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 25/2 _____ А.А. Козин

Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем
Курс 2

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

Краснодар
2022 г.

СОДЕРЖАНИЕ

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 — тип атаки на веб-системы, заключающийся во внедрении в выдаваемую веб-системой страницу вредоносного кода (который будет выполнен на компьютере пользователя при открытии им этой страницы) и взаимодействии этого кода с веб-сервером злоумышленника. Является разновидностью атаки «Внедрение кода».

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

```
$values['name'] = $data['name'];

$values['email'] = $data['email'];
```

Используем strip_tags() (возвращает строку str, из которой удалены HTML и PHP тэги) для строковых данных и intval() (Возвращает целое значение переменной var) для целочисленных:

```
$values['name'] = strip_tags($data['name']);

$values['email'] = strip_tags($data['email']);
```

3 Защита от SQL-INJECTION

SQL-INJECTION - один из распространённых способов взлома сайтов и программ, работающих с базами данных, основанный на внедрении в запрос произвольного SQL-кода.

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

```
$stmt = $db->prepare("INSERT INTO users (login, hash)

VALUES(:login,:hash)");

$stmt->bindParam(':login', $login);

$stmt->bindParam(':hash', $hash_pass);

$stmt->execute();
```

4 Защита от CSRF

CSRF — вид атак на посетителей веб-сайтов, использующий недостатки протокола HTTP.

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

```
<?php
header(‘Content-Type: text/html; charset=UTF-8’);
session_start();
$_SESSION[‘token’] = uniqid();
$token = $_SESSION[‘token’];
<input type = “hidden” name=”token_del”
<?php print “value=’$token’”;?>>
}
```

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.

ПРИЛОЖЕНИЕ

Ссылка на репозиторий GIT

https://github.com/KozinAlexandr/back_7