***Вогністий Олег Валентинович***

*Національний університет «Одеська юридична академія»,*

*студент 2-го курсу факультету кібербезпеки та інформаційних технологій*

# БЕЗПЕКА ПОПУЛЯРНИХ JAVASCRIPT ФРЕЙМВОРКІВ

Фреймворки JavaScript є своєрідним набором різноманітних інструментів для створення веб-застосунків, веб-сайтів як для десктопних, так і для мобільних пристроїв.

Станом на 2021 рік у світі серед веб-розробників найбільш популярними є такі фреймворки [1]:

1. React.js – 40.14%;
2. jQuery – 34.43%;
3. Express – 23.82%;
4. Angular – 22.96%;
5. Vue.js – 18.97%.

Кожен з них має унікальний функціонал, структуру, програмні модулі і компоненти. Також у фреймворків є свої стандарти для забезпечення безпеки майбутніх проектів. Але не зважаючи на те, що ці стандарти відрізняються один від одного, види порушень безпеки є доволі одноманітними.

Найпоширенішим видом загрози є міжсайтовий скриптинг або XSS-атака. Такого роду атаки дозволяють хакерам внести на веб сторінку JavaScript-код, який буде виконуватися щоразу, коли користувачі заходитимуть на цю сторінку [2]. Отримавши файли cookie, хакер матиме змогу отримати особисті дані користувача, наприклад: контактні дані, дані кредитної картки, дані авторизації (паролі, електронна пошта тощо).

Іншим видом загрози є SQL-ін’єкції. Вдалі SQL-ін’єкції надають доступ до бази даних, в яких хакери можуть змінювати або видаляти дані без відома користувача [3]. Завдяки цьому можна, наприклад, підробити якісь документи, змінивши в них дані, або зробити їх непридатними для використання.

Останнім з поширених видів порушень безпеки є підробка міжсайтових запитів (CSRF – Cross-site request forgery). Ця загроза дає змогу хакеру спонукати користувача до виконання певних дій, які не є обов'язковими, але на які може «клюнути» користувач-жертва. Наприклад, це може бути повідомлення про те, що треба змінити електронну пошту або оновити пароль, і в разі, якщо користувач перейде на сторінку із відповідним запропонованим запитом, хакер отримає контроль над його обліковим записом.

Як свідчить статистика [1], нині найбільш популярним фреймворком є React.js. Проте цей фреймворк не має стандартних налаштувань безпеки. Хоча деякі бібліотеки та компоненти, можуть допомогти ліквідувати загрозу, але цього замало. Тому веб-розробникам треба власноруч створювати механізми для захисту даних у своїх проектах, адже React.js уразливий до всіх видів загроз розглянутих вище. За часів існування даного фреймворка створено численні інструкції та рекомендації щодо захисту веб-сайтів чи веб-застосунків. Так, для захисту від XSS-атак є рекомендовано[4]:

1. використовувати брандмауер веб-застосунків у коді програми;
2. використовувати спеціальні інструменти для сканування вбудованих програм на наявність XSS-вразливостей;
3. реалізувати бібліотеки сніпетів, як-от: ES7 React, Redux тощо;
4. уникати написання коду, в який хакери потенційно можуть вставити інструкції для запуску шкідливих сценаріїв, наприклад, виключити HTML-теги: <script>, <object>, <embed> і <link>;

Такі заходи протидії загрозам себе виправдовують, але це не означає, що програми, написані у React.js, будуть захищені на 100%. Адже для цього треба доволі часто протягом всього життєвого циклу проекту тестувати програмний код на визначення рівня захищеності.

На відміну від React.js, фреймворк Angular вважається більш захищеним, завдяки наявності системних методів для забезпечення безпеки проектів. Angular підтримує серверну автентифікацію та авторизацію, що значно мінімізує виникнення можливих атак [5]. Оскільки стандартні API-інтерфейси DOM браузера не мають змоги захистити від атак, при розробці доцільно

використовувати шаблони Angular та уникати прямої взаємодії з DOM[6]. Завдяки потужності і захищеності цей фреймворк часто використовують для створення веб-сайтів або веб-застосунків для великих бізнес-компаній або для таких, які тісно взаємодіють із користувачем. У документації Angular є інструкції того, як уникати тих чи інших загроз для запобігання хакерських атак.

Фреймворк Vue.js переважно використовується для створення невеликих проектів, де потрібна швидкість. Через те, що Vue.js має порівняно слабку захищеність і від атак типу XSS, SQL-injection навряд чи захистить. Тому для забезпечення максимального захисту проекту, веб-розробникам у Vue.js, так само як і розробникам, що використовують React.js, треба самостійно встановлювати необхідні параметри для контролю вхідних і вихідних даних. Розробники у Vue можуть дезінфікувати HTML-код перед реалізацією або використовувати зовнішні бібліотеки для захисту від атак. Якщо HTML безпечний, можна відображати вміст HTML і захищати програму як до, так і після рендерингу [7].

Отже, безпека веб-сайтів та веб-застосунків при використанні фреймворків лише на 50% залежить від самого фреймворка. Якщо при розробці не приділити достатньо уваги саме забезпеченню безпеки проекта, то він автоматично себе сам захистити не зможе, і всі зусилля по його розробленню зійдуть нанівець. Для тих, хто має намір стати веб-розробником, треба ретельно вивчати і залучати інструменти захисту проекту від можливих загроз. А користувачам потрібно не поспішати клікати на всі підряд кнопки задля того, щоб не запустити вірус до своїх даних.

### *Список використаних джерел*

1. Most used web frameworks among developers worldwide, as of 2021. URL: https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/

2. Что такое XSS-уязвимость и как тестировщику не пропустить ее. URL: https://habr.com/ru/post/511318/

3. Веб-безопасность. URL: https://developer.mozilla.org/ru/docs/Learn/Server-side/First_steps/Website_security.

4. Руководство по безопасности React.js: угрозы, уязвимости и способы их устранения. URL: https://bestprogrammer.ru/izuchenie/rukovodstvo-po-bezopasnosti-react-js-ugrozy-uyazvimosti-i-sposoby-ih-ustraneniya.

5. Angular. FAQ. URL: http://angular-doc.herokuapp.com/misc/faq.

6. Лучший JavaScript-фреймворк 2021: React или Vue? URL: https://evrone.ru/react-vs-vue.

7. Наилучшие практики угловой безопасности. URL: https://www.cisin.com/coffee-break/ru/technology/angular-security-best-practices.html

**Ключові слова:** безпека вебсайтів, SQL-ін’єкції, міжсайтовий скриптинг.

**Ключевые слова**: безопасность веб-сайтов, SQL-инъекции, межсайтовый скриптинг.

**Keywords**: website security, SQL injection, cross-site scripting.

**Науковий керівник:** *к.т.н., доц. Трофименко О. Г.*

***Дрига Артем Вадимович***

*Національний університет «Одеська юридична академія», студент 2-го курсу факультету кібербезпеки та інформаційних технологій*

## РОЗГОРТАННЯ ОПЕРАЦІЙНИХ СИСТЕМ ЗА ДОПОМОГОЮ CLONEZILLA

Комп’ютерні класи та аудиторії потребують постійного технічного обслуговування, зокрема періодичної перестановки операційних систем. Це може зайняти багато часу та ресурсів через те, що процес встановлення операційної системи "з нуля" це тривала процедура, що вимагає присутності