Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Информационные сети. Основы безопасности

ОТЧЁТ

к лабораторной работе №7

на тему

**ЗАЩИТА ПО ОТ НЕСАНКЦИОНИРОВАННОГО ИСПОЛЬЗОВАНИЯ**

Выполнил: студент гр.253505 Сенько Н.С.

Проверил: ассистент кафедры информатики Герчик А.В.

Минск 2025

**СОДЕРЖАНИЕ**

[1 Цель работы 3](#_Toc190108682)

[2 Ход работы 4](#_Toc190108683)

[Заключение 6](#_Toc190108684)

# 1 ЦЕЛЬ РАБОТЫ

Целью исследования стало изучение и практическое применение методов обфускации JavaScript-кода на примере реализации анимации «матричного дождя» (эффект 404). Основные задачи включали: преобразование читаемого кода в запутанную версию с сохранением функциональности, замену осмысленных идентификаторов на бессмысленные комбинации символов, внедрение шестнадцатеричных значений, разбиение строковых констант, а также усложнение логики для противодействия реверс-инжинирингу. Работа направлена на демонстрацию того, как обфускация затрудняет анализ алгоритмов, защищает интеллектуальную собственность и повышает устойчивость кода к модификациям.

# 2 ХОД РАБОТЫ

Исходный код, реализующий анимацию падения символов на элементе canvas, был проанализирован для выявления ключевых компонентов: настройки холста, генерации символов, управления кадрами через setInterval. На первом этапе проведена минификация: удалены пробелы, комментарии, сокращены имена переменных (например, context → ctx). Далее выполнена обфускация: идентификаторы заменены на комбинации вида \_0x5a7b54, строковые литералы разбиты через split('') и закодированы в шестнадцатеричные значения (например, getElementById → ['getElementById']), а функции переписаны с использованием анонимных выражений и замыканий. Для усложнения логики добавлены фиктивные операции (например, вычисления с parseInt и псевдослучайными числами), бессмысленные проверки условий и избыточные вызовы методов. Ключевые части кода, такие как генерация символов и обновление позиций, были замаскированы через динамическое обращение к свойствам объекта (например, ctx['fillText'] вместо ctx.fillText). После преобразований проведено тестирование: проверена корректность отрисовки анимации, частота кадров, обработка изменений размеров окна. Результаты подтвердили, что функциональность сохранена, но анализ кода стал значительно сложнее из-за отсутствия понятных имен переменных и нагромождения ложной логики.

# ЗАКЛЮЧЕНИЕ

Практическое применение методов обфускации к JavaScript-коду продемонстрировало их эффективность для защиты алгоритмов от анализа. Замена идентификаторов, маскировка строк, внедрение избыточных операций и динамическое выполнение методов позволили сохранить работоспособность анимации «матричного дождя», но сделали код практически нечитаемым без дополнительной декомпиляции. Работа подтвердила, что обфускация является важным инструментом в веб-разработке для защиты клиентской логики, усложнения кражи или модификации кода, особенно в проектах с требованием к безопасности. Однако подчеркивается необходимость баланса между уровнем запутывания и производительностью: избыточные вычисления могут замедлить выполнение скрипта, что требует оптимизации на этапе тестирования.