УДК 621.382

АНАЛІЗ ЗАГРОЗ ТА МЕТОДІВ ЗАХИСТУ МЕРЕЖ ВІД XSS АТАК ТА SQL ІН'ЄКЦІЙ

Валуйський С.В., Собко Т.А.
Інститут телекомунікаційних систем КПІ ім. Ігоря Сікорського, Україна
E-mail: valuiskyi.stanislav@lll.kpi.ua

ANALYSIS OF THREATS AND METHODS OF PROTECTING NETWORKS FROM XSS ATTACKS AND SQL INJECTION

The article describes common threats that come from XSS attacks and SQL injection and possible damage they may cause. It also suggests effective methods of preventing these attacks and protection against them.

XSS-атаки та SQL-ін’єкції є одними із найпоширеніших видів атак на веб-додатки на веб-сайти. Вразливості до цих атак радикально знижують рівень захищеності інформації в системі (додатку). В даній статті проводиться огляд можливих загроз та наслідків цих атак, а також наводяться ефективні методи убезпечення від них.

Атаки типу XSS використовуються для наступних злочинних дій:

- крадіжка Cookies – дозволяє перехопити конфіденційні значення cookies користувача та індикатори сесії, які потім можуть бути використані для підробки сесії і несанкціонованого входу в систему;
- крадіжка даних з форм – перехоплення значень, які користувач вводить у поля форм на веб-сайті. Серед них може бути така критична інформація як логіни та паролі, дані кредитних карток, ідентифікаційні номери, інші дані які потім можуть бути використані у злочинних цілях;
- підробка контенту – внесення зловмисником змін у зовнішній вигляд та наповнення сторінок веб-сайту; може використовуватись як різновид фішингу, для реалізації попередньо описаної крадіжки даних (щоб спровокувати користувача ввести дані у підробне вікно), для несанкціонованого показу власної реклами на чужому веб-сайті та ін.

Також, з використанням XSS можуть реалізуватись інші види атак – DDoS-атака, підробка міжсайтових запитів (CSRF / XSRF)

XSS-атаки розділяють на два види – пасивні та активні.

Пасивні атаки відбуваються коли жертва переходить за спеціальним чином згенерованим посиланням, яке містить в собі зловмисний код (часто зашифрований). Для реалізації атак цього виду зазвичай використовується соціальна інженерія (наприклад, маскування повідомлення із посиланням під важливий лист, розсилка в соцмережах, розміщення посилання на форумах).

Убезпечення від таких атак лежить у площині інформаційної грамотності кожного конкретного користувача. Це такі правила, як: не переходити за підозрілими посиланнями, не відкривати повідомлення із невідомих та підозрілих джерел, не натискати на кнопки і рекламні посилання на ненадійних веб-сайтах (наприклад, сайтах з піратським контентом).

Активні XSS-атаки є більш небезпечними, оскільки їх тяжко, або й неможливо, помітити. При такій атаці зловмисний код впроваджується в базу даних або який-небудь файл на сервері. Таким чином, всі відвідувачі сайту автоматично стають жертвами. Інтеграція може бути здійснена зокрема за допомогою SQL-ін'єкції (описано далі). Загалом, вразливість до атак цього виду присутня скрізь, де веб-програма використовує вхідні дані користувача для генерування вихідних даних, без достатнього рівня фільтрації, перевірки та кодування.

Захист від таких вразливостей має бути реалізований при розробці, з використанням наступних правил [1]:

- не дозволяти вставку ненадійних даних (даних користувача) ніде, окрім дозволених місць, та не приймати код JavaScript із ненадійних джерел;
- кодувати всі ненадійні дані перед вставкою їх до дозволених місць. До цього відноситься:
    - кодування HTML перед вставкою ненадійних даних у вміст HTML-елемента,
    - кодування атрибутів перед вставкою ненадійних даних у значення загальних атрибутів HTML,
    - кодування JavaScript перед вставкою ненадійних даних у значення даних JavaScript,
    - кодування значення JSON у контексті HTML та зчитування даних за допомогою JSON.parse,
    - кодування CSS перед вставкою ненадійних даних у значення властивостей стилю HTML,
    - кодування URL перед вставкою ненадійних даних у значення параметрів URL в HTML,
- якщо дані, які вводяться користувачем, можуть містити розмітку HTML – перевіряти цю розмітку за допомогою призначеної для цього бібліотеки;
- впровадити Content Security Policy.

SQL-ін’єкції стали поширеною проблемою веб-сайтів, керованих базами даних. Вразливість легко виявити і легко використати, і будь-який веб-сайт чи додаток із навіть мінімальною базою користувачів, ймовірно, буде підданий спробі такого нападу [2]. Успішна SQL-ін’єкція може дозволити:

- отримати несанкціонований доступ до конфіденційних даних (логіни, паролі, дані кредитних карток, особисті дані користувачів);
- змінити дані в базі – підробити, закодувати, додати до них власні дані (наприклад, код для XSS-атаки);
- обмежити доступ до даних;
- знищити дані;
- створити бекдор у системі;
- змінити логіку програми.

Серйозність SQL-ін’єкцій обмежується майстерністю та уявою зловмисника та (меншою мірою) рівнем розробленого захисту. Загалом, ці атки мають високий рівень небезпеки.

Висновки. Потенційно вразливим до атак цього виду є будь-який веб-сайт чи додаток, що використовує бази даних на мові SQL. Запобігти успішній реалізації SQL-ін’єкцій можна використовуючи наступні методи:

- використовувати підготовлені вирази (з параметризованими даними) при генеруванні запитів до бази даних. Спеціально призначені для цього методи існують на Java, .NET, PHP, Hibernate;
- використовувати збережені процедури (цей варіант у деяких випадках може замінити попередній, але є менш пріоритетним);
- Перевіряти дані, введені користувачем, по білому списку (Whitelist Input Validation) перед формуванням запиту на основі цих даних;
- Використовувати принципи мінімізації привілеїв (кожен користувач баз даних повинен мати лише мінімальні необхідні йому права, не дозволяти всім записам мати повні права адміністратора), та створювати різних користувачів для різних додатків та баз даних (уникати використання одного і того ж облікового запису власника/адміністратора) [3].

Література

1. Cross Site Scripting Prevention Cheat Sheet [Електронний ресурс]. Режим доступу: https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html.
2. SQL Injection [Електронний ресурс]. Режим доступу: https://owasp.org/www-community/attacks/SQL_Injection.
3. SQL Injection Prevention Cheat Sheet [Електронний ресурс]. Режим доступу: https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html.