УДК 004.42

Ковалев Александр Сергеевич,
специалист по защите информации, инженер-программист,
ООО «НетКрэкер»,
г. Саратов

АНАЛИЗ WEB-СЕРВИСОВ НА НАЛИЧИЕ XSS-УЯЗВИМОСТЕЙ

Аннотация. В материале рассматривается актуальность XSS-уязвимостей и их виды. На основании полученных данных были разработаны алгоритмы для поиска различных видов XSS-уязвимостей, а также разработано ПО для автоматизации их поиска на web-ресурсах.

Ключевые слова: XSS-уязвимости, компьютерная безопасность, поиск уязвимостей.

Обеспечение информационной безопасности вычислительных систем является одной из главных задач для каждой организации, в хозяйственной деятельности которой применяются алгоритмы сбора, обработки, хранения, передачи информации. Из-за распространения web-приложений стали возможны множество угроз информационной безопасности. 10 лет назад преобладали статические web-приложения, они не имели интерактивных интерфейсов взаимодействия с пользователями. Следовательно, почти не было уязвимостей, которые могли бы быть использованы нарушителями. Это позволяло разработчикам игнорировать вопросы, связанные с безопасностью. На данный момент, практически все web-сайты и приложения являются динамическими, в них огромное количество новых технологий, используемых web-браузерам. Новейшие технологии позволяют подключать к web-приложениям все возможные модули, которые позволяют посетителю по максимуму использовать web-ресурсы (например, доски объявлений, формы обратной связи и т.д.). Но, к сожалению, пользователь может столкнуться с проблемами. Технологии, функционирующие в динамических web-сайтах, обеспечивают хорошую платформу нарушителям для проведения XSS-атак. В 53% приложений актуальна угроза атак на клиентов с помощью межсайтового скриптинга [2]. С помощью внедренного кода нарушитель может получить несанкционированный доступ к конфиденциальной

информации пользователя и совершать противоправные действия, как на локальных компьютерах пользователей, так и в сетевом оборудовании компании, меняя конфигурацию сети и программного обеспечения.

Для предотвращения возможных уязвимостей в разрабатываемых приложениях, было разработано ПО, осуществляющее поиск XSS-уязвимостей в приложениях.

Межсайтовый скриптинг (XSS) – это тип уязвимости программного обеспечения, свойственный web-приложениям (путем обхода ограничений безопасности браузера), который позволяет атакующему внедрить клиентский сценарий в web-страницы, просматриваемые другими пользователями [3].

Непостоянные (отраженные) атаки осуществляются, когда данные, предоставляемые web-клиентом, тут же используются серверными скриптами для генерации страницы с результатами для этого самого клиента [1]. Если пользовательские данные некорректны и содержатся внутри страницы с результатами без кодирования HTML - это позволяет внедриться клиентскому коду в динамическую страницу.

Постоянные (хранимые) атаки обладают наибольшим потенциалом. В этом случае вредоносный код хранится на web-сервисе (в базе данных, файловой системе или в другом месте), а затем отображаются посетителю web-страницы без кодирования с использованием специальных символов HTML [4].

Локальные XSS-атаки или атаки, основанные на DOM заключаются в том, что злоумышленник меняет данные на стороне клиента во время запроса страницы с сервера. Главным отличием уязвимости модели DOM от других типов является то, что сервер не возвращает результатов запроса; наоборот, происходит локальная обработка данных при помощи функций DOM и вредоносный сценарий исполняется с такими же правами, что и web-браузер на машине жертвы атаки.

Для анализа сайта на наличие уязвимостей, на вход разработанному приложению необходимо подать URL исследуемой страницы, HTTP-заголовок в который необходимо произвести инъекцию, куки, URL параметры в которые необходимо внедрить инъекцию и список инъекций с их ожидаемым отображением в DOM-модели страницы. Если необходимо сгенерировать отчет, нужно выбрать этот параметр и указать необходимую директорию, в которой будет сохранен отчет в Excel формате. В отчете указывается количество найденных при сканировании уязвимостей, дата и время выполнения сканирова-

ния, время открытия страницы, страница на которой была найдена уязвимость, место, в которое производилась инъекция (форма ввода данных, URL параметр или HTTP-заголовок), и сама инъекция.

Для поиска отраженных уязвимостей и уязвимостей основанных на DOM, в отправляемые значения внедряется инъекция, а затем анализируется ответ с сервера. Если в ответе содержится отправляемая XSS-инъекция, страница считается уязвимой. Алгоритм поиска хранимых уязвимостей во многом схож с алгоритмом поиска отраженных XSS-уязвимостей, за исключением того, что необходимо производить отправку формы и ждать ответа от сервера. В случае, если скрипт выполнился, страница считается уязвимой.

Данный подход позволяет выявить на тестовых примерах большинство содержащихся уязвимостей. На основе данного подхода было разработано ПО для поиска XSS-уязвимостей. Плюсами разработанного ПО является гибкость в настройках сканирования, простой интерфейс, составление отчетов о найденных уязвимостях и возможность анализа страниц, доступных только авторизованным пользователям.

Список литературы

1. Protecting Your Users Against Reflected XSS. — Текст : электронный // HACKSPLAINING : [сайт]. — URL: https://www.hacksplaining.com/prevention/xss-reflected (дата обращения: 22.05.2021).

2. Web Applications vulnerabilities and threats: statistics for 2019. — Текст : электронный // Positive technologies : [сайт]. — URL: https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020/?sphrase_id=87695 (дата обращения: 22.05.2021).

3. What is and how to prevent Cross-Site Scripting (XSS) | OWASP Top 10 (A7). — Текст : электронный // Hdiv : [сайт]. — URL: https://hdivsecurity.com/owasp-xss (дата обращения: 22.05.2021).

4. Пост-эксплуатация XSS: продвинутые методы и способы защиты. — Текст : электронный // SecurityLab : [сайт]. — URL: https://www.securitylab.ru/analytics/440187.php (дата обращения: 26.05.2021).