**Кыргызский государственный технический университет**

**Им. И. Раззакова**

**Институт Информационных Технологий**

**Кафедра Информатики и Вычислительной Техники**

*ОТЧЁТ*

***По лабораторным работам №1-2***

***По дисциплине: Безопасность систем баз данных***

**Выполнила: Тагайбекова Астра**

**Группа: ИБ инж 1-22**

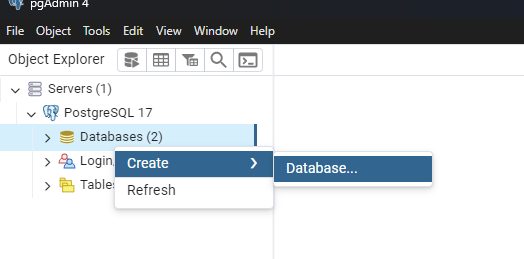
**Проверил: Куручбеков Б. З.**

***Бишкек 2024***

***Лабораторная работа №1***

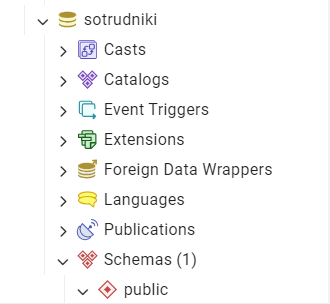
***Задание:*** Создать базу данных в PostgreSQL. Создать в ней таблицу, в которой будут данные работников, такие как: ФИО, заработная плата, ИНН. Зашифровать данные любого столбца в таблице через встроенные функции.

Выполнение задания:

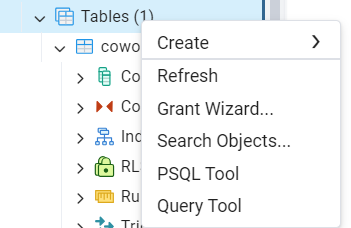


В программе pgAdmin создаю новую базу данных.

Servers > PostgreSQL 17 > Databases ( нажимаю правой кнопкой ) > Create > Database



В открывшимся окне даю название базе данных (в моем случае это “Sotrudniki”) и сохраняю.



В списке баз данных появилась моя созданная база данных. Для создания в ней таблицы я буду использовать запросы SQL.

Task1 > Schemas > Public > Tables (правой кнопкой) > Query Tool

Перед созданием таблицы, я установлю расширение pgcrypto для дальнейшего шифрования данных

CREATE EXTENSION IF NOT EXISTS pgcrypto;

CREATE TABLE coworkers (

id SERIAL PRIMARY KEY,

coworker\_name TEXT,

surname TEXT,

salary NUMERIC,

 inn CHAR

);

Командой CREATE TABLE, я создаю таблицу coworkers с полями id ( порядковый номер работника ), coworker\_name ( ФИО работника ), salary ( Заработная плата ), inn ( персональный 16-значный номер работника ), surname (Фамилия).

INSERT INTO coworkers (coworker\_name, surname, salary, inn) VALUES

('Aliya', 'Ivanova', '30000', '1234567899876543'),

('Argen', 'Petrov', '36000', '9876543211234567'),

('Emir', 'Kukushkin', '20000', '6754378907315265'),

('Aelita', 'Panikova', '10000', '9008443728515900'),

('Atai', 'Ovechkin', '15000', '7623981245679812'),

('Natasha', 'Loshadkina', '30000', '2167983412873490'),

('Igor', 'Vezuchiy', '45000', '9012569854908790'),

('Olga', 'Barboskina', '17000', '1155774499552288'),

('Dastan', 'Lopatkin', '20000', '9021784367125634'),

('Astra', 'Notmalnaya', '40000', '8954127854238965');

Командой INSERT INTO, я заполняю таблицу coworkers, а именно поля Coworker\_name, salary и inn,surname данными.

ALTER TABLE coworkers

ALTER COLUMN salary TYPE bytea USING pgp\_sym\_encrypt(salary::text, 'G5@lT!p9Xq3#bW8');

В данном коде, с помощью расширения pgcrypto, шифруется поле salary

G5@lT!p9Xq3#bW8

Это пароль для расшифровки.



После запроса просмотра таблицы, видно, что поле было успешно зашифровано.

Однако просто расшифровать их не получится. Для просмотра данных в расшифрованном виде, нужно использовать команду вывода с определенными параметрами вывода:

SELECT coworker\_name, surname,

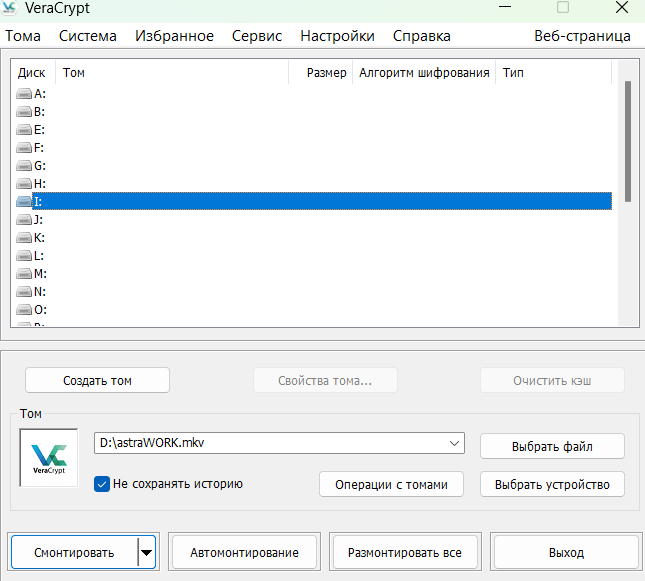
pgp\_sym\_decrypt(salary, 'G5@lT!p9Xq3#bW8')::text AS salary,

inn

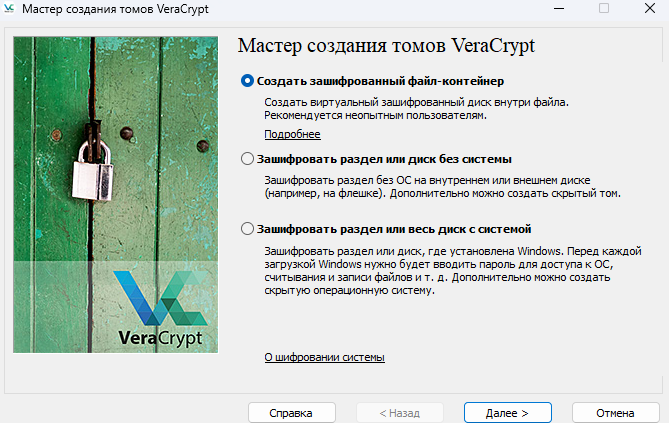
FROM coworkers;

***Лабораторная работа №2***

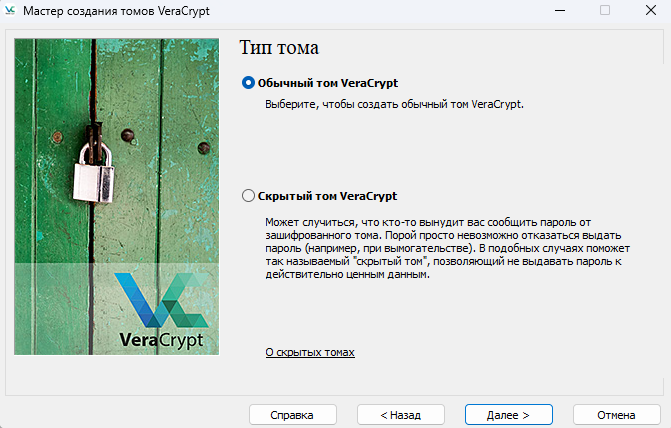
***Задание:*** Установить программу VeraCrypt, создать с помощью программы новый локальный диск, поместить в него файлы созданной базы данных и установить пароль на открытие данного диска.



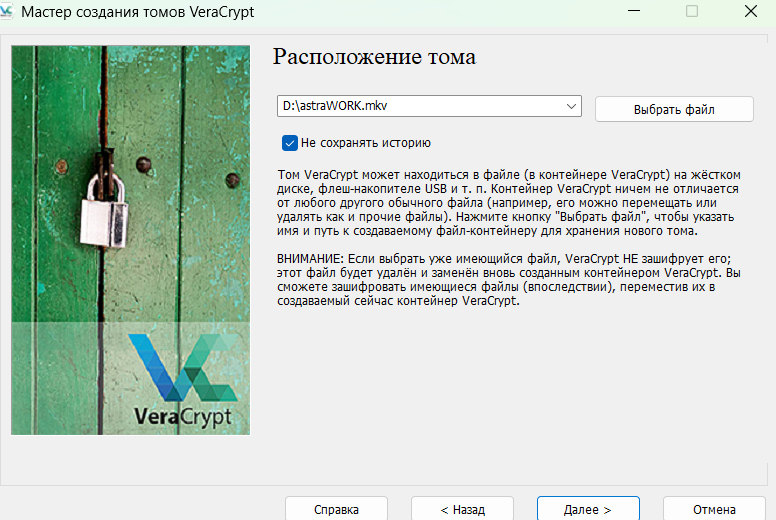
В программе выбираем любое имя для диска и создаем том.



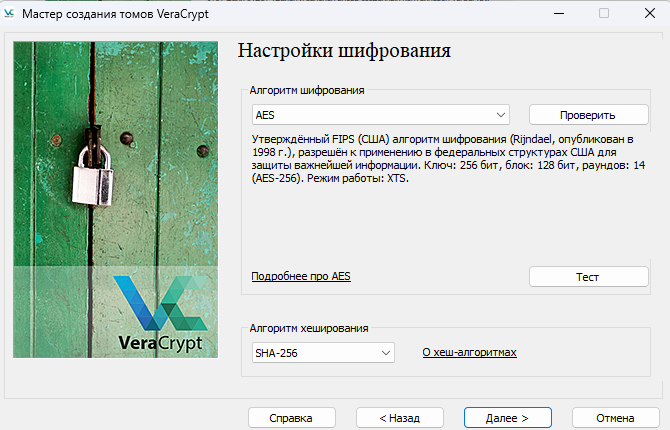
Создаем зашифрованный файл-контейнер



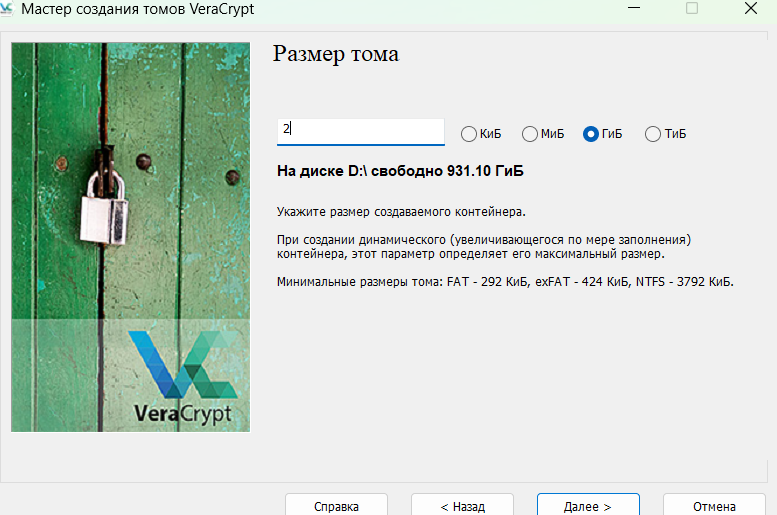
Выбираем обычный том.



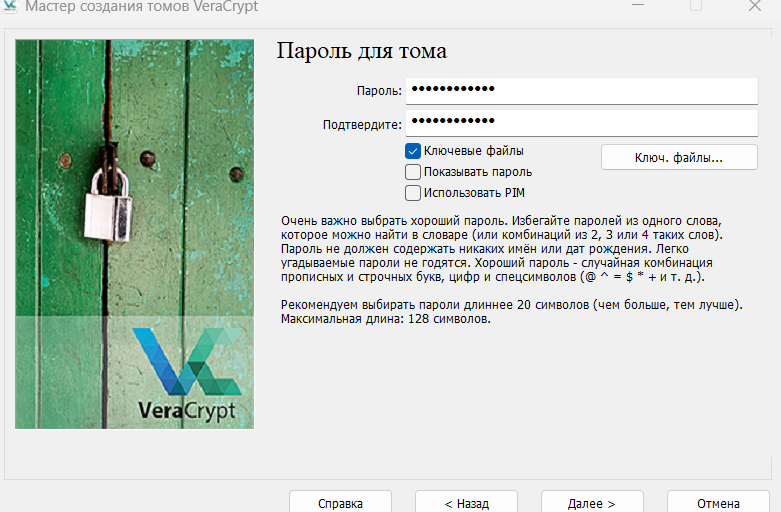
Выбираем расположение тома.



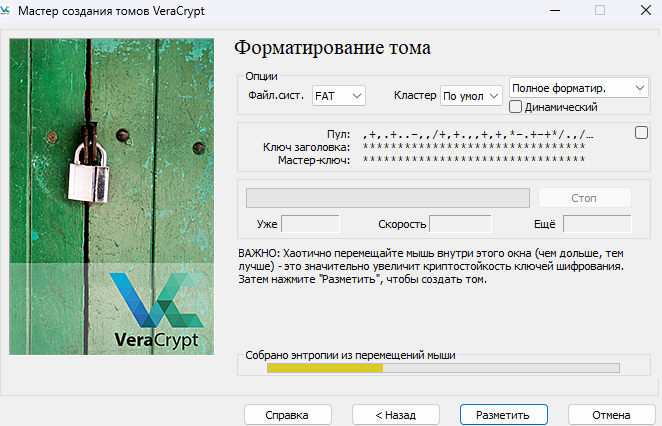
При необходимости меняем настройки шифрования.



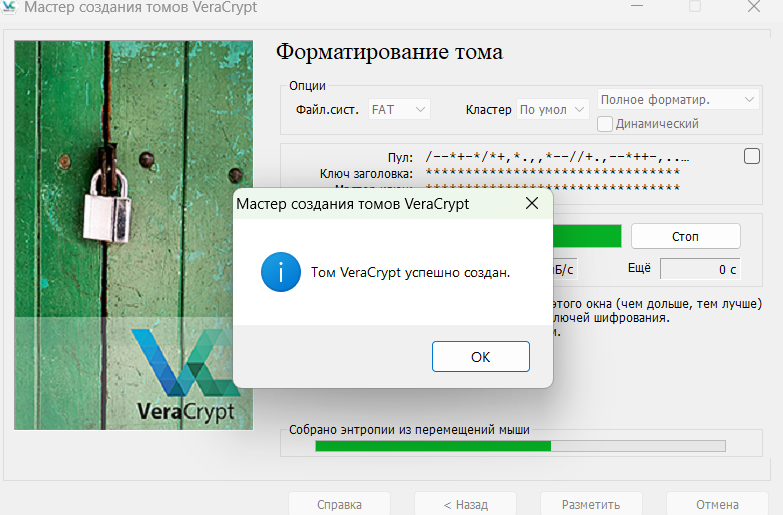
Выделяем ему желаемый размер.



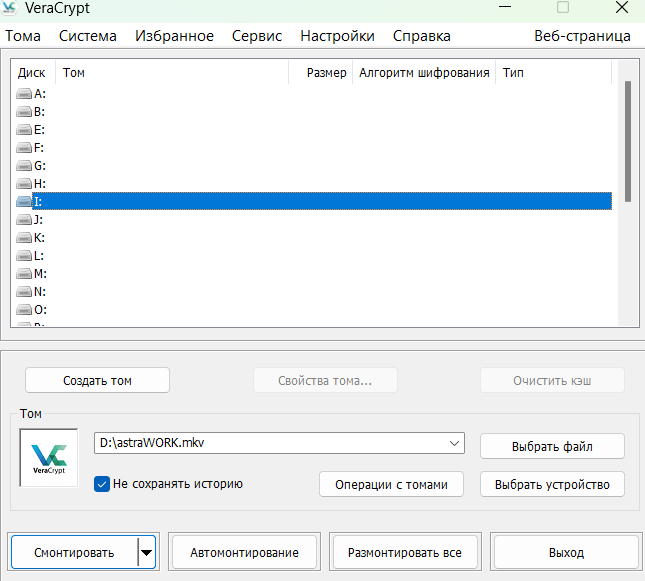
Задаем пароль, который будем вводить при открытии самого диска.

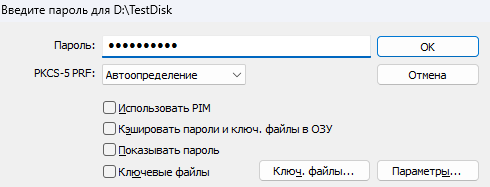


Движением курсора, задаем скорость создания шифрованного диска.

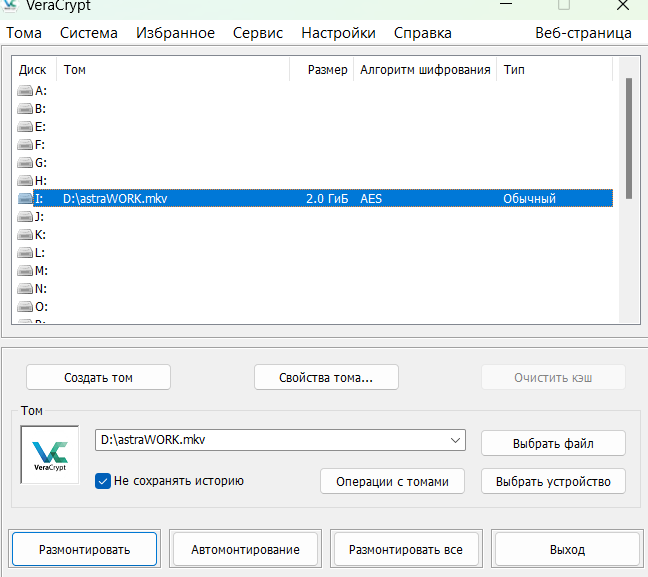


Том успешно создан.

  
В программе VeraCrypt снова выбираем имя для диска и указываем путь к созданному тому. После нажимаем кнопку “Смонтировать”.



Вводим наш пароль для открытия диска.



Новый локальный диск был успешно создан.

Переносим в него папку с ранее созданной базой данных.

Теперь в новом диске я могу хранить важные мне данные, который будут зашифрованы и доступ к ним будет только у меня.