# БЕЗОПАСНОСТЬ ВЕБ-ПРИЛОЖЕНИЙ

**Синегуб Александр Александрович**

бакалавр, Национальный Технический Университет Украины «Киевский Политехнический Институт», Украина, г. Киев

Рост популярности веб-приложений, а также появление новых технологий изменило способ ведения бизнеса, доступа и обмена информацией. Многие компании перенесли большую часть своих действий в интернет, поэтому представители удаленных офисов и деловые сотрудники из разных стран могут сотрудничать и делится конфиденциальными данными в режиме реального времени.

С внедрением в современные веб-приложения Web 2.0 и HTML5 требования клиентов изменились: они хотят иметь доступ к любым данным, которые им нужны двадцать четыре на семь. Такие требования подталкивают предприятия к тому, чтобы сделать эти данные доступными через интернет. Прекрасным примером этого являются онлайн-банкинг (рис. 1) и интернет-магазины (рис. 2).

***Рисунок 1. Онлайн банкинг***

***Рисунок 2. Интернет-магазин***

*Синегуб А.А. Безопасность веб-приложений // Студенческий форум: электрон. научн. журн. 2018. № 14(35). URL: https://nauchforum.ru/journal/stud/35/39486 (дата обращения: 10.09.2025).*

Все эти улучшения также привлекают и хакеров, мошенников, потому что, как и в любой другой отрасли, деньги можно получить незаконно.

Чтобы обеспечить безопасность такого приложения надо определить все недостатки безопасности и уязвимости в самом веб-приложении до того, как хакер определит и использует их. Вот почему очень важно, чтобы процесс выявления уязвимостей выполнялся на все этапах SDLC (Жизненный цикл программного обеспечения), а не когда приложение в режиме реального времени [1].

Существует несколько различных способов выявления уязвимостей. Можно сканировать веб-приложение с помощью сканера белого и черного ящика, выполнить аудит исходного кода для определения проблем с кодированием или выполнить ручную проверку безопасности и проверке проникновения.

Нет единого метода защиты, который можно использовать для выявления всех уязвимостей приложения. Каждый из упомянутых выше имеет свои плюсы и минусы.

Например, в то время как автоматизированный инструмент обнаруживает почти все технические уязвимости, он не может определить их в логике. Уязвимости логики можно определить только с помощью ручной проверки. С другой стороны, ручная проверка неэффективна и может занять значительное количество времени и обойтись в целое состояние. Тестирование белых ящиков осложнить процедуры разработки и может быть сделано только разработчиками, имеющими доступ к коду [2].

Если бюджет и время позволяют, рекомендуется использовать различные доступные средства и способы тестирования. Существует множество факторов, которые влияют на решение при выборе сканера уязвимостей веб-приложений. Первый очевидный: следует ли использовать коммерческое программное обеспечение или использовать бесплатное некоммерческое решение. Рекомендуется коммерческое по нескольким причинам: частые обновления программного обеспечения и проверки безопасности, простота использования, профессиональная поддержка и т. д.

Лучшим подходом для определения правильного сканера является запуск нескольких проверок с использованием разных сканеров.

Второй фактор - возможность идентифицировать поверхности атаки. Что бы определить сканер, который имеет возможность идентифицировать все поверхности атаки нужно сравнить список страниц, каталогов, файлов и входных параметров и посмотреть какой из них идентифицировал все параметры лучше. Если конкретный сканер не смог правильно сканировать веб-приложение, это может также означать, что его, возможно, необходимо настроить, что приводит к следующему фактору.

Третий фактор - простота в использовании веб-сканера безопасности. В то время как некоторые сканеры черного ящика могут сканировать практически любой тип веб-сайта, используя настройки из коробки, некоторые могут быть настроены перед запуском сканирования.

Поскольку безопасность веб-приложений является нишевой отраслью, не все компании будут иметь специалистов по веб-безопасности, которые смогут понять и сконфигурировать сканер уязвимостей. Поэтому стоит обратить внимание на простой в использовании сканер, который может автоматически обнаруживать и адаптироваться к большинству распространенных сценариев.

Простые в использовании сканеры будут иметь лучший возврат инвестиций, потому что не нужно нанимать специалистов или обучать членов команды их использованию.

Следующий фактор, используемый при выборе такого сканера - который из сканеров может идентифицировать большинство уязвимостей, которые не являются ложными. Бывает такое, что сканер выявил 100 уязвимостей, но большинство из них являются ложными.

Если сканер сообщает о множестве ложных ошибок, сотрудники по безопасности проведут больше времени проверяя результаты, а не сосредотачиваясь на исправлениях, поэтому стараются избегать этого.

Далее следует обратить внимание на способность автоматизации. Чем больше сканер может автоматизировать, тем лучше. Например, представьте себе веб-приложение с сотней видимых полей вводы. Если тест проникновения будет тестировать каждый вход в веб-приложении, ему нужно будет запустить около восьми ста различных тестов.

Если каждый из тестов займет примерно 3 минуты и если все работает плавно, такой тест продлится около 18 дней, если будет работать круглосуточно. И это касается только лишь видимого функционала, а как насчет скрытого. Как правило, в веб-приложении, скрыто происходить гораздо больше, чем то, что можно увидеть.

Поэтому автоматизация - еще одна важная функция. Автоматизация теста безопасности будет стоить меньше, а выполнятся более эффективно.

Безопасность веб-приложений это то, на что нужно ориентироваться на каждом этапе разработки. Чем раннее будет проводится тестирование, тем более безопасным будет приложение и тем дешевле и проще будет устранять выявившиеся проблемы на последующих этапах.

Например, сканер безопасности автоматизированного веб-приложения может использоваться на всех этапах жизненного цикла разработки программного обеспечения(SDLC). Даже когда приложение находится на первых этапах разработки, когда у него есть всего несколько входных данных. Тестирование на первых этапах разработки очень важно, поскольку, если такие входы являются основой последующей разработки, будет очень тяжело обеспечить их безопасность не переписав весь код приложения.

Есть также несколько других преимуществ использования такого сканера на всех этапах SDLC. Например, программисты автоматически обучатся написанию более качественного кода, потому что кроме определения уязвимостей большинство сканеров также предоставляют практическое решение об устранении выявленной проблемы. Это помогает им узнать больше о написании безопасных веб-приложений.

**Список литературы:**

1. Web application security [Электронный ресурс] - URL: https://en.wikipedia.org/wiki/Web_application_security (Дата обращения 16.07.2018).

2. Web application security scanners: How effective are they? [Электронный ресурс] - URL: https://searchsecurity.techtarget.com/Web-application-security-scanners-How-effective-are-they (Дата обращения 17.07.2018).