# СПОСОБЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ВЕБ-ПРИЛОЖЕНИЙ

**Могильникова Наталья Сергеевна**

магистрант, Пермский государственный аграрно-технологический университет им. академика Д. Н. Прянишникова, РФ, г. Пермь

Быстрое развитие сети Интернет способствовало стремительному росту количества веб-приложений. Повсеместное увеличение количества персональных компьютеров, имеющих доступ к различным вычислительным ресурсам и данным, множество уязвимостей в приложениях, угрозы, которые связаны с искажением, потерей или открытием данных (информации), обращенных или принадлежащих определенным пользователям служат обоснованием актуальности выбранной темы исследования, а именно обеспечения информационной безопасности веб-приложений.

Число компаний, которые применяют веб-технологии для повышения производительности работы и привлечения новых клиентов, растет с каждым годом. Несомненно, интернет-сервисы несут с собой множество преимуществ, но есть и обратная сторона медали - с ростом числа приложений увеличивается и количество кибер-угроз.

Так, компания Symantec в своем отчете Global Internet Security Threat Report (ISTR) указывает, что кибер-преступники при взломе веб-сайтов обычно используют уязвимости веб-приложений, работающих на сервере, или эксплуатируют некоторые уязвимости операционной системы, на которой работают эти приложения. Например, с помощью атак типа XSS хакер может перенаправить запросы пользователей на вредоносные веб-страницы, а с помощью SQL-инъекций - извлекать из баз данных сайта различную конфиденциальную информацию.

В ответ на массовые взломы систем безопасности был создан консорциум OWASP - Open Web Application Security Project, это открытый проект обеспечения безопасности веб-приложений.

Однако и злоумышленники, и специалисты в области кибер-безопасности продолжают находить уязвимости в веб-приложениях, которые могут привести к серьезным потерям со стороны бизнеса. Основной причиной большинства взломов в веб-приложениях является написанный разработчиками программный код.

Далее будут рассмотрены наиболее распространенные виды уязвимостей по версии OWASP:

SQL injections.

Использование SQL дает возможность злоумышленнику выполнить несанкционированное обращение к базе данных с помощью передачи в теле запроса произвольного SQL-запроса.

Наиболее действенные способы борьбы с такого рода уязвимостями:

· всегда проверять валидность входящих данных. Числа должны быть числами, строки – строками;

· фильтровать специальные символы (кавычки, тире).

· не передавать данные прямо в запрос. Использовать подготовленные запросы, а лучше

хранимые процедуры.

· не предоставлять скрипту полный доступ на операции с базой данных.

· не выводить системные сообщения об ошибках запроса. Это усложнит понимание структуры базы данных злоумышленником.

Некорректная аутентификация и управление сессией пользователя.

Злоумышленник может получить возможность перехвата сессии пользователя.

Методы обеспечения информационной безопасности в такой ситуации следующие:

· в случае если cookie-файлы в браузере пользователя отключены, запретить передачу сессии через URL;

· если хранится или передаётся конфиденциальная информация, то применять зашифрованные протоколы с сертификатом SSL;

· при особо важных действиях запрашивать пароль ещё раз;

· своевременно и достаточно часто завершать сессии.

Cross-Site Scripting (Межсайтовый скриптинг).

Основное применение XSS атак - это хищения аутентификационных данных администраторов сайта для доступа к административному разделу, и к данным других пользователей, имеющих личные кабинеты, или персональные доступы к закрытым разделам сайта.

Основным способом борьбы с такого рода атаками на стороне сервера является экранирование специальных символов перед выводом любых данных, которые были получены от пользователей, а также фильтрация входных данных.

Небезопасная конфигурация.

Уязвимости такого рода появляются при неправильно настроенном конфигурационном файле сервера.

Основными способами устранения таких уязвимостей являются:

· обновление программного обеспечения до новейших версий;

· запрет вывода на клиентский терминал необработанных сообщений об ошибках.

Утечка конфиденциальных данных.

Обычно такое происходит при взломе веб-приложения, в том случае, когда данные хранятся в открытом виде. Также утечка таких данных может произойти из-за их передачи в незашифрованном виде.

Способы противодействия:

· хранить конфиденциальную информацию только в зашифрованном виде;

· при передаче конфиденциальных данных использовать зашифрованные протоколы с сертификатом SSL;

· не хранить конфиденциальную информацию без необходимости;

· хранить пароли в хешированном виде, используя для их хеширования специальные

алгоритмы, такие как scrypt, PBKDF2 или bcrypt.

Подделка межсайтовых запросов (CSRF).

Основное применение CSRF - принуждение выполнения каких-либо действий на уязвимой странице от имени администратора или авторизованного пользователя:

· изменение учетных данных доступа (например, пароля администратора);

· восстановление пароля, доступа к почте;

· операции с платежными системами.

Способы противодействия данной атаке:

· использовать одноразовый токен для каждого действия;

· в каждом запросе требовать передачи логина и пароля;

· ограничивать «срок жизни сессии». Данный способ позволит ограничить время, в течение которого можно воспользоваться уязвимостью.

Важный организационный момент при построении системы защиты веб-приложений - тест на проникновение. Именно он станет оптимальным способом проверки защищенности информационной системы с помощью имитации направленных атак. Он дает возможность оценить защищенность информационной системы от несанкционированного воздействия, используя различные модели вторжений. Тест на проникновение для веб-приложений фокусируется только на оценке уровня защиты веб-приложений. Процесс состоит из активного анализа приложений и поиска в них уязвимостей, технических ошибок или других проблем. Информация обо всех слабые места отображается в итоговом отчете.

Таким образом, для обеспечения информационной безопасности веб-приложения необходима комплексная защита, позволяющая учитывать каждую из перечисленных уязвимостей и проведение теста на проникновение в обязательном порядке.

**Список литературы:**

1. Волосенков В.О., Гаврилов А.Д. Анализ уязвимостей компонентов распределенной вычислительной системы и методов её защиты // Проблемы безопасности российского об-щества – 2014 – № 2.

2. Что такое уязвимости? [Электронный ресурс]. – Режим доступа: http://insafety.org/classification_of_vulnerabilities.php/, свободный – Загл. с экрана. (Дата обращения: 14.10.2018).

3. Открытый проект о безопасности веб-приложений – The Open Web Application Security Project (OWASP) [Электронный ресурс]. – Режим доступа: http://owasp.org/, свободный – Загл. с экрана. (Дата обращения: 14.10.2018).