УЯЗВИМОСТИ WEB-ПРИЛОЖЕНИЙ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Федорцов П. С.

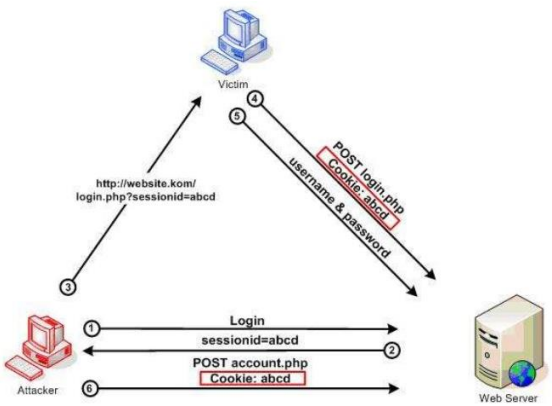
Гладкая В. С. – магистр техн.наук,
ассистент каф. ИПиЭ

Цель работы: исследовать уязвимости web-приложений. При разработке web-приложений часто все усилия направленны в основном на добавление нового функционала. Вопросам безопасности и качества кода при этом часто уделяется недостаточно внимания. В результате в приложениях появляются различные ошибки и уязвимости. Уязвимости обычно возникают в результате ошибок в проектировании приложения, недостаточной проверки данных, вводимых пользователем или ошибок при написании кода.

Термин уязвимость используется в web-безопасности для обозначения недостатков в коде сайта или программном обеспечении сервера, используя которые, можно нарушить целостность системы и вызвать неправильную работу. [2]

Некоммерческая организация OWASP (Open Web Application Security Project) после исследования предоставила список 10 наиболее опасных и распространенных уязвимостей в web-приложениях. [1]

1. Инъекции. К этой категории относятся различные уязвимости, такие как SQL, NoSQL, OS, LDAP инъекции. Они происходят, когда непроверенные данные, поступающие от пользователя, попадают в интерпретатор как часть запроса или команды. Это может привести к выполнению отправленного злоумышленником кода или к получению доступа к данным без авторизации.
2. Ошибки в системе аутентификации. Части приложений, связанные с аутентификацией, часто реализуются некорректно, что приводит к компрометации паролей или ключей.
3. Раскрытие важной информации. Многие приложения недостаточно хорошо защищают важную информацию, например, финансовую, информацию о состоянии здоровья, персональные данные и т.д. Атакующий может украсть эту информацию и использовать её для мошенничества, шантажа или применения социальной инженерии.
4. Внешние сущности XML. Много старых обработчиков XML обращаются к ссылкам на внешние сущности. Это может быть использовано для получения доступа ко внутренним ресурсам и удалённого выполнения кода.
5. Ошибки в управлении правами доступа. Ограничения доступа для аутентифицированных пользователей часто могут быть настроены неправильно. Это может привести к получению несанкционированного доступа к данным и функционалу, требующему особых привилегий.
6. Небезопасные настройки – это наиболее распространённая проблема. Часто эта уязвимость возникает из-за небезопасных настроек по умолчанию или неполных настроек. Например, слишком подробные сообщения об ошибках могут раскрыть важную информацию.
7. Межсайтовый скриптинг (XSS). Происходит при добавлении вводимых пользователем данных на web страницу без их проверки. XXS позволяет выполнять скрипты в браузере пользователя, что может привести к перехвату сессии, искажать данные или перенаправить пользователя на вредоносный или фишинговый сайт.
8. Небезопасная десериализация объектов. Небезопасная десериализация часто приводит к удалённому выполнению кода или возможности проведения других атак, например, повышения привилегий.
9. Использование компонентов с известными уязвимостями. Часто в разных компонентах, используемых в приложении находят разные уязвимости. Компоненты такие как фреймворки или библиотеки обычно работают с теми же привилегиями, что и основное приложение. Поэтому они позволяют проводить различные атаки на приложение.
10. Недостаточно подробное ведение журнала и мониторинг. Недостаточно подробное ведение журнала вместе позволяет атакующему в случае успеха дольше оставаться незамеченным, проводить дальнейшее исследование системы, незаметно воровать и подменять данные.



Список использованных источников:
1. https://www.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf
2. https://www.eduherald.ru/ru/article/view?id=12471