Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение   
высшего образования   
«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

ОЦЕНКА РАБОТЫ

**ОТЧЕТ**

**по лабораторной работе №3**

Разработка безопасных веб-приложений

Дата Подпись Ф.И.О.

Преподаватель С. Г. Мирвода

Студент Н. А. Примаков

Группа РИ – 571227

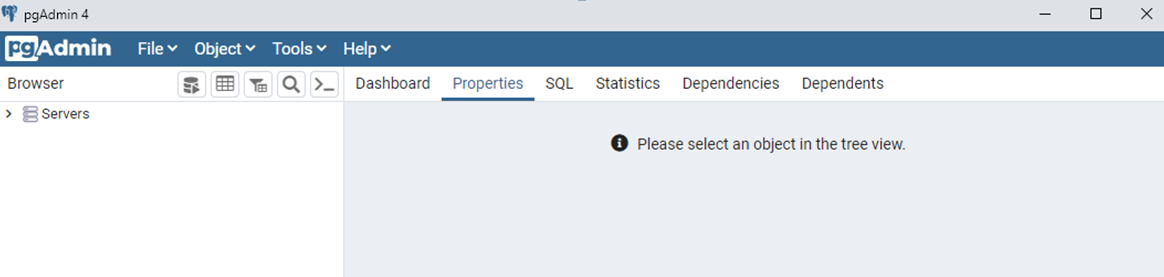
Екатеринбург 2021

**Цель работы.**

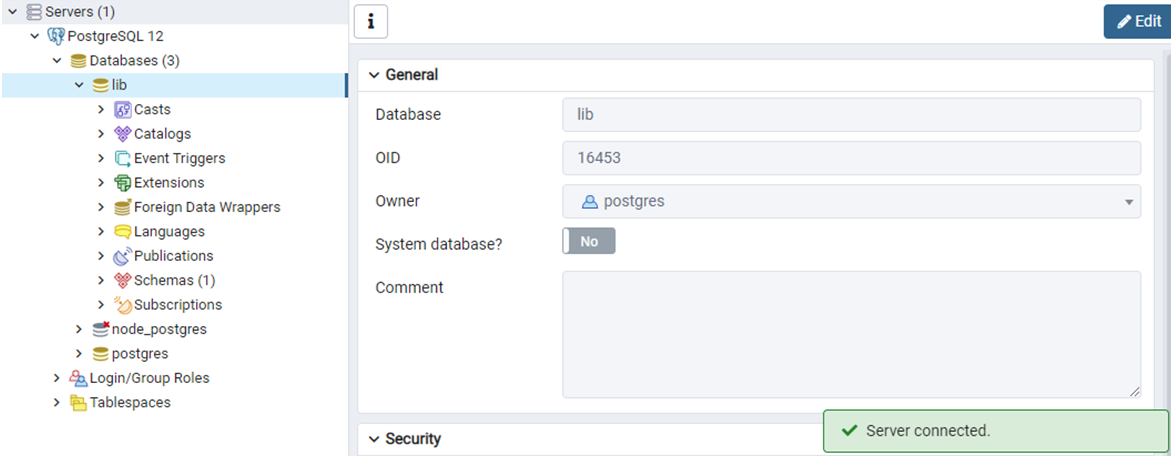
Поиск и устранение XSS уязвимостей.

**Выполнение работы.**

**1. Установим PostgreSQL сервер.**

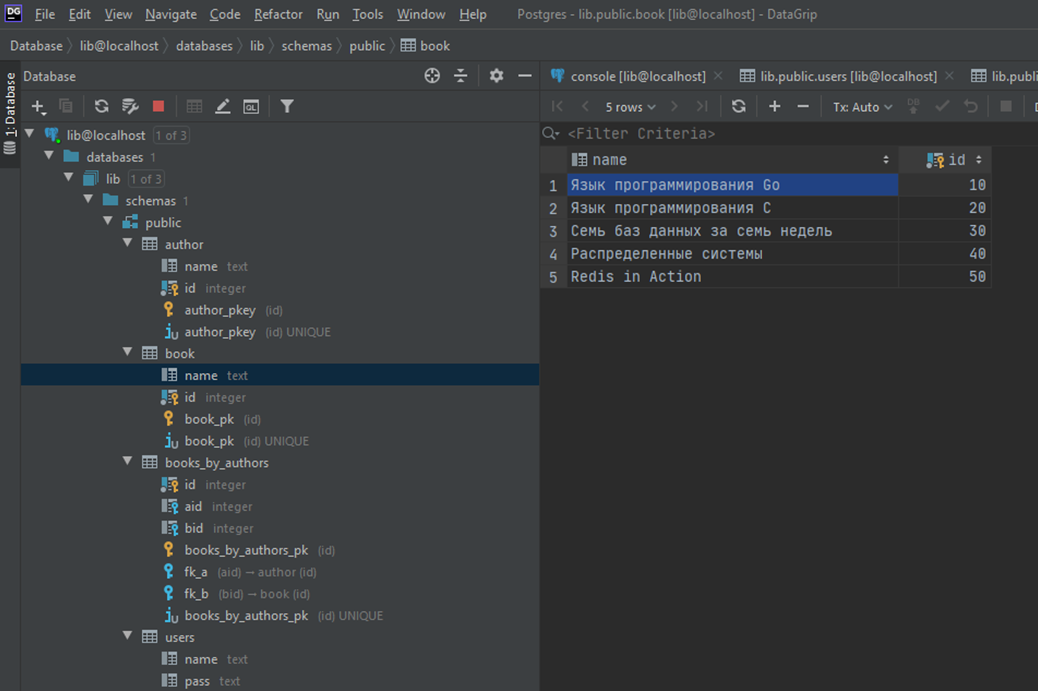


**2. Создадим БД lib.**

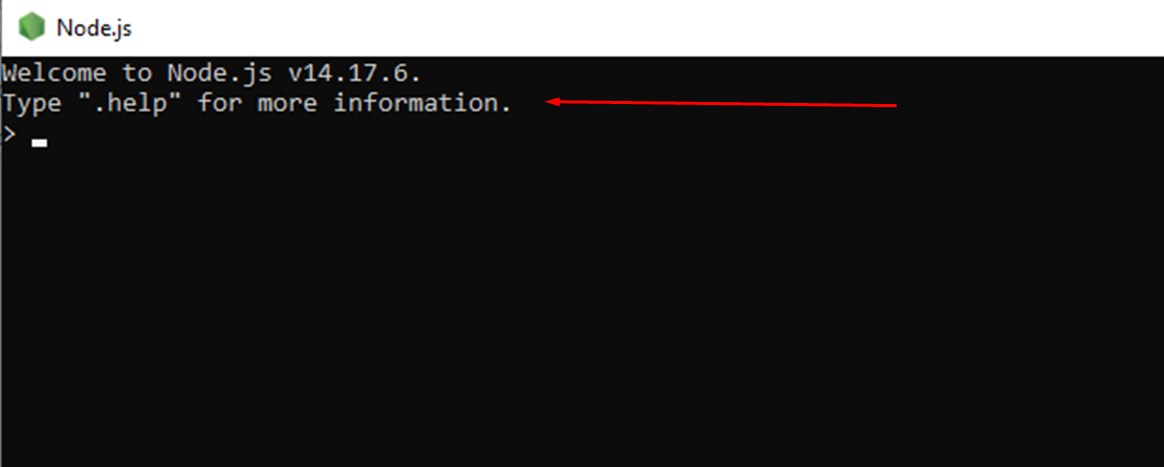


**3. Применим к ней скрипты из папки db (либо создать объекты и вставить данные в таблицы руками). Скрипты выполним в порядке, указанном в имени файла.**

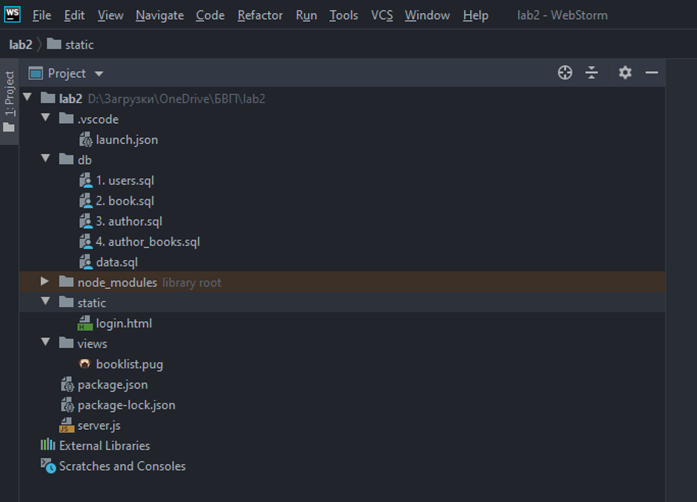
**3.1 Восстановим данные из файла data.sql.**

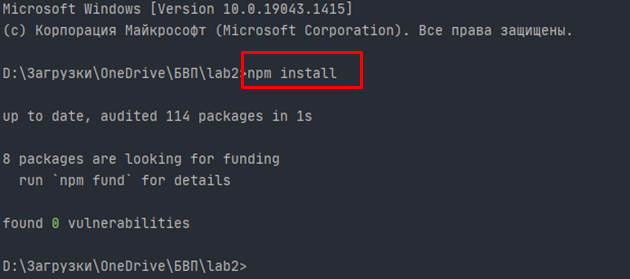


**4. Установим nodejs версии 14.**

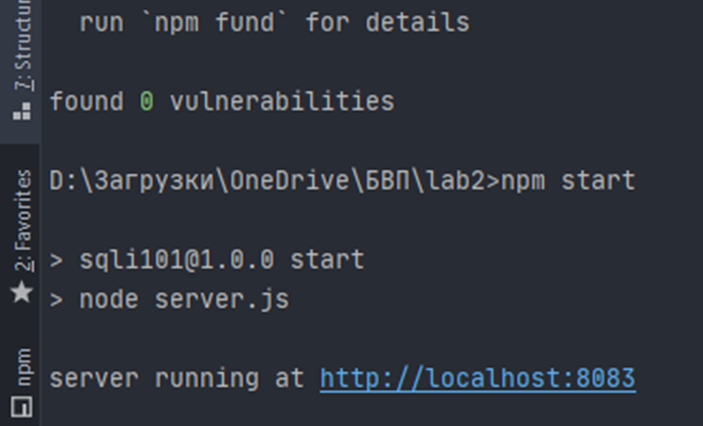


**5. Перейдем в папку lab2 и выполним в ней команду npm install.**

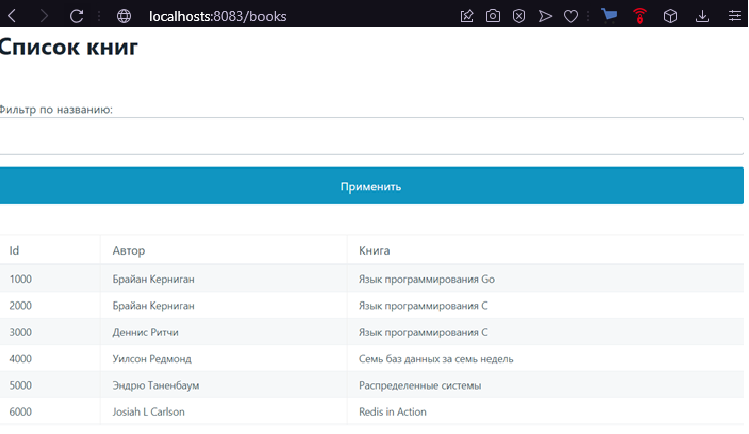




**6. Запустим сайт через Visual Studio Code или через команду npm start.**



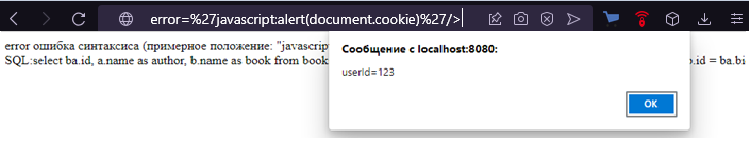
**7. Войти на сайт и увидеть список книг и авторов.**



**8. На странице со списком книг найти**

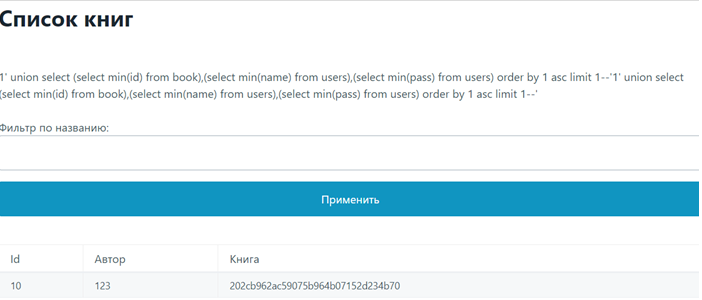
**8.1 Reflected XSS в поиске книг.**

Reflected XSS-уязвимость возникает, когда пользовательский ввод с URL-адреса или данных POST отражается на странице без сохранения, что позволяет злоумышленнику внедрить вредоносный контент.



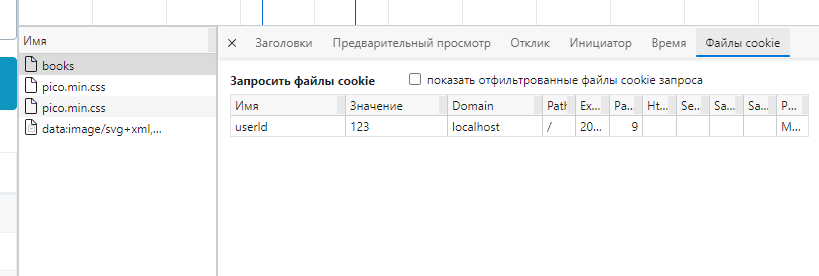
**8.2 Persisted (Stored) XSS при создании книги и отображении списка книг.**

Stored XSS-уязвимость может возникнуть, если имя пользователя онлайн-доски объявлений не очищено должным образом при выводе на страницу. В этом случае злоумышленник может вставить вредоносный код при регистрации нового пользователя в форме.



**8.3 Потенциальную уязвимость через Cookie Injection**

При входе на сайт генерируется сессионный Cookie файл, который позволяет авторизировать конкретного пользователя в сессии, что и может быть использовано во вред

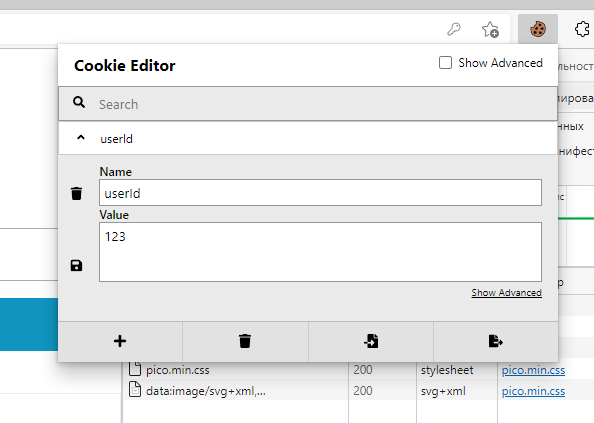


Для перехвата Cookie можно использовать xss, а именно команду:

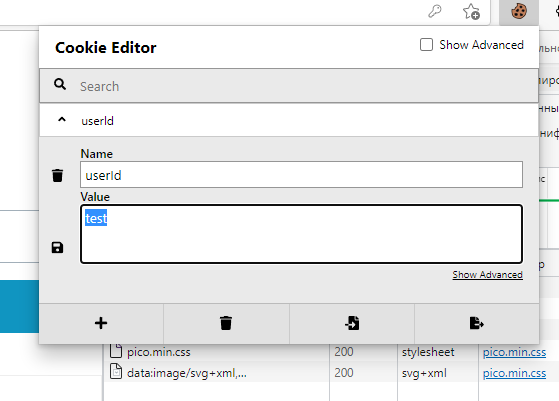
[http://localhost:8080/books?name=1<img%20src=1%20onerror=%27javascript:alert(document.cookie)%27/](http://localhost:8080/books?name=1%3cimg%20src=1%20onerror=%27javascript:alert(document.cookie)%27/)>

**8.4 Некорректное создание сессионной cookie, которое приводит к захвату сессии (Session hijacking)**

Для подмены Cookie можно использовать расширение Cookie Editor для браузера.



В нём отображается созданный Cookie, который можно заменить на конкретный желаемый.



Для этого зайдем под другим пользователем и заполним поле Value на уже обозначенное выше «123», тогда сайт будет воспринимать текущую сессию, для другого пользователя, что может быть использовано для зловредных действий, например смена пароля или удаление данных.

**9. Написать отчёт с описанием найденных уязвимости и примерами их эксплуатации**

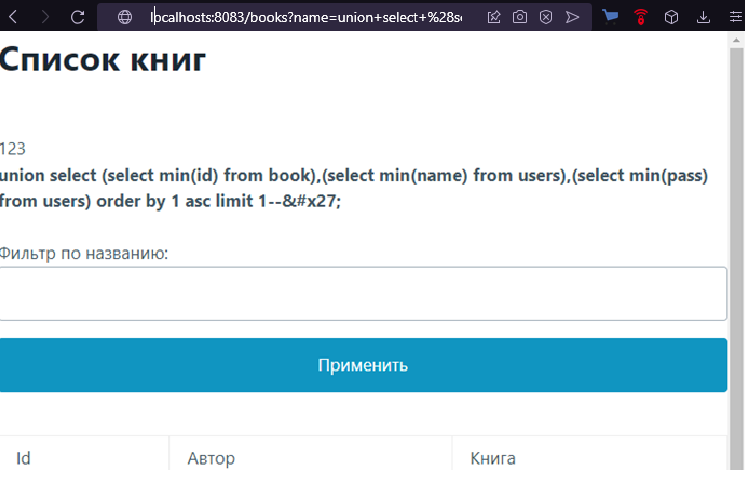
**10. Исправить уязвимость**

**10.1 В отчёте привести пример того, что уязвимости больше не эксплуатируются**

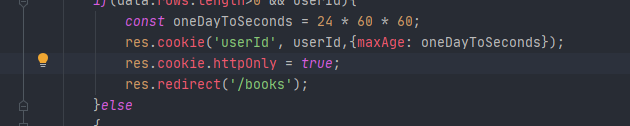
Для устранения уязвимостей XSS, требуется произвести Санитайзинг вводимых значений, для этого можно использовать простую функцию для отслеживания вводимых символов.



Таким образом вложенные sql запросы перестают действовать.



Для устранения уязвимости связанных с Cookie файлами следует устанавливать флаг HttpOnly, таким образом блокируя доступ на запись и чтение.



В результате санитайзинга доступ к Cookie также был невозможен.

