МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра информационных технологий**

**ОТЧЕТ ОБ АУДИТЕ**

Выполнил студент группы 22/2                                       Д.А.Ткачева

Направление подготовки  01.03.02  Прикладная математика и информатика Курс    2

Отчет приняла кандидат физико-математических наук, доцент                                                                                       Е.П. Лукащик

Краснодар

2022 г.

СОДЕРЖАНИЕ

[1 Цель работы 3](#_Toc98113292)

[2 Защита от XSS 3](#_Toc98113293)

[3 Защита от SQL-INJECTION 3](#_Toc98113294)

[4 Защита от CSRF 4](#_Toc98113295)

[5 Защита от Upload и Include уязвимости 4](#_Toc98113296)

[6 Вывод 4](#_Toc98113297)

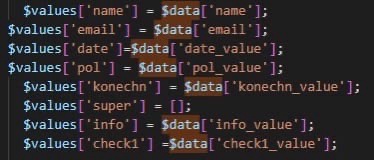
# 1 Цель работы

Провести аудит безопасности вашего приложения и исправить уязвимости. В нем должны быть разделы, посвященные уязвимостям XSS, SQL Injection, CSRF, Include, Upload. В отчете указать по каждой уязвимости примененные методы защиты с примерами вашего кода.

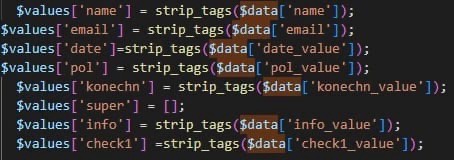
# 2 Защита от XSS

XSS - это когда злоумышленник пытается через формы на сайте добавить свой javascript-код, который затем выполнится в браузере админа/менеджера сайта или других пользователей и натворит дел.

Изначально, при загрузке в форму данные из базы не приводились к безопасному формату и была возможность провести XSS-атаку:



Используем strip\_tags() (возвращает строку str, из которой удалены HTML и PHP тэги) для строковых данных:



# 3 Защита от SQL-INJECTION

SQL-INJECTION - атаки, позволяющие злоумышленнику производить различные несанкционированные действия над базой данных. Они могут затрагивать как сами данные, так и структуру базы.

Необходимо экранировать символы поступающие из формы при этом, записав базу данных (DBO::quote) или использовать подготовленные запросы:

# https://sun9-30.userapi.com/s/v1/ig2/B4rqO4ApD7GcdAMOXiLK0UjCclcaRuLSpZX7D-pKRPlfgT1ddC7Od6h90i-UlqBZIrn9soiRbu-YHk6rbhfQR9y4.jpg?size=428x109&quality=95&type=album

# 4 Защита от CSRF

CSRF —вид атак на посетителей веб-сайтов, использующий недостатки протокола HTTP. Если жертва заходит на сайт, созданный злоумышленником, от её лица тайно отправляется запрос на другой сервер.

Защитим важные поля форм, (изменение записи в базе данных и удаление записи) добавив токен, привязанный к сессии пользователя и проверку токена:

# https://sun9-55.userapi.com/s/v1/ig2/G-aiGVtMZKFCFxHq_FxSTuptUidwNxSekCL5qQ7pqsST15aUc7ZYLpqs6-O1z6kqqbL7RwnELxVunGJpCPrSx2ez.jpg?size=510x96&quality=95&type=album

# 5 Защита от Upload и Include уязвимости

PHP-include—уязвимость, которая позволяет «принудить» произвольный файл и выполнить PHP код в любом файле на сервере.

Upload уязвимостей нет, т. к. пользователь не загружает на сервер файлы. Include уязвимостей нет, т. к. пользователь не дает данные для подключения модулей обработчиков.

# 6 Вывод

Защитили сайт от различного рода уязвимостей в коде, атак типа внедрения SQL и атак на посетителей веб-сайта. Теперь при загрузке в форму данные из базы приводятся к безопасному формату.