Список литературы

1. Правило близости и схожести [Электронный ресурс]. URL: https://typographyhandbook.com/ (дата обращения: 15.10.2023).

2. Иерархия [Электронный ресурс]. URL: https://vanseodesign.com/web-design/visual-hierarchy/ (дата обращения: 25.10.2023).

3. Шрифтовая пара [Электронный ресурс]. URL: https://www.learnui.design/blog/guide-pairing-fonts.html (дата обращения: 18.10.2023).

4. Ритм [Электронный ресурс]. URL: https://www.thoughtco.com/rhythm-design-principle-3470054#:~:text=As%20a%20principle%20of%20design%2C%20rhythm%20is%20also,Our%20senses%E2%80%94and%20therefore%2C%20the%20brain%E2%80%94respond%20to%20rhythm%20positively (дата обращения: 17.10.2023).

ПОДХОД К РЕАЛИЗАЦИИ ВЕБ-ПРИЛОЖЕНИЙ НА ОСНОВЕ КЛИЕНТ-СЕРВЕРНОЙ АРХИТЕКТУРЫ, ОБЕСПЕЧИВАЮЩИЙ ЗАЩИЩЕННЫЙ ОБМЕН ДАННЫМИ

Сверчков Р.В.

ФГБОУ ВО «Российский экономический университет имени Г.В. Плеханова», Москва, e-mail: rvsverchkov@gmail.com

В настоящее время в работе бизнеса различных отраслей стали очень распространены веб-приложения [2, с. 39], которые во многом повторяют логику и устройство аналогичных продуктов компаний, разработанных для какой-либо конкретной операционной системы. Такая ситуация также наблюдается во многих сферах, особенно там, где требуется обеспечение доступа к разработанному программному обеспечению практически с любого устройства, имеющего выход в интернет. Однако из-за такого подхода по созданию полностью аналогичных приложений не учитываются многие особенности работы интернет-браузеров, которые в свою очередь создают множество потенциальных возможностей для воздействия на данные со стороны злоумышленников [1, с. 60], а также необходимость создания корректного контроля доступов, тем самым ограничивая область действий потенциального пользователя. Данная проблема является как никогда актуальной сейчас, поскольку теперь практически любое программное обеспечение имеет свой аналог в виде веб-приложения и многие из них подвержены воздействию злоумышленников.

Текущие проблемы при использовании и разработки веб-приложений

Из-за того, что при разработке веб-приложений зачастую не учитывают их ключевые особенности, а именно то, что в силу своего строения доступ к ним может быть получен буквально с любого устройства, имеющего интернет-браузер, и у компаний, которые решают проводить полномасштабную интеграцию таких решений в свою корпоративную сеть, может возникнуть множество неочевидных проблем.

Одной из таких проблем является некорректная настройка маршрутизации в итоговом продукте, позволяющая потенциальным пользователям, не имеющим необходимого уровня доступа с помощью прямых ссылок просматривать ресурсы, доступ к которым для них не предусмотрен. Такие ошибки в проектировании практически всегда достаточно дорого обходятся компании, если доступ к таким ресурсам был получен недобросовестным пользователем, который в свою очередь можем поделиться данной уязвимостью с третьими лицами и тем самым подвергнуть колоссальному риску расположенную там информацию.

Вместе с этим также стоит упомянуть, что немаловажную опасность представляет собой недостаточная фильтрация входящей информации в базу данных, расположенную на сервере. Это связано в первую очередь с тем, что потенциальный нарушитель способен ввести в форму какой-либо SQL-запрос и получить в ответ необходимую конфиденциальную информацию из базы. Такая атака называется инъекцией, и она может причинить большой ущерб практически любому продукту, в частности если соответствующая фильтрация или валидация данных вовсе отсутствует.

Также имеется риск некорректной проектировки архитектуры веб-приложения, как пример разрешение пользователям загружать файлы любого формата без какой-либо проверки со стороны клиента, тем самым повышается шанс эксплуатации такой уязвимости со стороны злоумышленников, способных развернуть на стороне сервера исполняемый файл с каким-либо вредоносным кодом.

Стоит также упомянуть, что в большинстве случаев во время разработки веб-приложений редко уделяется достаточное время построению безопасной конфигурации самого приложения и из этого как раз вытекает множество проблем, напрямую связанных с общим уровнем безопасности. В такой ситуации, когда текущие настройки веб-приложения или же сервера не отвечают должным требованиям безопасности, колоссальным образом возникает риск утечки данных или же вовсе компрометации связок логинов и паролей для доступа к административным панелям со стороны серверной части или чего-либо еще.

Использование уязвимых или же вовсе устаревших компонентов также может сильно повлиять на общий уровень веб-приложения. Такая проблема возникает, когда в разработке применяются непроверенные фреймворки или же сторонние решения, уже имеющие в своем исходном коде ошибки безопасности. Злоумышленники же могут воспользоваться достаточно распространенными инструментами, позволяющими

производить поиск и успешно находить некорректно сконфигурированные веб-приложения, например поисковая система Shodan IoT.

Нельзя также не отметить, что ошибки идентификации и аутентификации являются в настоящее время одной из ведущих проблем в области разработки веб-приложений, поскольку многие команды разработки не придают должного значения выстраиванию корректного процесса авторизация и аутентификации пользователей приложения. Возможность установки «слабого пароля», как например «123456», и отсутствие ограничений на количество попыток входа может потенциально создать множество проблем для общего уровня защищенности веб-приложения, поскольку злоумышленнику не составит труда методом перебора подобрать корректный пароль и тем самым скомпрометировать чьи-либо конфиденциальные данные. Тем более такая ситуация может усугубляться, если реализованное приложение обращается с финансовыми операциями или же вовсе повторяет логику работы уже существующего продукта и имеет достаточно большого объема базу данных со сведениями о пользователях. Вместе с этим также стоит отметить, что далеко не всегда в веб-приложениях со стороны интерфейсы производится хеширование и обезличивание логинов, чтобы в случае компрометации базы данных не получилась такая ситуация, когда злоумышленники будут иметь доступ к множеству аккаунтов.

Ошибки логирования и мониторинга безопасности могут потенциально создать множество проблем для конечного продукта, размещенного в сети интернет, поскольку если система не производит корректной фиксации аномальных событий, напрямую касающихся безопасности, или же вовсе такие механизмы отсутствуют. Данная проблема является одной из наиболее критичных во многом из-за того, что если мониторинг безопасности работает некорректно или же отсутствует вовсе, то из-за этого атаки злоумышленников могут оказаться незамеченными. Это в свою очередь снизит вероятность оперативного реагирования на возникающие инциденты в реальном времени и обнаружение угроз в целом. Пример такой ситуации может послужить ситуация, когда при многократных попытках неудачной аутентификации не происходит логирования и каких-либо ограничений со стороны интерфейса для предотвращения подбора пароля, и тем самым у злоумышленника появляется множество возможностей для реализации атак.

Нельзя также не отметить, что всегда существует вероятность подделки запросов на стороне сервера (server-side request forgery, SSRF), то есть, когда злоумышленник заставляет сервер отправлять запросы к каким-либо внутренним ресурсам или же вовсе внешним сайтам. Такой подход достаточно часто применяется потенциальными злоумышленниками, для проведения атак на внутренние ресурсы, доступ к которым извне ограничен. Как пример, если серверная часть веб-приложения не имеется достаточной защиты от SSRF, злоумышленник потенциально может заведомо ввести зловредный URL-адрес, заставляющий сервер произвести отправку запроса к собственным внутренним ресурсам и в последствии получить оттуда конфиденциальные данные.

Подход к реализации веб-приложений, обеспечивающий защищенный обмен данными

Учитывая особенности и тенденции развития современных веб-приложений можно выработать следующий подход, включающий в себя следующие аспекты, которые в свою очередь должны быть обязательно проверены еще на этапе разработки программного обеспечения, чтобы обезопасить его и предотвратить множественные ошибки в последствии.

В первую очередь необходимо проводить регулярный мониторинг маршрутизации, а также проектировать контроль уровней доступа на основе принципа наименьших привилегий, чтобы пользователи имели только права, необходимые для выполнения их непосредственных задач. Также необходимо проводить корректную маршрутизацию внутри веб-приложения с целью недопущения доступа недобросовестных пользователей к конфиденциальным данным.

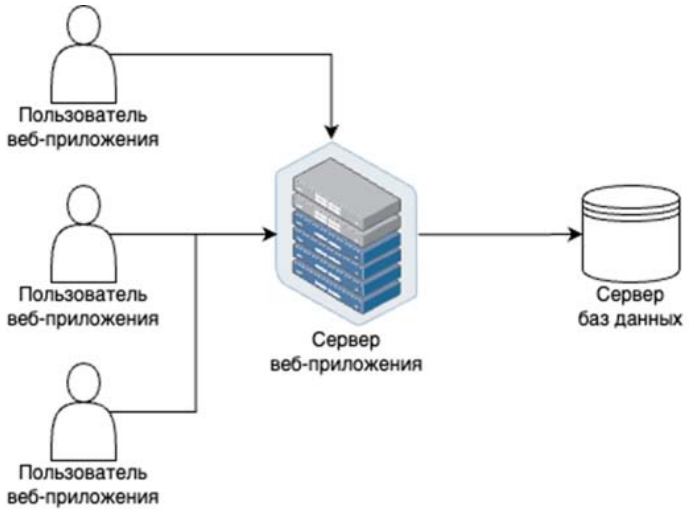
Для предотвращения эксплуатации уязвимостей, связанных с SQL-инъекциями [3, с. 66], необходимо применять на серверной части следующие подходы, а именно:

• Использовать параметризированные запросы или же ORM (object-relational mapping) для проведения работы с базой данных;

• Проводить валидацию и корректную фильтрацию входных данных;

• Применение во всем продукте принципа наименьших привилегий, тем самым ограничивая права доступа к базе данных тем сотрудникам, которым эти права не являются необходимыми для выполнения должностных обязанностей.

Следующим по важности шагом для обеспечения должного уровня защищенности в приложении является корректная настройка полного перечня компонентов, используемых в веб-приложении, а также установка разграничения прав доступа внутри системы, чтобы избежать компрометации ценной информации или же ее конфиденциальности. В случае же недостаточно защищенного простого приложения может хватить и базовых мер, описанных в разделе о клиент-серверной архитектуре, однако если приложение потенциально будет взаимодействовать с конфиденциальной информацией и будет являться высоконагруженным, то таких мер не хватит для защиты в полной мере.



Трехуровневая архитектура защиты данных

Вместе с этим также нельзя не отметить, что аутентификация [4, с. 147] пользователей является слабым звеном любых приложений, в том числе и реализованных с помощью веб-технологий. Именно поэтому при современном проектировании таких решений учитывается то, что за всю отображаемую логику и общее поведение пользователя будет отвечать исключительно серверная часть, а клиент будет служить лишь прослойкой между непосредственным пользователем и сервером. Данная особенность является несомненным преимуществом веб-решений, поскольку при авторизации пользователя в системе проверка пароля на корректность и общее соответствие прав будет происходить на удаленном сервере, а клиент в свою очередь лишь отобразит результат такой проверки и в случае, если все в порядке, то пропустит пользователя дальше. Лишь при таком условии реализации приложений получается добиться максимального уровня защищенности.

Также должен активно применяться и быть выстроен процесс безопасной разработки [6, с. 101] программного обеспечения с повсеместным использованием различных анализаторов кода, позволяющих проверить весь исходных код на наличие различного вида ошибок, потенциально способных привести к появлению уязвимостей.

Однако в случае, если архитектура [5, с. 233] уже была выстроена ранее, и компания не имеет возможности вносить изменения в уже написанный код, то наиболее эффективным решением будет являться применение WAF (Web Application Firewall). В случае его применения HTTP-трафик от потенциальных пользователей веб-приложения будет идти не напрямую к серверной части, а через WAF, где он будет подвергаться декодированию и множеству проверок на наличие атак. В пассивном же режиме будут доступны лишь функции мониторинга и создания оповещений об атаках.

Стоит также упомянуть о том, что WAF может содержать в себе различные модули, в том числе и модули, позволяющие проводить динамический анализ потенциальных уязвимостей защищаемого веб-приложения, тем самым обеспечивая еще более полноценную защиту для конфиденциальных данных и общего состояния системы.

Как итог, в современном защищенном веб-приложении должна применяться трехуровневая архитектура защиты данных и вышеперечисленные меры, в полной мере обеспечивающие решение большинства проблем, связанных с безопасностью и рисками утечки данных (рисунок).

Заключение

Таким образом, были рассмотрены основные проблемы при разработке веб-приложений в настоящее время, а также, с учетом тенденций развития и общего уровня защищенности большинства приложений в интернете, был предложен современный подход к реализации веб-приложений на основе клиент-серверной архитектуры, обеспечивающий защищенный обмен данными.

Описанный подход в свою очередь позволяет избежать большинства уязвимостей, которые могут возникнуть в процессе разработки продуктов, а также позволить обеспечить безопасность на всех этапах производства программного обеспечения, рассчитанного на работу в сети интернет в качестве веб-приложения.

Список источников

1. Яремчук С. Как повысить безопасность веб-приложений // Системный администратор. 2006. № 39. С. 60-64.

2. Кинтонова А.Ж., Баенова Г.М., Урынбасарова А.Ж. Вопросы безопасности веб приложений // Colloquium-journal. 2020. № 65. С. 38-39.

3. Быков М.Ю., Звягинцева А.В. Анализ актуальных угроз безопасности веб-приложений // Современные технологии обеспечения гражданской обороны и ликвидации последствий чрезвычайных ситуаций. 2019. № 10. С. 65-67.

4. Благоразумов А.К., Черников П.Е., Глухов Г.Е., Сёмин А.В. Методы обеспечения безопасности веб-приложений // Научный вестник ГОСНИИ ГА. 2022. № 41. С. 144-152.

5. Петрова А.Г., Кириллин Д.В., Бускаров В.В. Архитектура веб-приложений // Актуальные научные исследования в современном мире. 2021. № 78. С. 233-237.

6. Юрочкин А.Г., Жулябин Д.Ю. Разработка современной архитектуры веб-приложения для решения корпоративных задач // Вестник Воронежского института высоких технологий. 2018. № 25. С. 101-106.

ПРЕИМУЩЕСТВА И НЕДОСТАТКИ ИСПОЛЬЗОВАНИЯ ТЕХНИЧЕСКИХ И ИНФОРМАЦИОННЫХ РЕСУРСОВ В ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ

Скребина И.С., Аветисян Т.В.

Колледж Воронежского института высоких технологий, Воронеж, e-mail: vtatyana_avetisyan@mail.ru

Информационные ресурсы – это различные источники информации, опубликованные и предоставленные в цифровой форме. Они могут включать в себя веб-сайты, базы данных, электронные библиотеки, электронные журналы, электронные книги и др.

Использование технических и информационных ресурсов в профессиональной деятельности становится все более неотъемлемой частью нашей современной жизни. В эру цифровой трансформации и быстрого развития технологий, доступ к информации и использование технических средств стали ключевыми факторами успеха в различных профессиональных областях [1].

Технические ресурсы, такие как компьютеры, программное обеспечение, сети связи и другие устройства, облегчают выполнение задач и увеличивают производительность. Они помогают автоматизировать процессы, ускоряют передачу информации и обеспечивают доступ к огромным объемам данных.

Информационные ресурсы играют важную роль в различных сферах деятельности, включая образование, науку, бизнес, государственное управление и личную жизнь. Они предоставляют огромное количество знаний и информации, необходимых для эффективной работы в различных профессиональных сферах. Интернет, электронные базы данных, научные журналы и другие информационные источники дарят нам возможность получать актуальные и достоверные данные, анализировать их и применять в своей работе.

Благодаря постоянному развитию технологий и доступности информации, специалисты получают уникальные возможности для эффективного выполнения своих профессиональных обязанностей.

Информационные ресурсы используются во многих областях и сферах деятельности, например [2,3]:

1. Информационные ресурсы, такие как электронные учебники, онлайн-курсы и видеолекции, позволяют студентам и учащимся получать доступ к актуальной и разнообразной информации для обучения.

2. Информационные ресурсы используются в бизнесе для сбора, хранения и анализа данных, а также для выполнения и автоматизации различных бизнес-процессов. Это может включать такие ресурсы, как базы данных, отчеты, аналитические инструменты и программное обеспечение для управления информацией о клиентах.

3. Исследователи используют информационные ресурсы для доступа к актуальным научным статьям, журналам и базам данных, которые помогают им получать и оценивать новые знания и результаты исследований.

4. Информационные ресурсы используются в медицинской сфере для доступа к медицинским журналам, базам данных, медицинским историям пациентов и программному обеспечению для управления здравоохранением и диагностики.

5. Информационные ресурсы, такие как видеохостинги, потоковые сервисы и социальные сети, позволяют людям получать доступ к музыке, фильмам, шоу, играм и другим формам развлечений.

6. Информационные ресурсы используются для защиты и обеспечения безопасности информации, включая меры по предотвращению несанкционированного доступа, хранения и передачи данных.

Преимущества использования технических и информационных ресурсов в профессиональной деятельности [4,5]:

- Технические и информационные ресурсы позволяют автоматизировать и ускорить выполнение задач, что приводит к повышению производительности и сокращению времени, затраченного на выполнение задач;

- Возможность получать доступ к большому объему информации из различных источников. Это позволяет быстро находить необходимые данные и использовать их для принятия решений;

- Облегчают коммуникацию и сотрудничество между коллегами и клиентами. Электронная почта, видеоконференции и другие инструменты позволяют обмениваться информацией и идеями в режиме реального времени, независимо от географического расположения.

Недостатки использования технических и информационных ресурсов в профессиональной деятельности [4]:

- Использование технических и информационных ресурсов подразумевает наличие соответствующего оборудования и программного