**IT-Колледж “Сириус”**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ДОКЛАД**

по дисциплине “Введение в специальность”

на тему “Информационная безопасность веб-приложений”

Выполнил:  
Студент группы

1.9.7.2  
Колкарева Даяна Андреевна

Принял:

Старший преподаватель  
Тенигин Альберт Андреевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IT-Колледж “Сириус”  
2022

[**1. Введение:**](#_6bqgk5qcrarf) **2**

[A. Что такое информационная безопасность?](#_mlkhqgn7cz9x) 2

[B. Что такое веб-приложения.](#_pa1m55n812du) 4

[**2. Безопасность веб-приложений**](#_b1astubqi9tg) **6**

[A. Методы взлома](#_dttz3h03v8a) 7

[B. Методы защиты](#_gn5iqjkfnke6) 9

[**3. Вывод**](#_o6ys77genfwn) **11**

[**4. Список используемой литературы**](#_mxpfmyh4fu00) **12**

# 1. Введение:

**Информационная безопасность веб-приложений** — это защитные меры, при которых злоумышленник не сможет получить доступ к конфиденциальным данным. Не важно, при попытке взлома извне или внутри компании через нелегитимный, несанкционированный доступ.

Если же вдруг взломщиком окажется человек с определенными привилегиями, то он не сможет воспользоваться конфиденциальными данными, потому что его поведение будет отличаться от “типичного” и расценено как аномалия, следовательно, об этом сразу же будет оповещена служба информационной безопасности.

Веб-приложения становятся привлекательными с финансовой точки зрения не только для их разработчиков, но и также для тех, кто хочет нелегально воспользоваться хранящимися в них данными. Разновидности и количество атак на них растут в геометрической прогрессии.

Понятия информационная безопасность и веб-приложения очень обширны сами по себе и для того, чтобы рассмотреть их вместе, для начала необходимо углубиться в каждый из них отдельно.

## A. Что такое информационная безопасность?

**Информационная безопасность** — набор или совокупность способов, применяемых вне зависимости от формы, которую могут принимать данные, и необходимых для предотвращения нелегального, несанкционированного доступа, использования, раскрытия, искажения, изменения, исследования, записи или уничтожения информации. Другими словами, это набор технологий, стандартов и методов, используемых для защиты данных и информационной системы.

Основная **цель обеспечения информационной безопасности** заключается в том, чтобы защитить данные и систему от любых видов и форм вмешательства, которые могут стать причиной потери или изменения данных.

**Угрозы информационной безопасности** можно разделить на следующие:

* Внутренние (источники угрозы, которые находятся внутри системы).
* Внешние (источники угроз за пределами системы)
* Естественные (не зависящие от человека: пожары, наводнения, ураганы, другими словами, природные катаклизмы).
* Искусственные (происходящие из-за человека), которые также делятся на *непреднамеренные* (совершенные людьми по неосторожности, незнанию, неопытности и т.д.) и *преднамеренные* (совершаемые людьми с какой-то целью, специально: хакерские атаки, противоправные действия конкурентов или месть сотрудников).

Исходя из того, что угрозы по-разному воздействуют на систему, их делят на пассивные, то есть не изменяющие структуру и содержание информации, и активные, которые меняют структуру и содержание системы, например, специальные программы.

Наиболее опасными являются преднамеренные угрозы, которые все чаще пополняются новыми разновидностями, что связано, в первую очередь, с распространением электронных платежей.

Злоумышленники не стоят на месте, а ищут новые пути добраться до данных пользователей и компаний.

Чтобы обезопасить компанию от потери денежных средств и интеллектуальной собственности, необходимо уделять больше внимания информационной безопасности. Это возможно благодаря средствам защиты информации в лице передовых технологий.

Для оптимальной работы систем информационной безопасности предприятия необходимо: ввести контроль, чтобы гарантировать уровень безопасности данных; обеспечить последовательность внутренней и внешней корпоративной информации, а также гарантию предотвращения искажения информации; обеспечить надежный и эффективный доступ к информации уполномоченных лиц, предсказуемость сетевой среды, чтобы можно было получить доступ к информации и данным, когда это необходимо, а также возможность восстановления данных и системы после ее сбоя так, чтобы это не влияло на работу отрицательно. Проще говоря, главными являются три фактора: конфиденциальность, целостность и доступность [1.1].

## B. Что такое веб-приложения.

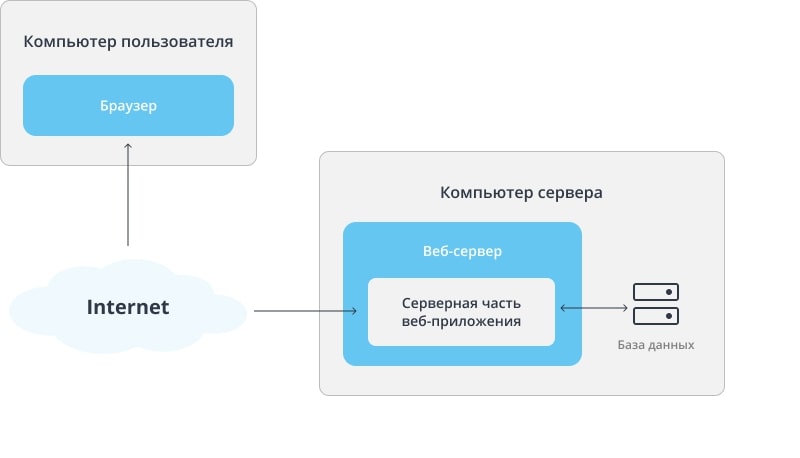
**Веб-приложение** — это программа, не требующая установки на устройство, так как пользователь получает доступ к нему через интернет. Веб-приложение является интерактивной компьютерной программой, что дает пользователям возможность взаимодействовать с разными элементами: например, отправить заявку на покупку какого-нибудь товара, оставить комментарии, поставить лайк (или дизлайк, если так хочется), написать пост, общаться в чатах и так далее.

Существуют различные классификации веб-приложений: в зависимости от их функционала и назначения. Выделяют три основных шаблона построения сайтов:

* **MPA** (multi-page application): многостраничное приложение, которое отправляет запрос на сервер и полностью обновляет страницу, когда с ней совершается действие;
* **SPA** (single-page application): одностраничное приложение, содержащее HTML-страницу, которая динамически обновляется в зависимости от действий пользователя — без полной перезагрузки;
* **PWA** (progressive web application): приложение, которое пользователь устанавливает и может использовать в режиме офлайн.

Веб-приложения работают по принципу “клиент-сервер”. В данном случае клиент *-* это *браузер*,который связывается с *веб-сервером*. И после того, как пользователь отправит свой запрос, у него формируется содержание приложения.

Принципы работы веб-приложений:

* **Статические страницы**. Пользователь делает запрос в браузере, а веб-сервер обрабатывает его и отправляет в ответ заранее созданную веб-страницу.
* **Динамические страницы.** Пользователь делает запрос в браузере, но он не отправляется напрямую от веб-сервера браузеру, а сначала направляется на сервер приложения, где считывается код и подбираются данные для формирования страницы, и только потом уже страница отправляется на веб-сервер, а после в браузер (рис.1).  рис.1

# 2. Безопасность веб-приложений

Вот уже второе десятилетие, веб-приложения по-прежнему являются самой привлекательной мишенью для хакеров. Они были и остаются первоочередной целью при взломе внешнего периметра компании, как в тестах, так и в жизни.

Слабая безопасность веб-приложений вызвана множеством факторов: от качества кода и языка разработки, до настроек сервера. Большая проблема еще и то, что безопасность веб-приложений отстранена от общей информационной безопасности. И к сожалению, больше половины ИБ подразделений в российских компаниях даже не подозревают, кто ответствечает за официальный сайт компании, не говоря уже о том, кто занимается обеспечением его безопасности. Поэтому веб-приложения и подвергаются взлому так часто.

Проводимые пентесты - тесты моделирующие атаки - подтверждают эти доводы. Так пентестеры используя уязвимости веб-приложений могли выполнять команды внутри операционной системы.

Большинство уязвимостей в веб-приложениях находятся в коде, их называют уязвимостями нулевого дня, потому что у разработчиков было 0 дней на исправление. Это все потому, что уязвимости для каждого приложения являются уникальными и никогда не были замечены ранее. Лучшая защита от подобных атак — продуманная система безопасности, заложенная еще на этапе разработки. Разработчики должны понимать как работают различные виды атак и уметь придумывать защиту от них.

## A. Методы взлома

Оповещение разработчиков об обнаруженных уязвимостях является основной целью проекта Open Web Application Security Project (OWASP) - списка самых популярных / часто замеченных уязвимостей. Данный список обновляется раз в три года, последний раз обновлялся в 2021 году.

Команда компании IBM выделила самые распространенные виды атак основываясь на отчётах OWASP:

1. **Инъекции**

Инъекции позволяют хакерам изменить запрос бэкэнда команды через несанкционированный ввод данных пользователем. Говоря простыми словами SQL инъекция - изменение SQL команды и использование её в произвольных целях.

2. **Повреждение аутентификации и управление сеансами**

Повреждение аутентификации - взлом системы входа на сайт. Может быть произведен как грубым брутфорсом (простым перебором паролей), так и более хитрыми методами.

3. **Межсайтовый скриптинг**

Межсайтовый скриптинг - уязвимость, позволяющая злоумышленникам вставить JavaScript код на страницу. Если вредоносный код выполнился успешно, они могут полностью изменить содержимое сайта, отправить учетные данные пользователей на любой другой сервер и.т.п.

4. **Незащищенные прямые ссылки на объект**

Изменяя имя файла в ссылке можно получить доступ к базе данных. Так при загрузке файла на сайт можно его взломать.

5. **Неправильная настройка безопасности**

Просто неверная настройка сервера.

6. **Чувствительная экспозиция данных**

Отсутствие шифрования данных во время перемещения и в состоянии покоя. Если веб-приложения, которые вы используете не защищают или защищают неправильно личные данные, например, кредитные карты или данные аккаунта, хакеры могут украсть или изменить их.

7. **Отсутствие функции контроля уровня доступа**

Вместо разделения уровня доступа пользователей (на простых юзеров и администраторов) стоит простая заглушка которая “прячет” информацию для админов.

8. **Межсайтовая подделка запроса**

Эта атака используется в сочетании с социальными проектами. Атака направлена на пользователей, опирается на недостатки HTTP (протокол прикладного уровня передачи данных). Когда юзер заходит на сайт, созданный злоумышленником, от его лица скрытно посылается запрос на другой сервер, осуществляющий вредоносную операцию.

9. **Использования компонентов с уже известными уязвимостями**

Эта уязвимость включает в себя различные приложения, которые продолжают использовать компоненты даже после обнаружения проблемы в них. Злоумышленники с легкостью могут использовать уязвимости устаревших компонентов на приложения, которые их используют, так как эти уязвимости были уже давно общедоступны. Любой скрипт-кидди (начинающий хакер пользующийся инструментами находящимися в общем доступе, не понимающий при этом механизм их действия) может произвести такой взлом.

10. **Непроверенный переход и редирект**

Такие уязвимости используются в фишинговых атаках, в которых пользователей хитростью перенаправляют на вредоносный сайт. Хакеры могут управлять URL-ми реального сайта так, чтобы перенаправить жертву на нужную им страницу. [2.1]

Говоря про уязвимости внутри web-приложений, обычно идет речь о проблемах на уровне кода или результате ошибок в его написании. При этом такие ошибки могут быть легко замечены ранее внутри архитектуры приложения. Обычно разработка архитектуры приложения приводит либо к большому, либо к малому количеству брешей в защите, в зависимости от того как системы защиты сформированы и распределены по коду.

Из-за этого способность находить слабые места в обороне приложения очень полезная техника. Плохо разработанные функции будут первой целью на пути к поиску брешей. [2.3]

## B. Методы защиты

**Прямой способ защиты приложений** — межсетевой экран или брандмауэр. Для большего числа веб-приложений применяется **Web Application Firewall** (WAF).

Можно использовать приложения и фреймворки для автоматического обнаружения брешей в системе (такими пользуются белые хакеры):

* OpenVAS сканирует узлы сети на наличие уязвимостей и позволяет управлять уязвимостями.
* OWASP Xenotix XSS Exploit Framework сканирует ресурс на возможность эксплуатации XSS-уязвимостей.
* Approof от Positive Technologies проверяет конфигурацию веб-приложения, сканирует на наличие уязвимых компонентов и вредоносного кода.

Также можно (а на сегодняшний день обязательно) использовать **HTTPS** (HyperText Transfer Protocol Secure) - расширение к уже упомянутому HTTP для шифрования данных пользователей при их передаче. HTTPS гарантирует целостность и конфиденциальность взаимодействия с сервером.

Обязательным при обеспечении безопасности является - **обновление ПО.** Так как с новыми обновлениями всегда приходят правки безопасности. А долго пользоваться одной и той же версией программного обеспечения очень рискованно.

Ну и нельзя не упомянуть такие методы как: защита от SQL инъекций, нежелательного редиректа, шифрование паролей, контроль вводимых сообщений и загружаемых файлов, а также распределение прав доступа к тому или иному функционалу. [2.2]

# 3. Вывод

Таким образом - основным параметром обеспечения безопасности web-приложения является простая внимательность к деталям. При ответственном подходе к вопросу, своевременном обновлении ПО и регулярном проведении пентестов уровня защиты будет достаточно для отражения большей части атак.

# 4. Список используемой литературы

1.1: Информационная безопасность // pirit URL: <https://pirit.biz/reshenija/informacionnaja-bezopasnost> (дата обращения: 30.10.22).

1.2: Веб-приложение: что это, виды и преимущества разработки // Azoft URL: <https://www.azoft.ru/blog/web-apps/> (дата обращения: 30.10.22).

2.1: 10 атак на веб-приложения в действии // Хабр URL: <https://habr.com/ru/company/ua-hosting/blog/272205/> (дата обращения: 31.10.22).

2.2: Как защитить веб-приложение: основные советы, инструменты, полезные ссылки // tproger URL: <https://tproger.ru/translations/webapp-security/> (дата обращения: 1.11.22).

2.3: Andrew Hoffman Web Application Security. - Gravenstein Highway North, Sebastopol: O’Reilly Media, 2020-06-05. - 331 с. // URL: <https://t.me/progbook2/1668>

Дополнительные источники:

* Хроники пентестера // Хабр URL: <https://habr.com/ru/post/128943/>
* Безопасность веб-приложений // Хабр URL: <https://habr.com/ru/amp/post/654783/>
* Информационная безопасность web-приложений - современные решения // gardatech URL: [https://gardatech.ru/articles/smi/informatsionnaya-bezopasnost-web-prilozheniy-sovremennye-resheniya](https://gardatech.ru/articles/smi/informatsionnaya-bezopasnost-web-prilozheniy-sovremennye-resheniya/)/
* рис.1: <https://www.azoft.ru/wp-content/uploads/2021/09/1.jpg>