



Уязвимости и угрозы веб-приложений в 2019 году

Содержание

Тенденции	3
Анализ защищенности веб-приложений	4
Самые распространенные уязвимости	5
Анализ угроз	8
Сравнение тестовых и продуктивных систем	10
Анализ защищенности методом белого ящика	11
Заключение	12
Портрет участников	13
Методика	14

Г Резюме

Уровень защищенности веб-приложений продолжает постепенно расти, но все еще остается довольно низким.

Про веб-приложения:

- **В 9 из 10 веб-приложений преступники могут проводить атаки на пользователей.** В том числе — перенаправлять клиентов на подконтрольный им ресурс, похищать учетные данные с помощью фишинговых атак, заражать компьютер вредоносным ПО.
- **Несанкционированный доступ к приложению возможен на 39% сайтов.** Кроме того, в 2019 году полный контроль над системой был получен в 16% веб-приложений, а в 8% систем полный контроль над сервером веб-приложения позволял проводить атаки на локальную сеть организации.
- **Угроза утечки важных данных присутствует в 68% веб-приложений.** Среди «утекших» данных на первом месте персональные (47% утечек), а на втором — учетные (31%).

Про уязвимости:

- **82% уязвимостей содержались в коде приложения.**
- **Число уязвимостей, которое в среднем приходится на одно веб-приложение, снизилось по сравнению с 2018 годом в полтора раза.** В среднем на одну систему приходятся 22 уязвимости, четыре из которых имеют высокий уровень риска.
- **Каждая пятая уязвимость — высокого уровня риска.**

Тенденции

В 2019 году значительно (на 17 процентных пунктов по сравнению с 2018 годом) снизилась доля веб-приложений, содержащих уязвимости высокого уровня риска. Число критически опасных уязвимостей, которое в среднем приходится на одно приложение, снизилось по сравнению с прошлым годом почти в полтора раза.

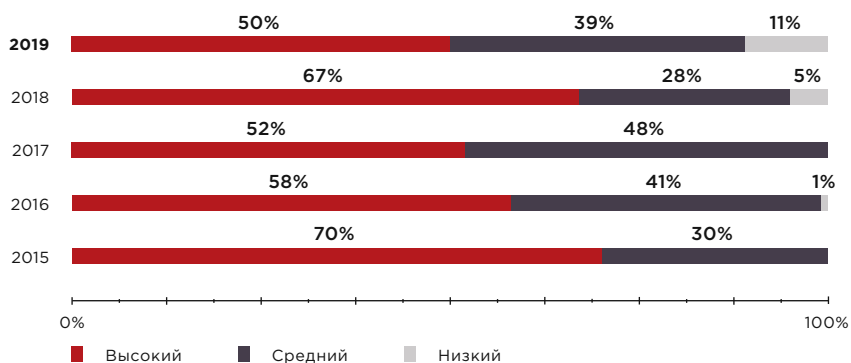


Рисунок 1. Доли уязвимых сайтов в зависимости от максимальной степени риска уязвимостей

Анализируя данные за последние пять лет, мы видим закономерное снижение доли сайтов, содержащих критически опасные веб-уязвимостей, и, соответственно, общее повышение уровня защищенности.

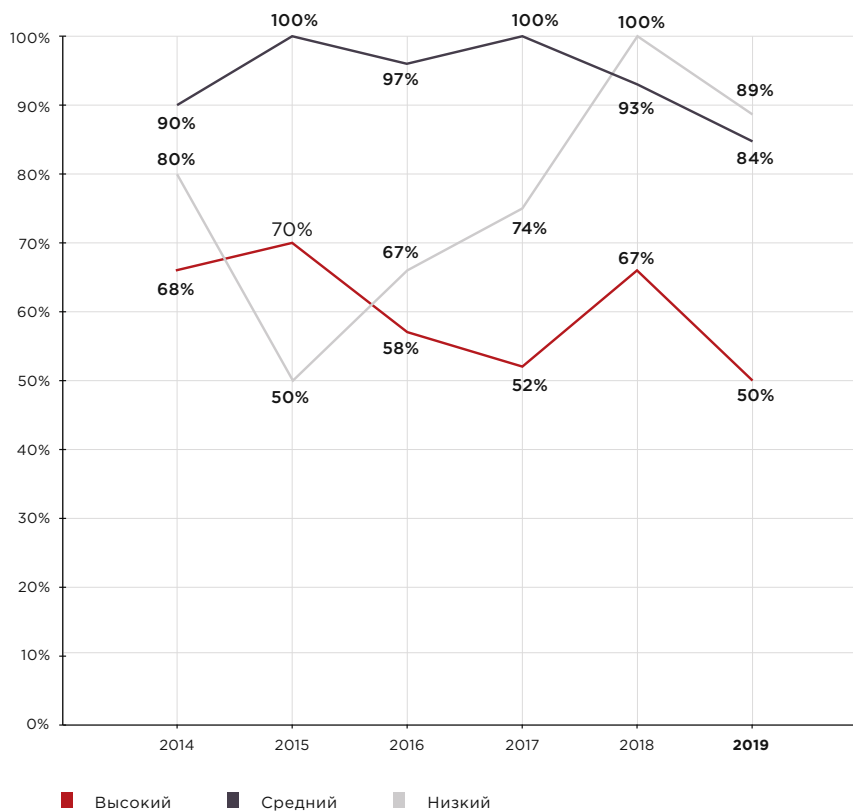


Рисунок 2. Доли сайтов с уязвимостями различной степени риска

Анализ защищенности веб-приложений

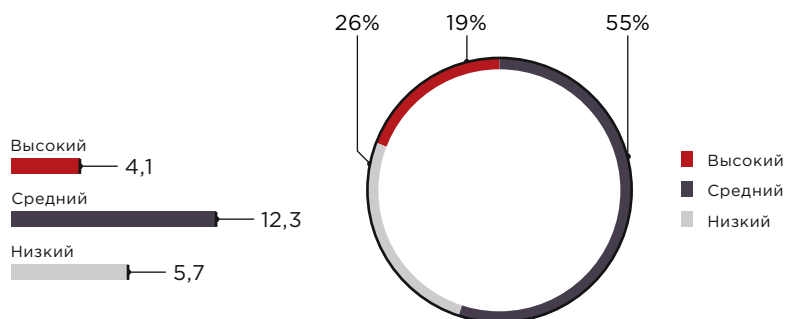


Рисунок 3. Среднее число уязвимостей на одно веб-приложение

Рисунок 4. Доля уязвимостей различной степени риска

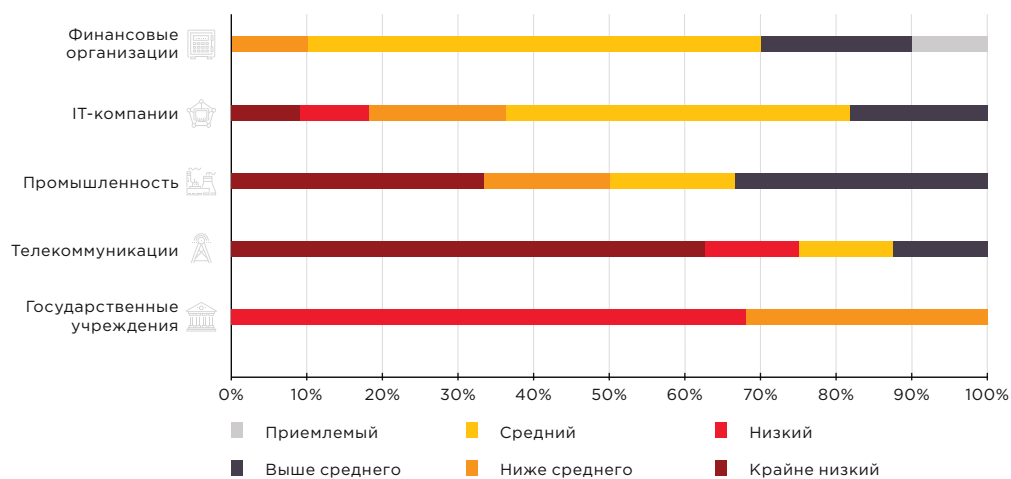


Рисунок 5. Доли приложений различного уровня защищенности по отраслям



Уровень защищенности веб-приложения определяется экспертами по результатам проведенных проверок и зависит от потенциально возможного воздействия на анализируемую систему с учетом специфики обрабатываемой в ней информации.

Самые распространенные уязвимости

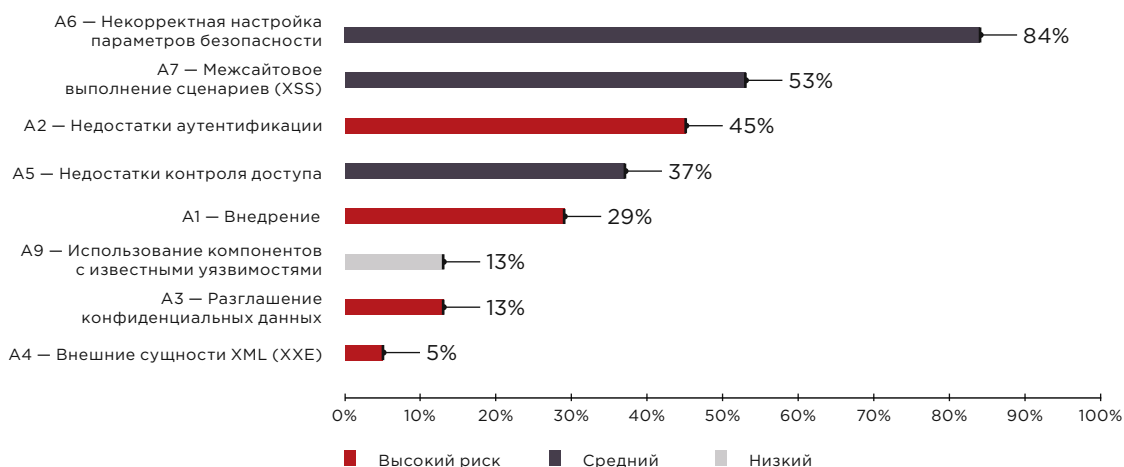


Рисунок 6. Наиболее распространенные уязвимости из списка OWASP Top 10 (доля приложений)

Чаще всех других в 2019 году в веб-приложениях встречались уязвимости, связанные с некорректными параметрами безопасности (Security Misconfiguration). Так, в каждом пятом проанализированном приложении были выявлены уязвимости, позволяющие проводить атаку на сессию, в частности отсутствие флагов HttpOnly и Secure у конфиденциальных Cookie-параметров. С помощью данных недостатков злоумышленник может, например, провести атаку типа «Межсайтовое выполнение сценариев» (Cross-Site Scripting, XSS), чтобы перехватить идентификатор сессии пользователя и от его имени выполнять различные действия в приложении.

В 45% веб-приложений были обнаружены недостатки аутентификации (Broken Authentication). Почти треть выявленных уязвимостей из этой категории — это некорректное ограничение количества неудачных попыток аутентификации. В результате эксплуатации этой уязвимости злоумышленник может подобрать учетные данные пользователя и таким образом получить доступ к веб-приложению. Так, например, для одного приложения потребовалось всего 100 попыток, чтобы успешно войти с правами администратора.



Большинство атак на аутентификацию связано с использованием исключительно паролей. Ранее считавшиеся хорошими требования к смене пароля и его сложности способствуют использованию ненадежных паролей пользователями.

Согласно последним рекомендациям NIST, организациям следует применять многофакторную аутентификацию.

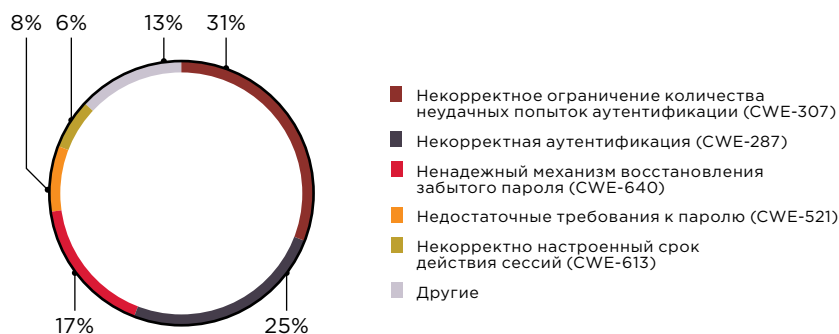


Рисунок 7. Уязвимости, связанные с недостатками аутентификации (Broken Authentication)

Недостатки контроля доступа (Broken Access Control) в 2019 году встречались в каждом третьем приложении. Обход ограничений доступа обычно приводит к несанкционированному разглашению, изменению или уничтожению данных. Так, например, в одном из проектов небезопасная авторизация позволяла изменять содержимое профиля любого пользователя. Специалисты Positive Technologies узнали логин администратора приложения, в его профиле изменили адрес электронной почты на собственный, а затем через стандартную процедуру восстановления пароля получили доступ к сайту с правами администратора.

Количество уязвимостей, связанных с аутентификацией и авторизацией, как правило, можно минимизировать, если при разработке веб-приложения придерживаться практик безопасного программирования SSDLC.

Помимо уязвимостей из списка Top 10–2017, сообщество OWASP выделяет ряд недостатков, наличие которых рекомендуется проверять¹. Треть веб-приложений оказались уязвимы для атаки типа Clickjacking (содержали уязвимость «Некорректное представление важной информации интерфейсом пользователя», CWE-451) и столько же для атаки «Подделка межсайтового запроса» (Cross-Site Request Forgery, CSRF). В ходе CSRF-атаки злоумышленник с помощью специально сформированных сценариев может выполнять действия от лица пользователя, авторизованного в уязвимом веб-приложении. Представим ситуацию: вы авторизованы на сайте, уязвимом для подделки межсайтового запроса (пусть это будет некий онлайн-банк). В это время вам приходит



В ходе атаки типа Clickjacking пользователь, как правило, уже находится на сайте злоумышленника, на котором так и манит нажать на кнопку (например, обещают большие скидки или рецепт вечной молодости). А поверх этой кнопки хакер реализует прозрачный HTML-фрейм (iframe) уязвимого сайта, и когда пользователь кликает по кнопке, происходит действие на уязвимом сайте, например ставится лайк под чьей-то фотографией. Таким образом происходит накрутка счетчика лайков, голосов и т. п. Одним из способов защиты от этого типа атак является использование HTTP-заголовка X-Frame-Options.

1. owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf

фишинговое письмо со ссылкой, и вы по ней переходите. А дальше от вашего имени на уязвимый сайт (в ваш онлайн-банк) отправляется специально сформированный хакером запрос, который выполняет нужные злоумышленнику действия (например, переводит деньги на его счет). Онлайн-банк не сможет отличить этот несанкционированный запрос от легитимных, если не используется защита от CSRF-атак. Защита от данного типа атак чаще всего основывается на использовании уникальных одноразовых ключей (CSRF-токенов), подтверждении подлинности (например, с помощью пароля) или доказательстве, что запрос выполняется реальным пользователем (например, с помощью CAPTCHA), на установке дополнительного флага SameSite для Cookie-параметров.

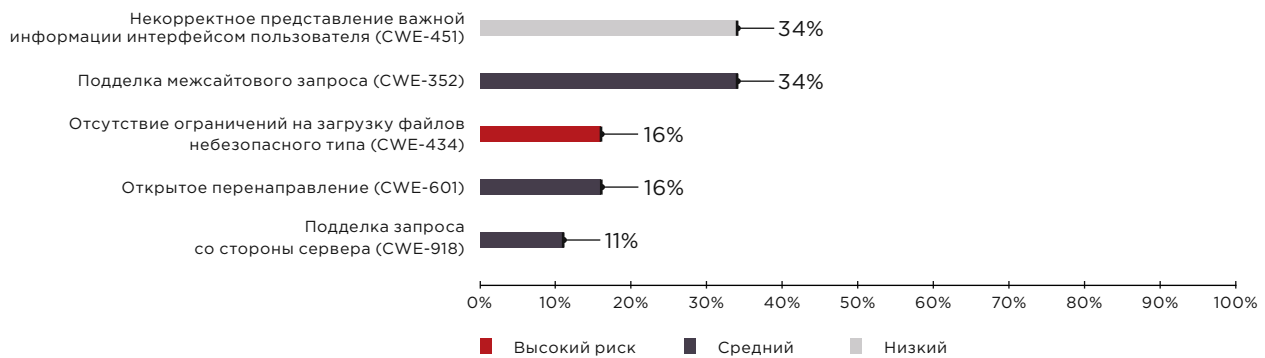


Рисунок 8. Распространенные уязвимости, не вошедшие в OWASP Top 10 (доля приложений)

Анализ угроз

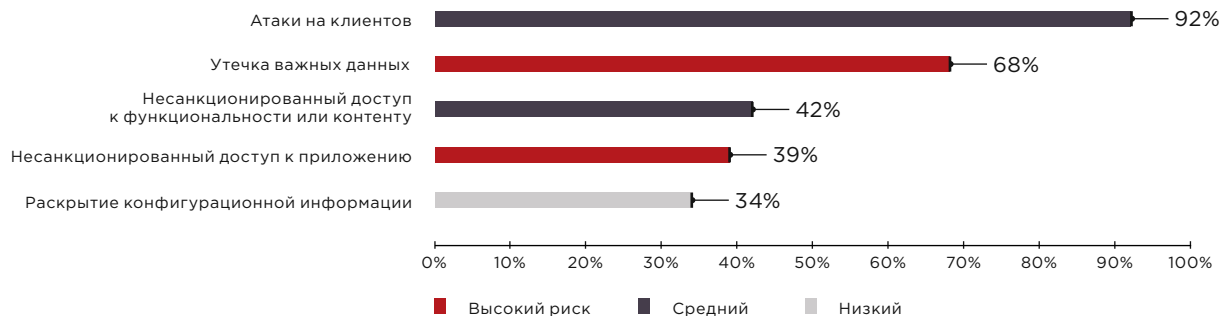


Рисунок 9. Топ-5 наиболее распространенных угроз (доля веб-приложений)

Весной 2019 года владельцы сайтов на WordPress стали жертвами массовых атак из-за XSS в плагине Yuzo Related Posts.

Как и в 2018 году, в 2019-м для 9 из 10 веб-приложений актуальна угроза атак на клиентов. Как и прежде, существенную роль при этом играет «Межсайтовое выполнение сценариев» (Cross-Site Scripting, XSS). В результате эксплуатации уязвимостей злоумышленник может заражать компьютеры пользователей вредоносным ПО, проводить фишинговые атаки, например для получения учетных данных, а также выполнять действия от имени пользователя. В качестве общих рекомендаций по защите отметим, что все данные, которые поступают со стороны пользователя и затем отображаются в браузере, включая заголовки HTTP-запроса, такие как User-Agent, Referer, — должны проходить предварительную обработку. Потенциально небезопасные символы, которые могут быть использованы при форматировании HTML-страницы, должны быть заменены на их эквиваленты, не являющиеся символами форматирования. Кроме того, мы рекомендуем использовать современные межсетевые экраны уровня приложения (web application firewalls), поскольку они умеют блокировать межсайтовое выполнение сценариев.



Рисунок 10. Уязвимости, позволяющие проводить атаки на клиентов

Угроза раскрытия важной информации возникает, как правило, вследствие недостаточной авторизации и недостаточной аутентификации в веб-приложении.

Утечка важной информации — это вторая наиболее актуальная угроза безопасности сайтов. Так, почти в половине утечек (в 47%) под угрозу попали персональные данные, а в 31% — учетные данные пользователей. Как показывает наш анализ киберинцидентов в 2019 году², именно кража информации является приоритетной целью злоумышленников в атаках на юридических лиц.

2. ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2019-q3/

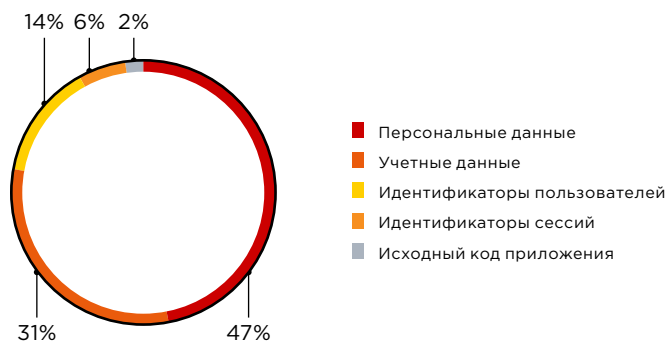


Рисунок 11. Разглашенные чувствительные данные

Самые опасные угрозы

В 16%

веб-приложений можно получить полный контроль

В 8%

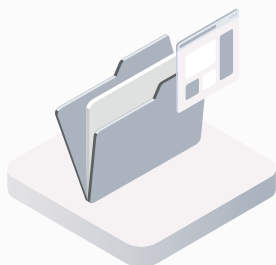
веб-приложений возможны атаки на ресурсы ЛВС

Результаты нашего исследования свидетельствуют о том, что на сегодняшний день не все компании готовы обеспечить надежную защиту персональных данных.

В 16% веб-приложений были найдены критически опасные уязвимости, позволяющие получить контроль не только над приложением, но и над ОС сервера.

Злоумышленник, получивший контроль над веб-приложением может, к примеру, внедрить в его код JavaScript-сниффер и продолжить атаку уже на пользователей сайта. Снифферы могут использоваться для кражи как учетных и персональных данных, так и данных банковских карт. В 2018—2019 годах среди атак на частных лиц наиболее опасными оказались именно атаки с использованием JavaScript-снифферов. Поскольку снифферы внедряют в код, для того, чтобы их обнаружить, нужно проводить анализ защищенности методом белого ящика.

В случае целенаправленной атаки на организацию уязвимости веб-приложения могут помочь злоумышленникам получить данные о внутренней сети компании — о структуре сегментов сети, используемых портах, сервисах и т. п. В ряде случаев нарушители даже могут получить доступ к внутренним ресурсам и хранящейся там конфиденциальной информации.



Злоумышленники могут собирать украденные учетные данные в специальные базы и затем использовать их для атак на другие веб-ресурсы (атаки типа credential stuffing). От подобных атак в мае 2019 года пострадали полмиллиона клиентов двух интернет-магазинов.

Сравнение тестовых и продуктивных систем

В 2019 году доля продуктивных систем, содержащих уязвимости высокого уровня риска, снизилась до 45% (71% в 2018 году), но осталась выше результатов 2017 года (было 25%). В тестовых системах ситуация как в прошлом году: доля приложений, содержащих уязвимости высокого уровня риска, составила 56%.

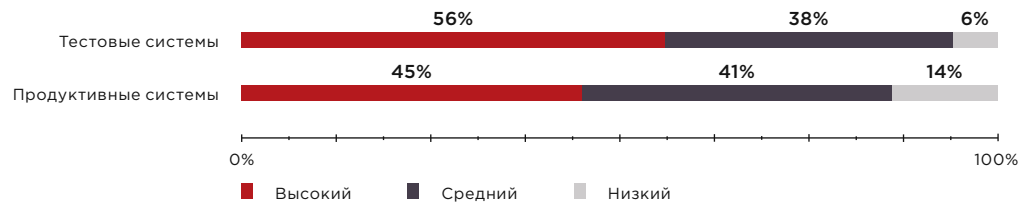


Рисунок 12. Доли систем по максимальному уровню риска уязвимостей

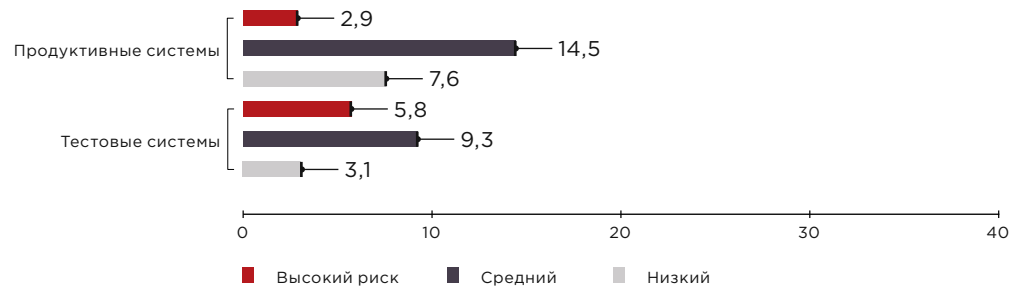
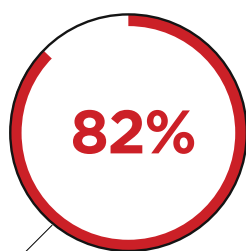


Рисунок 13. Среднее число уязвимостей на одну систему

В 2019 году продуктивные системы составили ровно половину всех проанализированных (вместо 79% в 2018-м). Стоит отметить, что при анализе защищенности продуктивных систем компании стараются исключить риски, связанные с нарушением работы веб-приложения, и отказываются от проведения некоторых проверок. Это приводит к тому, что исследователям не удастся продемонстрировать возможность эксплуатации некоторых потенциальных уязвимостей. В такой ситуации их демонстрация может проводиться на тестовом стенде.

Мы не раз отмечали важность внедрения процесса безопасной разработки в жизненный цикл веб-приложения, поскольку анализ защищенности, проведенный для системы, которая еще только разрабатывается, позволяет более тщательно подойти к исправлению выявленных недостатков. Стараясь закрыть уязвимости в уже функционирующем веб-приложении, разработчики могут в спешке использовать не самые надежные методы защиты, или, проще говоря, «костыли».

Анализ защищенности методом белого ящика



Уязвимости кода веб-приложения

Наш опыт показывает, что большинство уязвимостей сайтов связаны с ошибками в коде веб-приложения. И это главный повод предоставить экспертам исходный код для проведения анализа защищенности — либо самостоятельно использовать анализатор кода в рамках процесса безопасной разработки.

Анализ защищенности методом белого ящика выполняется несколькими специалистами одновременно, для того чтобы не упустить ни одной детали и выявить наибольшее количество недостатков. Кроме того, данный вид работ включает как ручной анализ кода, так и анализ с использованием автоматизированных средств. Автоматизированный поиск уязвимостей ускоряет процесс тестирования, но требует ручной проверки для исключения ложных срабатываний, а ручной анализ кода занимает больше времени, но гарантирует, что выявленные уязвимости актуальны.

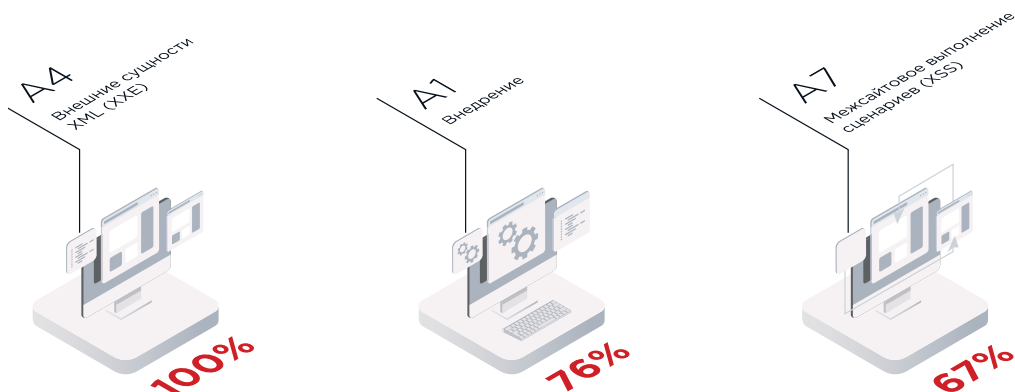


Рисунок 14. Доля уязвимостей из списка OWASP Top 10, выявленных методом белого ящика



В рамках работ по анализу защищенности одного веб-приложения эксперты Positive Technologies смогли прочитать исходный код одного из скриптов и обнаружили фрагмент, который позволял удаленно выполнять произвольный код. Используя эту уязвимость, любой внешний злоумышленник мог получить контроль над сервером, читать чувствительную информацию, редактировать и удалять данные на страницах приложения, а также полностью вывести сайт из строя. Примечательно, что рядом с этим фрагментом стоял комментарий разработчика: «Что это?». Вероятно, этот артефакт был забыт другим разработчиком при отладке приложения, а позже его коллеги не посчитали вопрос важным и не стали вникать в смысл и назначение этих строк кода.

Заключение

Подводя итоги, отметим, что уровень защищенности большинства веб-приложений продолжает оставаться низким. В каждом втором сайте присутствуют уязвимости высокого уровня риска. Впрочем, мы видим, что с каждым годом постепенно снижается доля веб-приложений, содержащих критически опасные уязвимости. Число уязвимостей, которое в среднем приходится на одно приложение, снизилось по сравнению с 2018 годом в полтора раза. Положительная тенденция заключается еще и в том, что компании начинают серьезней относиться к защите веб-приложений, причем не только публичных, но и используемых для внутренних нужд. Мы надеемся, что в следующем году уровень защищенности сайтов продолжит расти, а количество успешных атак на веб-приложения пойдет на спад.

Достижение и последующее поддержание высокого уровня защищенности веб-приложения — это непростой процесс. На наш взгляд, наиболее эффективно его можно выстроить, придерживаясь двух главных правил:

- **исправлять выявленные уязвимости как можно раньше;**
- **автоматизировать процессы, где это возможно.**

Для их выполнения, помимо проведения анализа защищенности веб-приложений, компаниям стоит уделить внимание обучению разработчиков методам безопасной разработки и использовать инструменты для автоматизированного анализа исходного кода. Это позволит сократить количество ошибок и уязвимостей еще на этапе разработки. Кроме того, для защиты от атак на веб-приложения мы всегда рекомендуем применять превентивные меры защиты, такие как межсетевой экран уровня приложений (web application firewall, WAF). Веб-приложения постоянно модернизируются, а вместе с новыми «фичами» в них могут появляться и новые уязвимости. Использование WAF позволит снизить соответствующие риски. При этом WAF должен не только обнаруживать и предотвращать известные атаки на уровне приложения и бизнес-логики, но и выявлять эксплуатацию уязвимостей нулевого дня, предотвращать атаки на пользователей, анализировать и сопоставлять множество событий для выявления цепочек атак.

Портрет участников

38

веб-приложений
проанализированы
в 2019 году

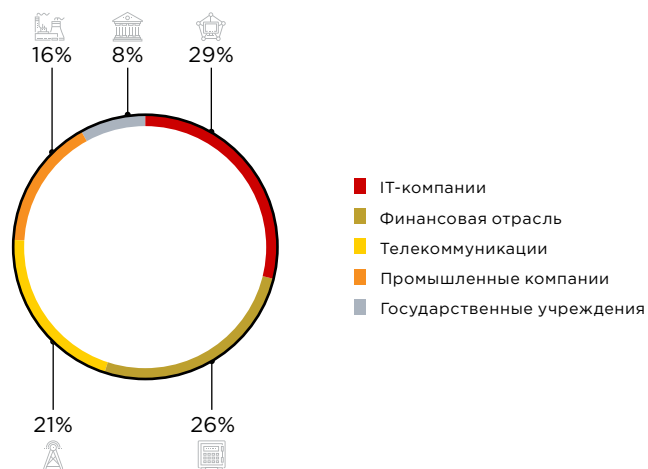


Рисунок 15. Портрет участников исследования

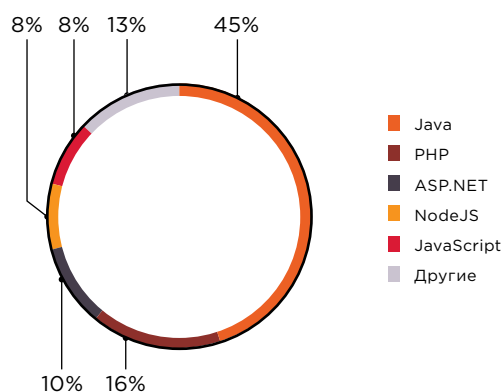


Рисунок 16. Средства разработки (доля приложений)

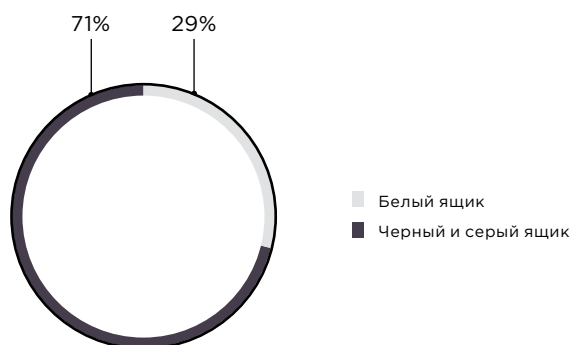


Рисунок 17. Методы тестирования (доля приложений)

Доля продуктивных систем

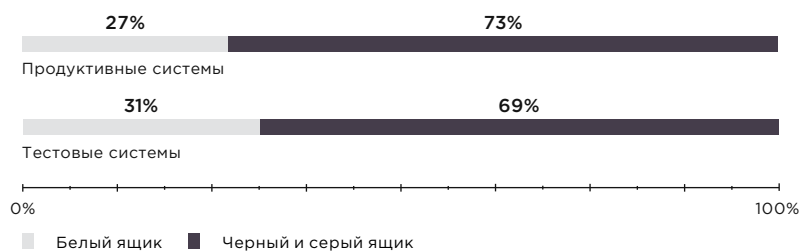
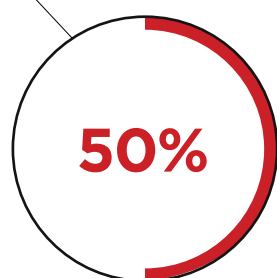


Рисунок 18. Методы тестирования продуктивных и тестовых систем

Методика

Отчет содержит результаты исследования 38 полнофункциональных веб-приложений, для которых в 2019 году проводился углубленный анализ с наиболее полным покрытием проверок. Результаты проектов по тестированию на проникновение, инструментальному сканированию и исследованию систем ДБО не вошли в статистику: эта информация представлена в других наших отчетах. Кроме того, в выборке не представлены системы, владельцы которых не дали своего согласия на использование результатов анализа защищенности в исследовательских целях.

Оценка защищенности проводилась вручную методами черного, серого и белого ящика с использованием вспомогательных автоматизированных средств. Метод черного ящика заключается в проведении работ по оценке защищенности информационной системы со стороны внешнего атакующего без предварительного получения какой-либо дополнительной информации о ней от владельца. Метод серого ящика аналогичен, но в качестве нарушителя рассматривается пользователь, имеющий определенные привилегии в системе. При анализе методом белого ящика для оценки защищенности информационной системы используются все имеющиеся данные о ней, включая исходный код приложений.

Обнаруженные уязвимости классифицированы по системе Common Weakness Enumeration (CWE). Для удобства, поскольку эта классификация уязвимостей очень подробная, мы выделили из их числа те, которые входят в рейтинг [OWASP Top 10–2017](#), — и проанализировали, как часто они встречались в исследованных нами веб-приложениях.

В настоящем документе приведены только уязвимости, связанные с ошибками в коде и конфигурации веб-приложений. Другие распространенные проблемы информационной безопасности (к примеру, недостатки процесса управления обновлениями ПО) не рассматриваются. В статистике также не учтены уязвимости из категории A10 — Insufficient Logging & Monitoring рейтинга OWASP Top 10–2017, так как в рамках работ по анализу защищенности веб-приложений мы не оценивали достаточность журналирования и мониторинга. Степень риска уязвимостей оценивалась согласно системе Common Vulnerability Scoring System (CVSS) версии 3.1; на основе этой оценки выделялись качественные оценки высокого, среднего и низкого уровня риска.

О компании

ptsecurity.com
pt@ptsecurity.com
facebook.com/PositiveTechnologies
facebook.com/PHDays

Positive Technologies — один из лидеров европейского рынка систем анализа защищенности и соответствия стандартам, а также защиты веб-приложений. Организации во многих странах мира используют решения Positive Technologies для оценки уровня безопасности своих сетей и приложений, для выполнения требований регулирующих организаций и блокирования атак в режиме реального времени. Благодаря многолетним исследованиям специалисты Positive Technologies заслужили репутацию экспертов международного уровня в вопросах защиты SCADA- и ERP-систем, крупнейших банков и телеком-операторов.

Деятельность компании лицензирована Минобороны России, ФСБ России и ФСТЭК России, продукция сертифицирована Минобороны России и ФСТЭК России.