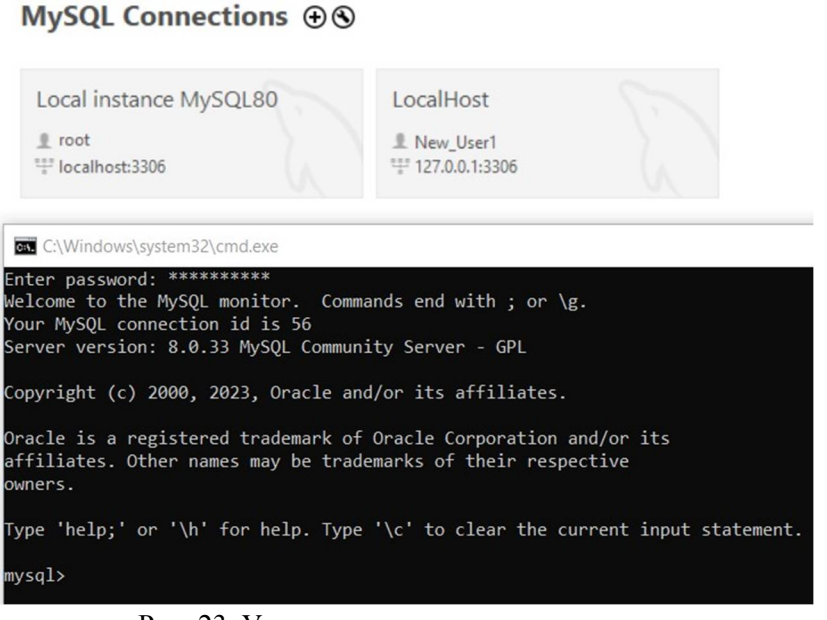

Рис. 23. Успешное подключение к серверу

Настройки доступа непривилегированного пользователя к серверу с использованием утилиты «*MySQL Workbench*» значительно повышает безопасность системы, т. к. предполагает проверку его прав доступа через командную строку и аутентификацию по паролю в соответствии с требованиями информационной безопасности ФСТЭК к АС класса защищенности 1Б [11], что устраняет угрозу несанкционированного доступа к базе данных.

Запрос выборок из таблицы

Для проверки настроенных прав доступа для пользователя «*New_User1*» и выполнения выборок из таблицы «*Phone_Numbers*» для конкретных имён можно использовать следующий запрос (рис. 24):

SELECT UserAddress, UserPhone FROM Phone_Numbers WHERE UserName = 'Максим';
SELECT UserAddress, UserPhone FROM Phone_Numbers WHERE UserName = 'Петр';

```
mysql> SELECT UserAddress, UserPhone FROM Phone_Numbers WHERE UserName = 'Максим';
+-----------------------+-----------+
| UserAddress           | UserPhone |
+-----------------------+-----------+
| Петродворцовый проезд | 183-52-62 |
+-----------------------+-----------+
1 row in set (0.01 sec)

mysql> SELECT UserAddress, UserPhone FROM Phone_Numbers WHERE UserName = 'Петр';'
+-------------+-----------+
| UserAddress | UserPhone |
+-------------+-----------+
| ул. Правды  | 295-47-37 |
+-------------+-----------+
1 row in set (0.00 sec)
```

Рис. 24. Выборка из таблицы «*Phone_Numbers*» по конкретным именам

Выборка всех записей таблицы с сортировкой по полю *UserName* в алфавитном порядке проводится командой, представленной на рис. 25:

*SELECT * FROM Phone_Numbers ORDER BY UserName asc;*

```
mysql> SELECT * FROM Phone_Numbers ORDER BY UserName asc;
+----+----------+-----------------------+------------+
| ID | UserName | UserAddress           | UserPhone  |
+----+----------+-----------------------+------------+
|  1 | Алексей  | наб. р. Волковки      | 434-42-68  |
|  3 | Иван     | ул. Ленина            | 268-35-63  |
|  4 | Максим   | Петродворцовый проезд | 183-52-62  |
|  5 | Наталья  | пл. Мужества          | 527-52-42  |
|  9 | Ольга    | пр. Маршала Блюхера   | 276-86-35  |
|  6 | Петр     | ул. Правды            | 295-47-37  |
|  8 | Тимофей  | ул. Канта             | 296-38-47  |
|  2 | Федор    | пр. Славы             | 289-26-56  |
|  7 | Ярослав  | Удельный переулок     | 274-66-45  |
+----+----------+-----------------------+------------+
9 rows in set (0.00 sec)
```

Рис. 25. Выборка всех записей таблицы с сортировкой по полю *UserName* в алфавитном порядке

Осуществим проверку на корректность работы механизма задания прав пользователя, путем удаления таблицы *Phone_Numbers* с использованием следующей команды (рис. 26):

DROP TABLE Phone_Numbers;

```
mysql> use bd_telephone_directory;
Database changed
mysql> DROP TABLE Phone_Numbers;
ERROR 1142 (42000): DROP command denied to user 'New_User1'@'localhost' for table 'phone_numbers'
mysql>
```

Рис. 26. Попытка удаления таблицы пользователем *New_User1*

В данном случае, пользователю *New_User1* не разрешено удаление таблицы *Phone_Numbers*, в соответствии с настройками, которые были применены ранее. Это свидетельствует о том, что настроенный механизм безопасности на права доступа

пользователя *New_User1* к базе данных и таблице, функционирует должным образом.

Удаление таблиц, баз данных и пользователей из файловой системы *SQL*

Удаление таблиц, баз данных и пользователей из файловой системы *SQL* может осуществить только привилегированный *root*-пользователь (администратор) со своего АРМ. Удалим таблицу *Phone_Numbers* из созданной базы данных командой (рис. 27):

DROP TABLE Phone_numbers;

```
mysql> show tables;
Empty set (0.00 sec)
```

Рис. 27. Таблицы базы данных после удаления

Удалим базу данных *BD_Telephone_directory* командой *DROP DATABASE BD_Telephone_Directory;* (рис.28).

```
mysql> DROP DATABASE BD_Telephone_Directory;
Query OK, 0 rows affected (0.01 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+--------------------+
6 rows in set (0.00 sec)
```

Рис. 28. Базы данных после удаления *BD_Telephone_Directory*

Удалим все права доступа пользователя *New_User1* с помощью команды (рис.29):

*REVOKE ALL PRIVILEGES ON *.* FROM ‘New_User1’@’localhost;*

```
mysql> REVOKE ALL PRIVILEGES ON *.* FROM 'New_User1'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> show grants for 'New_User1'@'localhost';
+-------------------------------------------+
| Grants for New_User1@localhost            |
+-------------------------------------------+
| GRANT USAGE ON *.* TO `New_User1`@`localhost` |
+-------------------------------------------+
1 row in set (0.00 sec)
```

Рис. 29. Удаление прав пользователя.

Удалим пользователя New_User1 с помощью команды: *drop user 'New_User1'@'localhost';* (рис.30).

```
mysql> drop user 'New_User1'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> select User, Host from mysql.user;
+------------------+-----------+
| User             | Host      |
+------------------+-----------+
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
+------------------+-----------+
4 rows in set (0.00 sec)
```

Рис. 30. Пользователи после удаления *New_User1*

Удаление баз данных, пользователей и таблиц произведено успешно.

Заключение

Предложена методика установки и администрирование *SQL*-сервера на примере сервера *MySQL*, а также настройки параметров безопасности сервера базы данных с учетом принятой Политики ИБ организации и требований информационной безопасности ФСТЭК к автоматизированным системам класса защищенности 1Б.

Методика может быть использована для реализации требований политики информационной безопасности организации и регулятора, а также в учебном процессе ВУЗов при подготовке студентов соответствующего направления.

Список литературы

1. *Борзенкова С. Ю.* АНАЛИЗ МЕТОДОВ ОЦЕНКИ УРОВНЯ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ В ПРОЦЕССЕ ИХ ЭКСПЛУАТАЦИИ / *Борзенкова С. Ю., Казарина Е. Е.* // Известия ТулГУ. Технические науки. 2020. №5. – URL: https://cyberleninka.ru/article/n/analizmetodov-otsenki-urovnya-zaschischennosti-informatsionnyh-sistem-v-protsesseih-ekspluatatsii (дата обращения: 09.09.2023).

2. *Горлов А. П., Рытов М. Ю., Лысов Д. А.* Автоматизированная система оценки эффективности программно-аппаратных средств защиты информации // Автоматизация и моделирование в проектировании и управлении. 2019. №2 (4). – URL: https://cyberleninka.ru/article/n/avtomatizirovannaya-sistema-otsenkieffektivnosti-programmno-apparatnyh-sredstv-zaschity-informatsii (дата обращения: 10.09.2023).

3. *Чипчагов М. С., Вербицкий А. С., Титов В. А.* ЗАЩИЩЕННОСТЬ ИНФОРМАЦИИ В РАСПРЕДЕЛЁННЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ // Вестник УМЦ. 2018. №2 (19). – URL: https://cyberleninka.ru/article/n/zaschischennostinformatsii-v-raspredelyonnyh-informatsionnyh-sistemah (дата обращения: 10.09.2023).

4. *Алькаев В. А.* Средства анализа защищенности, применяемые для оценки эффективности функционирования средств защиты информации/ *Алькаев В. А., Фатеев А. Г.* // Инжиниринг и технологии. – 2018. – Vol. 3(2). – DOI 10.21685/2587-7704-2018-3-2-6.

5. *Аль-Аммори А., Дяченко П. В., Клочан А. Е., Бакун Е. В., Козелецкая И. К.* МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИИ // The Scientific Heritage. 2020. №51-1. – URL: https://cyberleninka.ru/article/n/metody-i-sredstva-zaschityinformatsii (дата обращения: 01.09.2023).

6. *Клочкова Т. В.* Роль аудита информационных технологий в информационной безопасности // Вопросы науки и образования. 2019. №10 (56). – URL: https://cyberleninka.ru/article/n/rol-audita-informatsionnyhtehnologiy-v-informatsionnoy-bezopasnosti (дата обращения: 20.09.2023).

7. *Запороцков П. А.* РАЗРАБОТКА МЕТОДА ПРОВЕДЕНИЯ АУДИТА СИСТЕМЫ ТЕХНИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ // NBI-technologies. 2020. №4. – URL: https://cyberleninka.ru/article/n/razrabotka-metoda-provedeniyaaudita-sistemy-tehnicheskoy-zaschity-informatsii (дата обращения: 20.09.2023).

8. *Егоров М. А.* Методика аудита нформационной безопасности в современных условиях // Вестник науки и образования. 2019. №11-2 (65). – URL: https://cyberleninka.ru/article/n/metodika-audita-informatsionnoybezopasnosti-v-sovremennyh-usloviyah (дата обращения: 3.09.2023).

9. *Рахимов Г. М.* УДАЛЕННАЯ РАБОТА АУДИТОРА В УСЛОВИЯХ ИСПОЛЬЗОВАНИЯ ЦИФРОВЫХ ТЕХНОЛОГИЙ: ПЕРСПЕКТИВЫ ТРАНСФОРМАЦИИ // ELS. 2022. №3. – URL: https://cyberleninka.ru/article/n/udalennaya-rabotaauditora-v-usloviyah-ispolzovaniya-tsifrovyh-tehnologiy-perspektivytransformatsii (дата обращения: 6.09.2023).

10. *Якимова В. А.* ВОЗМОЖНОСТИ И ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ ЦИФРОВЫХ ТЕХНОЛОГИЙ В АУДИТОРСКОЙ ДЕЯТЕЛЬНОСТИ // Вестник Санкт-Петербургского университета. Экономика. 2020. №2. – URL: https://cyberleninka.ru/article/n/vozmozhnosti-i-perspektivyispolzovaniya-tsifrovyh-tehnologiy-v-auditorskoy-deyatelnosti (дата обращения: 7.09.2023).

11. *Якубова И. И.* ВЛИЯНИЕ ИННОВАЦИОННЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИТИКИ БОЛЬШИХ ДАННЫХ НА КАЧЕСТВО АУДИТА // Индустриальная экономика. 2021. №5. – URL: https://cyberleninka.ru/article/n/vliyanie-innovatsionnyhinformatsionnyh-tehnologiy-i-analitiki-bolshih-dannyh-na-kachestvoaudita (дата обращения: 10.09.2023).

12. *Клишин Д. В., Чечулин А. А.* АНАЛИЗ СТАНДАРТОВ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ // Системы анализа и обработки данных. 2023. №1 (89). – URL: https://cyberleninka.ru/article/n/analiz-standartov-obespecheniyainformatsionnoy-bezopasnosti (дата обращения: 11.09.2023).

13. *Смирнов Г. Е., Макаренко С. И.* Использование тестовых информационно-технических воздействий для превентивного аудита защищенности информационно-телекоммуникационных сетей // Экономика и качество систем связи. 2020. №3 (17). – URL: https://cyberleninka.ru/article/n/ispolzovanie-testovyhinformatsionno-tehnicheskih-vozdeystviy-dlya-preventivnogo-auditazaschischennosti-informatsionno (дата обращения: 12.09.2023).

14. *Макаренко С. И., Смирнов Г. Е.* Анализ стандартов и методик тестирования на проникновение // Системы управления, связи и безопасности. 2020. №4. – URL: https://cyberleninka.ru/article/n/analiz-standartov-i-metodiktestirovaniya-na-proniknovenie (дата обращения: 12.09.2023).

15. *Двойнишников Н. Э., Исламутдинова Д. Ф.* Понятие и сущность аудита безопасности информационных систем // Московский экономический журнал. 2019. №10. – URL: https://cyberleninka.ru/article/n/ponyatie-i-suschnost-auditabezopasnosti-informatsionnyh-sistem (дата обращения: 13.09.2023).

16. *Мошак Н. Н., Касаткин В. В.* Сравнительный анализ методик аудита информационной безопасности информационнфх систем // VIII Межрегиональной научно-практической конференции «Перспективные направления развития отечественных информационных технологий (ПНРОИТ-2022)». Севастополь, 20-24 сентября 2022 г. / Севастопольский государственный университет; науч. ред. Б.В. Соколов. – Севастополь: СевГУ, 2022.– 252с. ISBN 978-5-6049992-2-6, с. 57-71.

17. *Мошак Н. Н.* Способ защиты автоматизированного рабочего места пользователя от несанкционированного доступа // Перспективные направления развития отечественных информационных технологий: материалы V межрегиональной научно-практической конф. Севастополь, 24-28 сентября 2019 г. / Севастопольский государственный университет; науч. ред. Б..В. Соколов. – Севастополь: СевГУ, 2019. – 464с. ISBN 978-5-6043402-0-2. С.289-291.

SECURITY POLICY ADMINISTRATION AND CONFIGURATION MySQL RELATIONAL DATABASE SERVERS

NIKOLAY N. MOSHAK
Dr.Sci.Tech., Associate Professor, Professor of the St. Petersburg State University of Telecommunications named after Prof. M. A. Bonch-Bruevich, Bolsheviks prospect, 22, St. Petersburg, 193323, Russia, nnmoshak49@mail.ru
Dr.Sci.Tech., Associate Professor, Professor of the St. Petersburg State University of Aerospace Instrumentation, Bolshaya Morskaya str, 67, lit. A, St. Petersburg, 190,000, Russia, nnmoshak49@mail.ru

SABINA R. RUDINSKAYA
Senior lecturer Educational Institution "Belarusian State Academy of Communications," Republic of Belarus, St. F. Skorina, 8/2, Minsk, 220114, Republic of Belarus, sabina.rudin@mail.ru

ALEXEY A. GRUZDEV
student of the St. Petersburg State University of Telecommunications named after Prof. M. A. Bonch-Bruevich, Bolshevikov prospect 22, St. Petersburg, 193232, Russia, gruzdev.a.a26@mail.ru

ABSTRACT

Introduction. A methodology for administering and configuring the security policy of the MySQL relational database server is proposed, taking into account the requirements of the information security policy of the organization and the regulator. **The purpose of the study**: to create and test a method of administration, which can be used to implement the requirements of the information security policy of the organization and the regulator, as well as in the educational process of higher educational institutions in the preparation of students of the corresponding direction. **The method of conducting the study**: an approach was used that involves performing each stage of the study step by step, with a detailed description of each step and presentation of the results in the form of visual materials. This allows us to clearly and clearly present the intermediate and final results of the study. **The results obtained:** a method of installing and administering a SQL server using the example of a MySQL server, as well as setting the security parameters of a database server using the example of the regulator's information security requirements for automated systems of the 1B security class, is proposed. All steps of interaction with MySQL are considered: installation, configuration, creation of a user and a database, as well as execution of CRUD (Create, Read, Update, Delete) queries to it. The settings of the database server security parameters were made taking into account the specified information security requirements of the regulator. It is shown that the settings for the access of an unprivileged user to the server using the "MySQL Workbench" utility significantly increase the security of the system, since it involves checking its access rights via the command line and password authentication in accordance with the regulator's information security requirements for automated systems of the 1B security class, which eliminates the threat of unauthorized access to the database.

Keywords: MySQL database; MySQL security policy; MySQL administration information security; create and configure a server.

REFERENCES

1. Borzenkova S. Yu. *ANALIZ METODOV OCENKI UROVNYa ZAShhISHhENNOSTI INFORMACIONNY`X SISTEM V PROCESSE IX E`KSPLUATACII* [ANALYSIS OF METHODS OF ASSESSING THE LEVEL OF SECURITY OF INFORMATION SYSTEMS DURING THEIR OPERATION]. Borzenkova S. Yu., Kazarina E. E. *Izvestiya Tul`skogo gosudarstvennogo universiteta. Texnicheskie nauki* [Izvestiya Tula State University. Technical sciences]. 2020. no. 5. Available at: https://cyberleninka.ru/article/n/analizmetodov-otsenki-urovnya-zaschischennosti-informatsionnyh-sistem-v-protsesseih-ekspluatatsii (accessed: 09.09.23) (in Russian).
2. Gorlov A. P., Rytov M. Yu., Lysov D. A. *Avtomatizirovannaya sistema ocenki e`ffektivnosti programmno-apparatny`x sredstv zashhity` informacii* [AUTOMATED SYSTEM FOR ESTIMATION-EFFICIENCYSOFTWARE MEANS OF IN-FORMATION PROTECTION]. *Avtomatizaciya i modelirovanie v proektirovanii i upravlenii* [Automation and modeling in design and management]. 2019. no. 2(4). Available at: https://cyberleninka.ru/article/n/avtomatizirovannaya-sistema-otsenkieffektivnosti-programmno-apparatnyh-sredstv-zaschity-informatsii (accessed: 10.09.23) (in Russian).
3. Chipchagov M. S., Verbitskiy A. S., Titov V. A. *ZAShhISHhENNOST` INFORMACII V RASPREDELYoNNY`X INFORMACIONNY`X SISTEMAX* [INFORMATION SECURITY IN DISTRIBUTED INFORMATION SYSTEMS]. *Vestnik Universiteta mirovy`x civilizacij* [Bulletin of the University of World Civilizations]. 2018. no. 2(19). Available at: https://cyberleninka.ru/article/n/zaschischennostinformatsii-v-raspredelyonnyh-informatsionnyh-sistemah (accessed: 10.09.23) (in Russian).
4. Al'kaev V. A. *Sredstva analiza zashhishhennosti, primenyaemy`e dlya ocenki e`ffektivnosti funkcionirovaniya sredstv zashhity` informacii* [Security analysis tools for evaluation of performance efficiency of information protection means]. Al'kaev V. A., Fateev A. G. *Inzhiniring i texnologii* [Engineering and technology]. 2018. Vol. 3(2). DOI: 10.21685/2587-7704-2018-3-2-6. (in Russian).
5. Al-Ammouri A., Dyachenko P. V., Klochan A. E., Bakun E. V.., Kozeletska I. K. *METODY` I SREDSTVA ZAShhITY` INFORMACII* [METHODS AND MEANS OF PROTECTING INFORMATION]. The Scientific Heritage. 2020. no. 51-1. Available at: https://cyberleninka.ru/article/n/metody-i-sredstva-zaschityinformatsii (accessed: 01.09.23) (in Russian).
6. Klochkova T. V. *Rol` audita informacionny`x texnologij v informacionnoj bezopasnosti* [The role of information technology audit in information security]. *Voprosy` nauki i obrazovaniya* [Issues of science and education]. 2019. no. 10(56). Available at: https://cyberleninka.ru/article/n/rol-audita-informatsionnyhtehnologiy-v-informatsionnoy-bezopasnosti (accessed: 20.09.23) (in Russian).
7. Zaporotskov P. A. *RAZRABOTKA METODA PROVEDENIYa AUDITA SISTEMY` TEXNIChESKOJ ZAShhITY` INFORMACII* [DEVELOPMENT OF A METHOD FOR CONDUCTING AN AUDIT OF THE INFORMATION SECURITY SYSTEM]. NBI-technologies]. 2020. no. 4. Available at: https://cyberleninka.ru/article/n/razrabotka-metoda-provedeniyaaudita-sistemy-tehnicheskoy-zaschity-informatsii (accessed: 20.09.23) (in Russian).
8. Egorov M. A. *METODIKA AUDITA INFORMACIONNOJ BEZOPASNOSTI V SOVREMENNY`X USLOVIYaX* [METHODS OF AUDIT OF INFORMATION SECURITY IN MODERN CONDITIONS]. *Vestnik nauki i obrazovaniya* [Bulletin of Science and Education]. 2019. no. 11-2(65). Available at: https://cyberleninka.ru/article/n/metodika-audita-informatsionnoybezopasnosti-v-sovremennyh-usloviyah (accessed: 3.09.23) (in Russian).
9. Rakhimov G. M. *UDALENNAYa RABOTA AUDITORA V USLOVIYaX ISPOL`ZOVANIYa CIFROVY`X TEXNOLOGIJ: PERSPEKTIVY` TRANSFORMACII* [REMOTE WORK OF AN AUDITOR IN THE CONTEXT OF USING DIGITAL TECHNOLOGIES: PROSPECTS FOR TRANSFORMATION]. ELS. 2022. no. 3. Available at: https://cyberleninka.ru/article/n/udalennaya-rabotaauditora-v-usloviyah-ispolzovaniya-tsifrovyh-tehnologiy-perspektivytransformatsii (accessed: 6.09.23) (in Russian).
10. Yakimova V. A. *VOZMOZhNOSTI I PERSPEKTIVY` ISPOL`ZOVANIYa CIFROVY`X TEXNOLOGIJ V AUDITORSKOJ DEYaTEL`NOSTI* [OPPORTUNITIES AND PROSPECTS FOR USING DIGITAL TECHNOLOGIES IN AUDITING]. *Vestnik Sankt-*

Peterburgskogo universiteta. E`konomika [Bulletin of St. Petersburg University. Economy]. 2020. no. 2. Available at: https://cyberleninka.ru/article/n/vozmozhnosti-i-perspektivyispolzovaniya-tsifrovyh-tehnologiy-v-auditorskoy-deyatelnosti (accessed: 7.09.23) (in Russian).

11. Yakubova I. I. *VLIYaNIE INNOVACIONNY`X INFORMACIONNY`X TEXNOLOGIJ I ANALITIKI BOL`ShIX DANNY`X NA KAChESTVO AUDITA'* [IMPACT OF INNOVATIVE INFORMATION TECHNOLOGIES AND BIG DATA ANALYTICS ON AUDIT QUALITY]. *Industrial`naya e`konomika* [Industrial economy]. 2021. no. 5. Available at: https://cyberleninka.ru/article/n/vliyanie-innovatsionnyhinformatsionnyh-tehnologiy-i-analitiki-bolshih-dannyh-na-kachestvoaudita (accessed: 10.09.23) (in Russian).
12. Klishin D. V., Chechulin A. A. *ANALIZ STANDARTOV OBESPEChENIYa INFORMACIONNOJ BEZOPASNOSTI* [ANALYSIS OF INFORMATION SECURITY STANDARDS]. *Sistemy` analiza i obrabotki danny`x* [Data analysis and processing systems]. 2023. no. 1(89). Available at: https://cyberleninka.ru/article/n/analiz-standartov-obespecheniyainformatsionnoy-bezopasnosti (accessed: 11.09.23) (in Russian).
13. Smirnov G.E., Makarenko S. I. *Ispol`zovanie testovy`x informacionno-texnicheskix vozdejstvij dlya preventivnogo audita zashhishhennosti informacionno-telekommunikacionny`x setej* [USE OF TEST INFORMATION AND TECHNICAL IMPACTS FOR PREVENTIVE SECURITY AUDIT OF INFORMATION AND TELECOMMUNICATION NETWORKS]. *E`konomika i kachestvo sistem svyazi* [Economics and quality of communication systems]. 2020. no. 3(17). Available at: https://cyberleninka.ru/article/n/ispolzovanie-testovyhinformatsionno-tehnicheskih-vozdeystviy-dlya-preventivnogo-auditazaschischennosti-informatsionno (accessed: 12.09.23) (in Russian).
14. Makarenko S. I., Smirnov G. E. *Analiz standartov i metodik testirovaniya na proniknovenie* [Analysis of penetration testing standards and methodologies]. *Sistemy` upravleniya, svyazi i bezopasnosti* [Management, communication and security systems]. 2020. no. 4. Available at: https://cyberleninka.ru/article/n/analiz-standartov-i-metodiktestirovaniya-na-proniknovenie (accessed: 12.09.23) (in Russian).
15. Dvoinishnikov N. E., Islamutdinova D. F. *Ponyatie i sushhnost` audita bezopasnosti informacionny`x sistem* [CONCEPT AND ESSENCE OF INFORMATION SYSTEMS SECURITY AUDIT]. *Moskovskij e`konomicheskij zhurnal* [Moscow Economic Journal]. 2019. no. 10. Available at: https://cyberleninka.ru/article/n/ponyatie-i-suschnost-auditabezopasnosti-informatsionnyh-sistem (accessed: 13.09.23) (in Russian).
16. Moshak N. N., Kasatkin V. V. *Sravnitel`ny`j analiz metodik audita informacionnoj bezopasnosti informacionnfx sistem* [Methodological audit of the medium-term analysis of information security Information system]. *VIII Mezhregional`noj nauchno-prakticheskoj konferencii «Perspektivny`e napravleniya razvitiya otechestvenny`x informacionny`x texnologij (PNROIT-2022)»* [VIII Interregional scientific and practical conference “Promising scientific and practical technologies for the development of information technologies (PNROIT-2022)”]. *Sevastopol`: SevGU* [Sevastopol: SevSU]. 2022. 252 p. ISBN 978-5-6049992-2-6, pp. 57-71 (in Russian).
17. Moshak N. N. *Sposob zashhity` avtomatizirovannogo rabochego mesta pol`zovatelya ot nesankcionirovannogo dostupa* [WAY OF PROTECTION OF THE AUTOMATED WORKPLACE OF THE USER AGAINST UNAUTHORIZED ACCESS] *Perspektivny`e napravleniya razvitiya otechestvenny`x informacionny`x texnologij: materialy` V mezhregional`noj nauchno-prakticheskoj konferencii* [Promising directions for the development of domestic information technologies: materials in the interregional scientific and practical conference]. *Sevastopol`: SevGU* [Sevastopol: SevSU]. 2019. 464p. ISBN 978-5-6043402-0-2. pp. 289-291 (in Russian).