**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет безопасности информационных технологий**

**Дисциплина:**

**«Разработка систем аутентификации и криптографии»**

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

**«Методы аутентификации»**

**Выполнил:**

Магистрант гр. N42514с

*И. М. Гарипов*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

**Проверил:**

Ассистент ФБИТ

*Р. И. Фёдоров*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

Санкт-Петербург

2020г.

**СОДЕРЖАНИЕ**

[1 Цель работы 3](#_Toc57845690)

[2 Описание выбранных средств реализации и обоснования выбора 4](#_Toc57845691)

[3 Описание алгоритма 5](#_Toc57845692)

[4 Демонстрация работы программы и код программы 6](#_Toc57845693)

[5 Выводы 7](#_Toc57845694)

1. Цель работы

**Задача**: реализация механизма аутентификации в клиент-серверном веб-приложении

*Требования к реализации*:

* необходимо реализовать метод аутентификации в клиент-серверном приложении согласно варианту;
* клиент должен представлять собой веб-страницу с формой авторизации пользователя;
* сервер должен включать в себя две части:

1. таблица идентификаторов (данные о пользователях для аутентификации: логин/пароль/токен/и т. д. в зависимости от метода аутентификации);
2. процесс с реализованной логикой метода аутентификации.
3. Описание выбранных средств реализации и обоснования выбора

В качестве средства реализации метода аутентификации был выбран язык программирования Python, поскольку данный язык является кроссплатформенным. Это означает, что реализация кода будет доступна для работы независимо от операционной системы.

К тому же Python имеет довольно лёгкий синтаксис, прост в изучении и не требует больших ресурсов для его использования.

Также выбор данного инструмента разработки обусловлен личной симпатией автора.

1. Описание алгоритма

Реализовать аутентификацию по паролю с подтверждением по email. В таблице идентификаторов должны храниться: логин, email, пароль, временный код подтверждения. Таблица идентификаторов должна представлять собой таблицу в реляционной БД, данные должны передаваться через SQL-запросы. При аутентификации на сервере сравниваются пароли и на email пользователя отправляется сгенерированный на сервере временный код подтверждения. На клиенте после отправки данных с паролем должен произойти редирект на форму для ввода временного кода подтверждения. После отправки кода на сервере сравниваются пришедший код и код из БД. При совпадении кодов аутентификация считается успешной и происходит редирект на страницу-заглушку.

1. Демонстрация работы программы и код программы

Код программы и демонстрация работы представлены в репозитории на GitHub в описании к данной лабораторной работе и доступны по ссылке <https://github.com/BetterThanLeonid/Garipov_Labs_RSAC/tree/main/LR2_RSAC_20_Garipov_WebAuth>

1. Выводы

В ходе лабораторной работы разработана система парольной аутентификации с подтверждением по электронной почте в клиент-серверном приложении. Написана реализация изученного метода на языке Python с использованием пакета Flask