Лабораторная работа №13
Применение алгоритмов хэширования данных

1 Цель работы

1.1 Познакомиться с методами применения алгоритмов хэширования данных.

2 Литература

2.1 Зверева В. П., Сопровождение и обслуживание программного обеспечения компьютерных систем : учебник для студ. учреждений сред. проф. Образования / В. П. Зверева, А. В. Назаров. – М. : Издательский центр «Академия», 2018. – 256 с.

3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

5.1 Разработать оконное приложение для вычисления хэш-сумм файлов и последующей отправки файлов и их хэш-сумм по сети. Получатель должен проверять полученные файлы на целостность.

5.2 Модифицировать приложение для применения криптографической соли при хэшировании файлов. (см п.9.1)

6 Порядок выполнения работы

6.1 Повторить теоретический материал п. 3.1;

6.2 Выполнить задания 5.1-5.2

6.3 Ответить на контрольные вопросы п. 8;

6.4 Заполнить отчет п. 7.

7 Содержание отчета

7.1 Титульный лист;

7.2 Цель работы;

7.3 Ответы на контрольные вопросы п. 6.3;

7.4 Вывод по проделанной работе.

8 Контрольные вопросы

8.1 Перечислите наиболее популярные алгоритмы хэширования

8.2 Зачем применяется криптографическая соль?

9 Приложение

9.1 Криптографическая **соль (salt)** — это случайные данные, которые **добавляют к паролю перед хешированием**, чтобы защититься от нескольких

типов атак. Она не делает пароль «крепче», но делает его **уникальным** даже при одинаковых паролях.

Допустим, два пользователя выбрали один и тот же пароль: qwerty.

Если система хранит хеши (например, SHA-256), получатся одинаковые значения:

```
SHA256("qwerty") = a1b2c3...
```

Значит злоумышленник сразу понимает:

- какие пользователи выбрали одинаковые пароли,
- может использовать радужные таблицы (precomputed hash tables).

Что даёт соль

Перед хешированием пароль модифицируется:

```
hash = SHA256(salt + password)
```

Теперь даже при одинаковых паролях хеши **разные**, потому что используются **разные соли**.

Если хеш зависит от случайной соли, радужные таблицы бесполезны, нужно создавать таблицу для каждой соли, а их миллионы.

Соль хранится вместе с хешем в базе данных. Она **не секретная**. Нормально, что злоумышленник знает соль — безопасность обеспечивается не ею, а тем, что пароль хешируется с использованием индивидуальной соли.