Белорусско-Российский университет
Кафедра «Программное обеспечение информационных технологий»

Защита информации

Защита internet ресурсов, сайтов [OWASP]

КУТУЗОВ Виктор Владимирович

Республика Беларусь, Могилев, 2024

Защита от киберпреступников

Введение

- **Атаки на веб-приложения — один из наиболее популярных методов кибератак.**
- По данным исследования Positive Technologies в 2021 году, 17% от общего числа атак пришлось на эксплуатацию уязвимостей и недостатков защиты веб-приложений.
- Злоумышленники могут использовать скомпрометированные сайты в различных целях: для распространения вредоносного ПО, кражи конфиденциальных данных, несанкционированного внедрения информации, для мошенничества или проникновения во внутреннюю инфраструктуру компании. Все вышеперечисленное — прямая угроза для функционирования и репутации организаций, поэтому веб-приложения нужно защищать и не оставлять слабых мест при их разработке.

Про угрозы:

- **В абсолютном большинстве веб-приложений (98%) злоумышленники имеют возможность проводить атаки на пользователей.** Подобные атаки могут привести к распространению вредоносного ПО, перенаправлению на сайты злоумышленников или краже данных с использованием методов социальной инженерии.
- **Утечки важных данных имели место в 91% веб-приложений.** Чаще всего раскрывались идентификаторы пользователей (в 84% случаев). Две трети приложений оказались подвержены раскрытию персональных данных, а около половины — утечкам учетных данных пользователей.
- **Возможность несанкционированного доступа была отмечена в 84% веб-приложений.** При этом полный контроль над сайтом удалось получить в 5% случаев.

Positive Technologies. Уязвимости и угрозы веб-приложений в 2020–2021 гг.
https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020-2021/

Про уязвимости:

- **Наиболее опасными уязвимостями веб-приложений** стали недостатки механизмов авторизации и аутентификации пользователей. Эти уязвимости позволяют получить несанкционированный доступ к конфиденциальной информации и функциям приложения.
- **Недостатки авторизации также были связаны с уязвимостями протокола oAuth.** Уязвимости реализации авторизации с помощью oAuth могут быть использованы злоумышленниками для перехвата сессионных и учетных данных пользователей, а после привести к несанкционированному доступу к приложению.

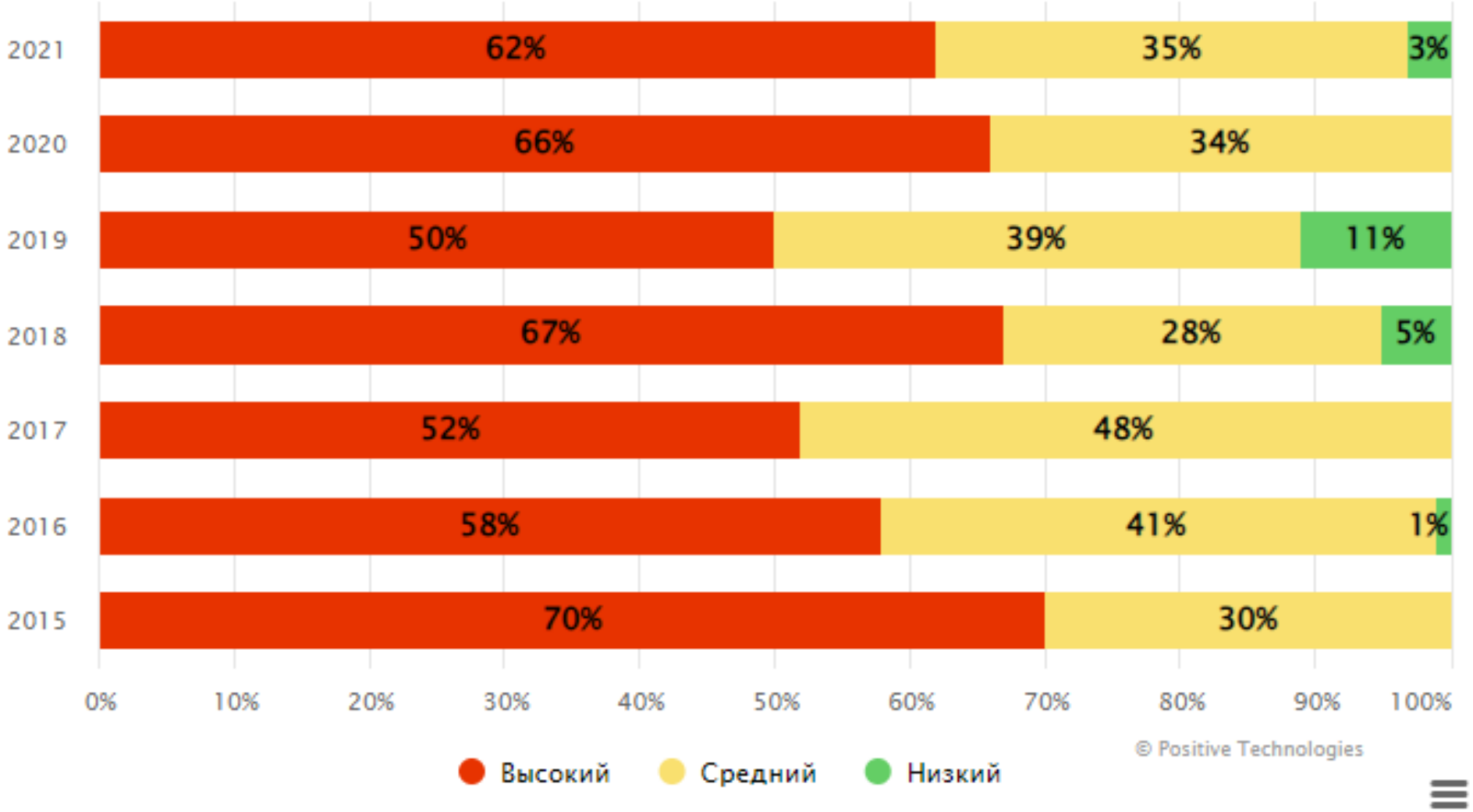
Positive Technologies. Уязвимости и угрозы веб-приложений в 2020–2021 гг.
https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020-2021/

Про уязвимости:

- **Среднее количество уязвимостей на одно веб-приложение сократилось более чем на треть по сравнению с 2019 годом.** В среднем на один сайт приходится 15 уязвимостей, две из которых — высокой степени риска.
- **15% — доля уязвимостей высокой степени риска от общего числа выявленных уязвимостей.** Две трети сайтов содержат такие уязвимости.
- **72% уязвимостей были связаны с ошибками в коде веб-приложений.**

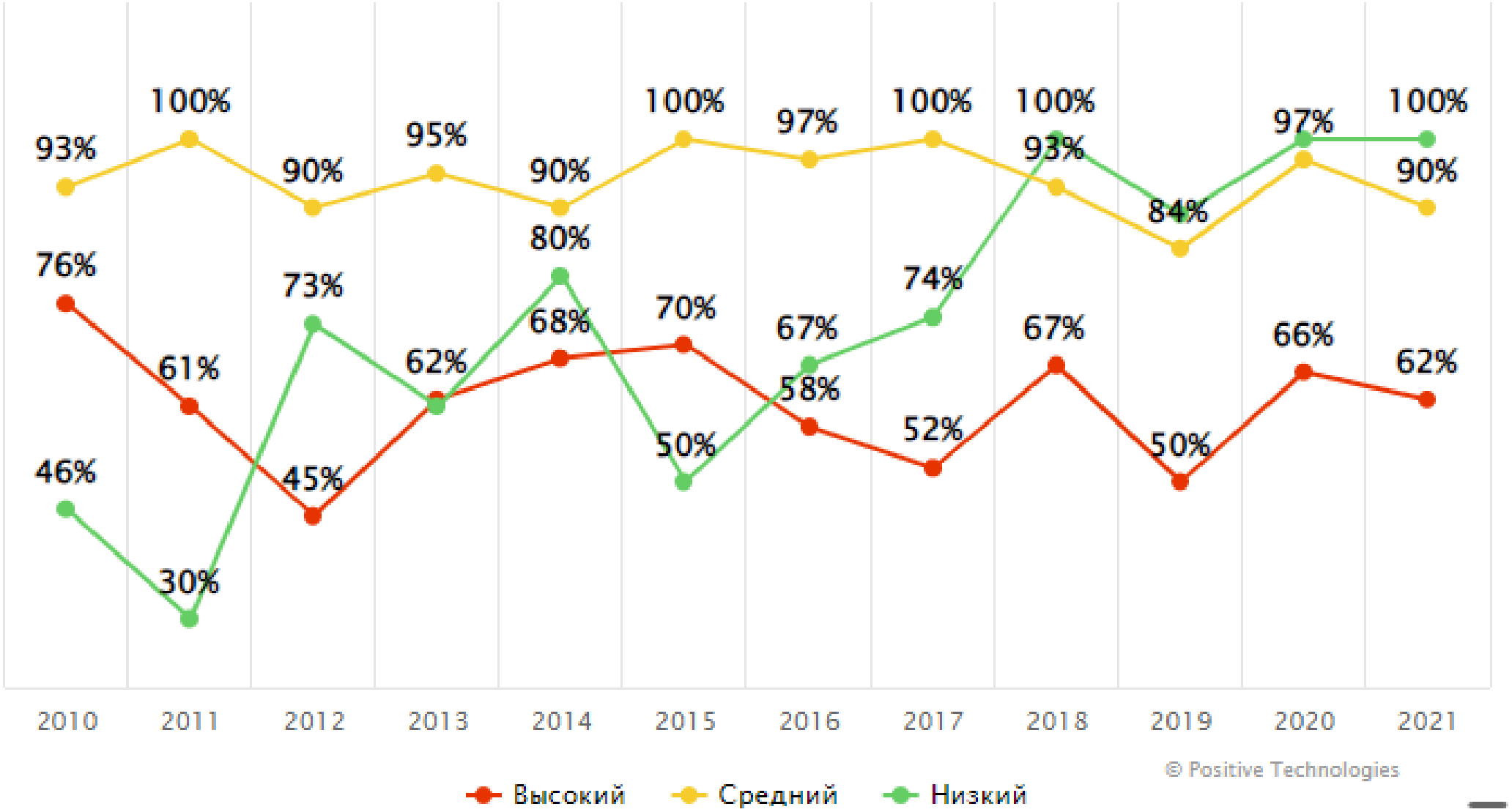
Positive Technologies. Уязвимости и угрозы веб-приложений в 2020–2021 гг.
https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020-2021/

Уязвимости веб-приложений



Доли уязвимых веб-приложений в зависимости
от максимальной степени риска уязвимостей

Уязвимости веб-приложений

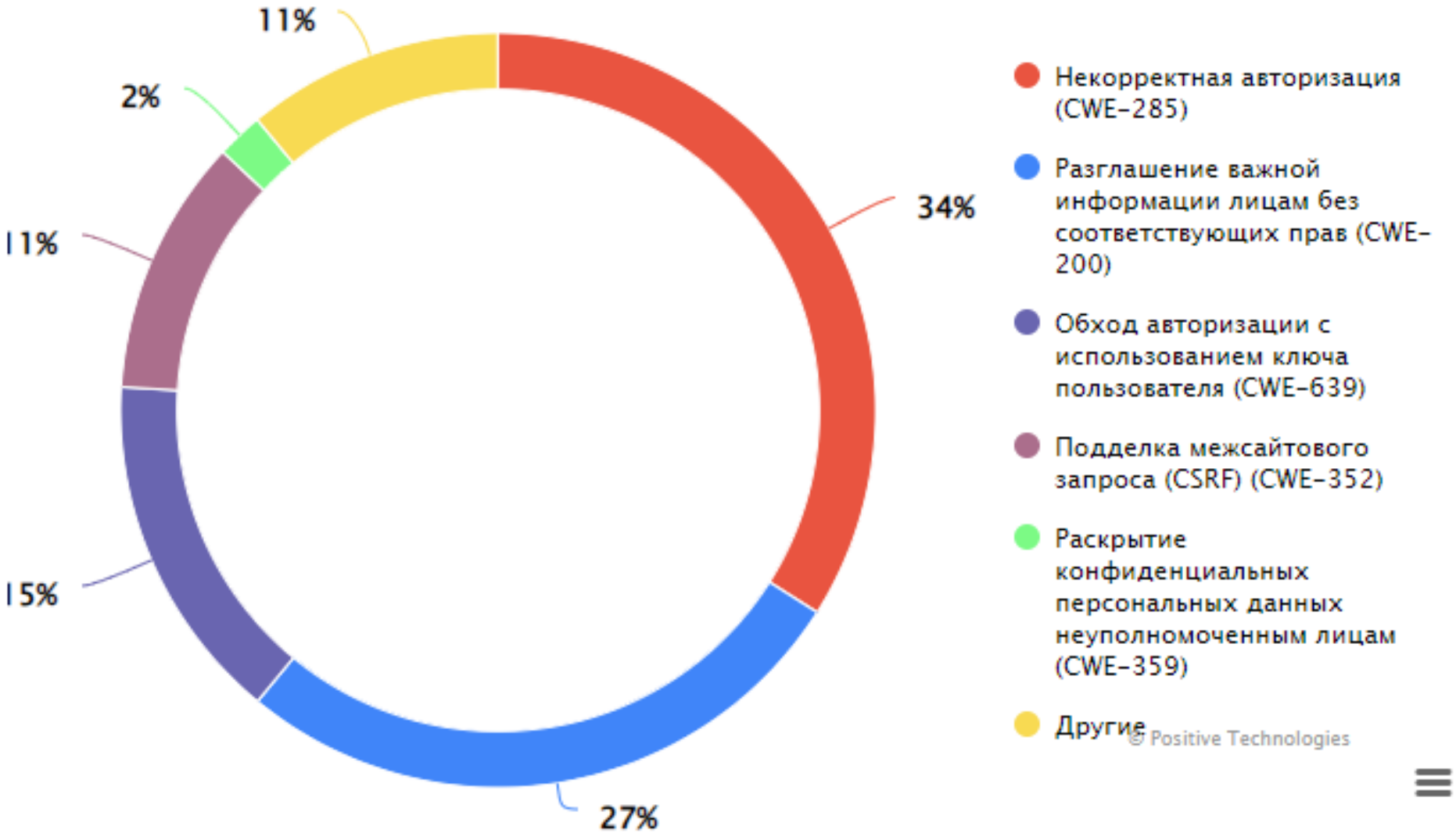


Доли веб-приложений с уязвимостями различной степени риска

Уязвимости веб-приложений

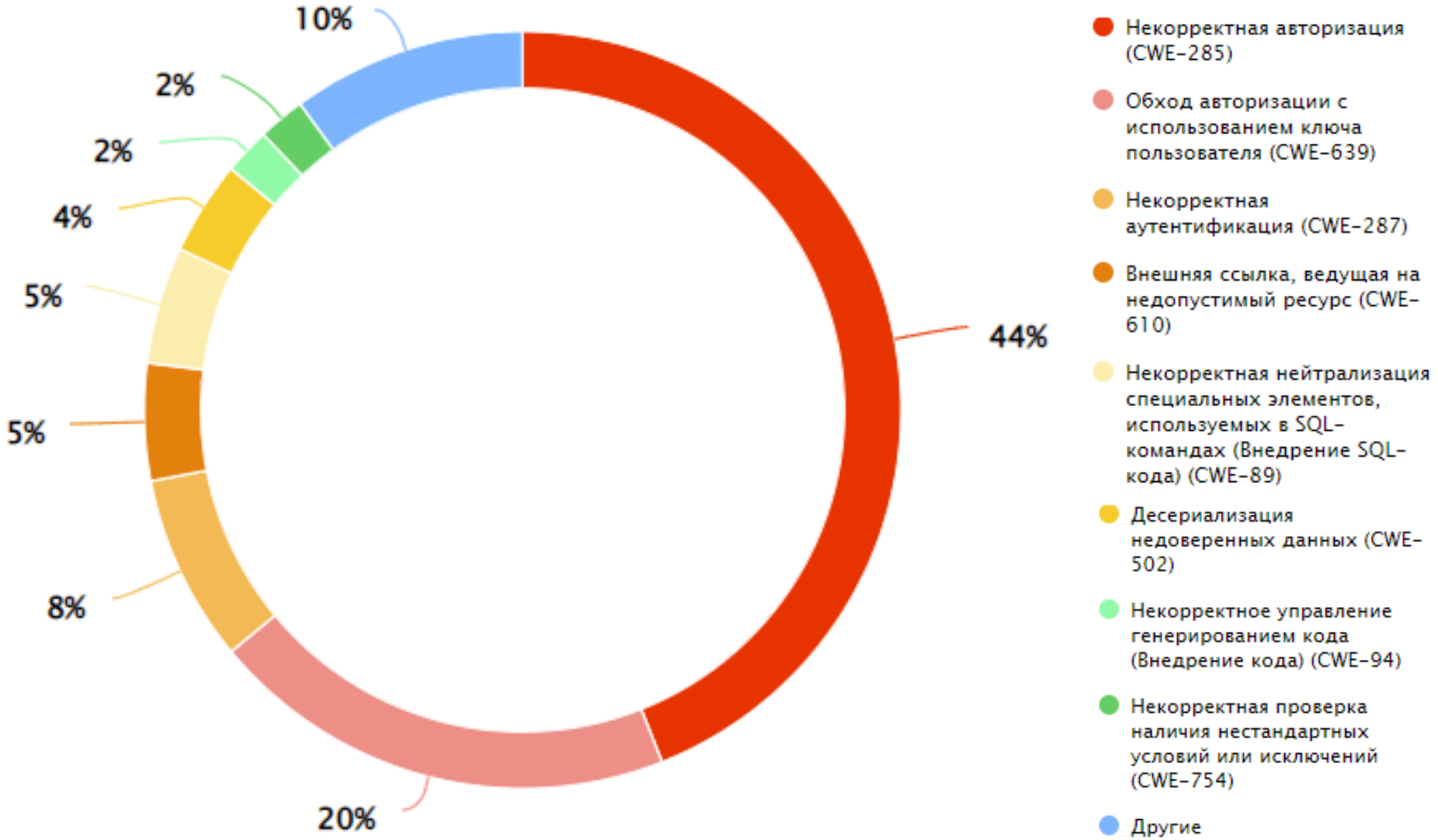
Распределение уязвимостей по категориям OWASP Top 10 — 2021 (доля приложений)

Недостатки механизма авторизации



Уязвимости, связанные с недостатками контроля доступа (Broken Access Control)

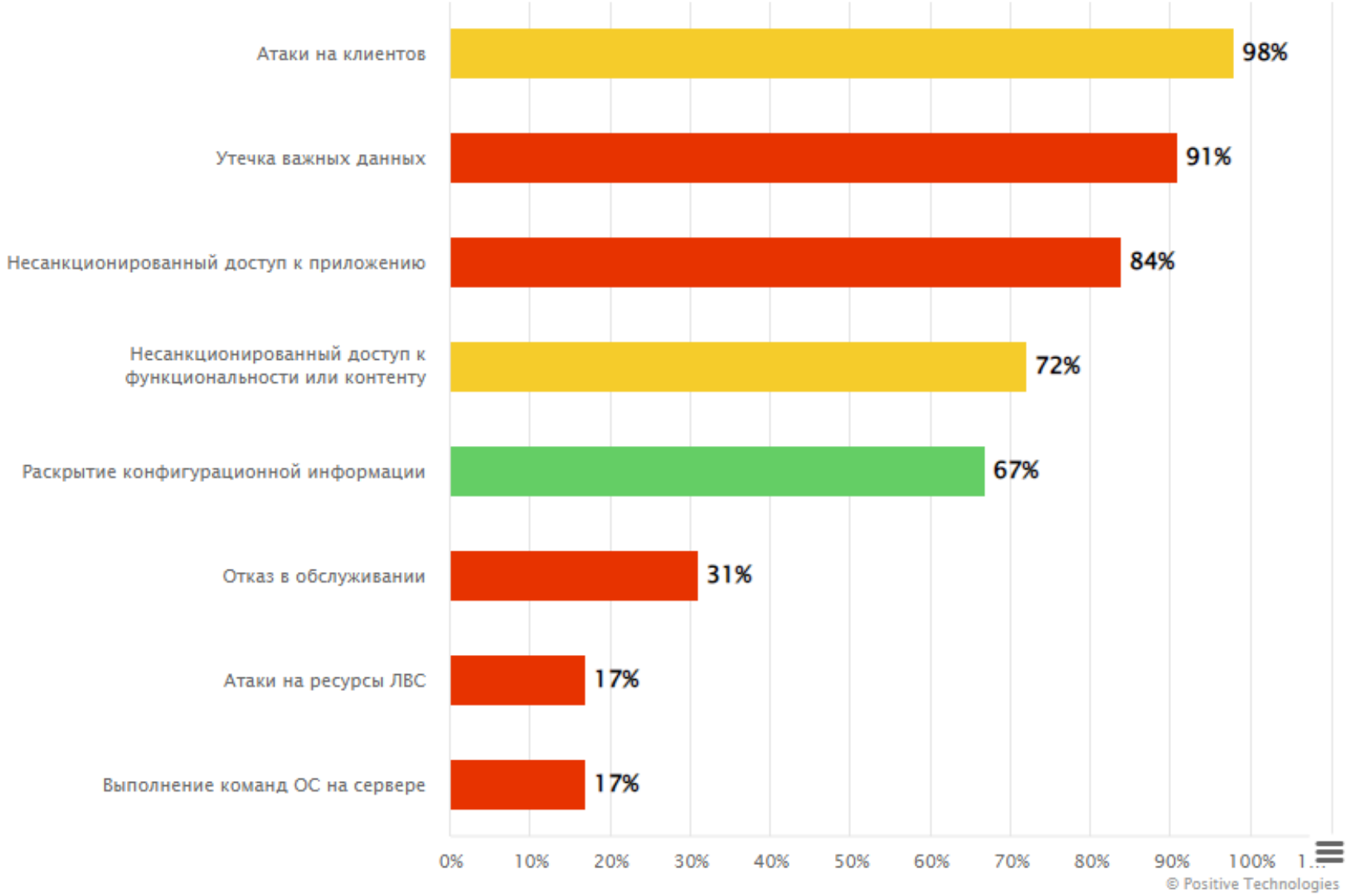
Недостатки механизма авторизации



Распределение уязвимостей высокого уровня риска

Угрозы веб-приложений

Распространенные угрозы веб-приложений (доля веб-приложений)



OWASP (Open Web Application Security Project)

https://owasp.org/

OWASP (Open Web Application Security Project) https://owasp.org/

- **Open Web Application Security Project (OWASP)** – открытый проект обеспечения безопасности web-приложений.

- Сообщество OWASP включает в себя корпорации, образовательные организации и частных лиц со всего мира. OWASP работает над созданием общедоступных статей, учебных пособий, документации, инструментов и технологий.

- Проект OWASP ссылается множество стандартов, инструментов и организаций, включая MITRE, PCI DSS, DISA, FTC и множество других.

OWASP (Open Web Application Security Project) https://owasp.org/

- Наиболее востребованные документы, опубликованные OWASP, включают в себя:
  1. Руководство OWASP,
  2. Обзорное Руководство по Коду OWASP
  3. Проект Топ-10 OWASP.

- **OWASP Топ-10** — это список из десяти самых распространённых на данный момент уязвимостей веб-приложений.

OWASP (Open Web Application Security Project)

OWASP PROJECTS CHAPTERS EVENTS ABOUT Search OWASP.org Store Donate Join

Who is the OWASP® Foundation?

The Open Web Application Security Project® (OWASP) is a nonprofit foundation that works to improve the security of software. Through community-led open-source software projects, hundreds of local chapters worldwide, tens of thousands of members, and leading educational and training conferences, the OWASP Foundation is the source for developers and technologists to secure the web.

- Tools and Resources
- Community and Networking
- Education & Training

For nearly two decades corporations, foundations, developers, and volunteers have supported the OWASP Foundation and its work. Donate, Join, or become a Corporate Member today.

Project Spotlight: OWASP Top 10

We are back again with yet another OWASP Spotlight series and this time we have a project which needs no introduction and I got the chance to interact with Andrew van der Stock, OWASP Foundation Executive Director and the project leader for OWASP Top 10.

Featured Event: OWASP's 20th Anniversary Event Celebration

Join us September 24 for 24-hours as we honor the past, celebrate the present and embrace the future of OWASP and cybersecurity. Hear from world renowned keynotes and

Некоторые из проектов OWASP

- **Стандарт Подтверждения Безопасности Приложений OWASP** (OWASP Application Security Verification Standard (ASVS)) — Стандарт для проведения проверок уровня безопасности приложений. https://owasp.org/www-project-application-security-verification-standard/
- **Руководство по Разработке OWASP** дает практические советы и содержит примеры кода на J2EE, ASP.NET и PHP. Серьёзно переработанное в 2014 году, Руководство по Разработке охватывает обширный массив вопросов безопасности для уровня приложений, от SQL инъекций до современных проблем, таких как фишинг, обработка кредитных карт, фиксация сессий, подделка межсайтовых запросов, согласование и конфиденциальность. https://wiki.owasp.org/index.php/OWASP_Guide_Project
- **Руководство по Тестированию OWASP** включает «лучшую практическую» основу для тестирования проникновений, которую пользователи могут использовать в своих организациях и «низкоуровневое» руководство по тестированию проникновений, которое описывает техники тестирования наиболее распространенных проблем с безопасностью в веб-приложениях и веб-сервисах. https://wiki.owasp.org/index.php/OWASP_Testing_Project

Некоторые из проектов OWASP

- **Руководство по Обзору Кода OWASP версии 1.1** является вторым по продажам печатным изданием, выпущенными OWASP в 2008 году. При этом уже версия 1.0 собрала множество положительных отзывов и стала одним из ключевых продуктов, позволяющих OWASP бороться с проблемами в безопасности программного обеспечения. https://wiki.owasp.org/index.php/Category:OWASP_Code_Review_Project
- **OWASP ZAP Проект: Прокси Зет-Атаки** — это простой в применении встроенный инструмент тестирования проникновений, служащий для нахождения уязвимостей веб-приложений. Он разработан для использования людьми с различным опытом в сфере безопасности и является эталоном для разработчиков и тестировщиков функционала, которые не имеют опыта в тестировании проникновений. https://owasp.org/www-project-zap/
- **OWASP Топ-10: цель проекта Топ-10** — увеличение осведомленности о безопасности приложений при помощи определения наиболее критичных рисков, угрожающих организациям. На проект Топ-10 ссылается множество стандартов, инструментов и организаций, включая MITRE, PCI DSS, DISA, FTC, и множество других. Первая версия рейтинга OWASP Top 10 появилась в 2004 году, и с тех пор документ обновляется каждые три-четыре года. Обновленные версии публиковались в 2007, 2010, 2013 и 2017 годах. https://owasp.org/www-project-top-ten/

Некоторые из проектов OWASP

- **OWASP Модель Завершенности Программного Обеспечения:** этот проект стремится к созданию полезной основы для помощи организациям в формулировании и воплощении стратегии безопасности приложений, с учетом специфических бизнес-рисков, которые предстают перед организацией. https://wiki.owasp.org/index.php/OWASP_SAMM_Project
- **OWASP Mantra Security Framework**: Коллекция хакерских утилит, расширений и скриптов основанная на Mozilla Firefox. https://owasp.org/projects/
- Множество других инструментов и приложений для обеспечения безопасности OWASP. https://wiki.owasp.org/index.php/Category:OWASP_Project

OWASP TOP 10 2023
критичные риски безопасности API

OWASP TOP 10 2023
критичные риски безопасности API

- **API** — неотъемлемая составляющая современных ИТ-сервисов, без них не обходится практически ни одно мобильное или веб-приложение.

- Но поскольку через API проходят огромные объемы данных, а многие API передают конфиденциальные данные, они все чаще становятся объектом кибератак.

- Прогнозируется, что количество инцидентов с использованием API в качестве точек входа возрастет на 996% к 2030 году!

OWASP TOP 10 2023
критичные риски безопасности API

1. API1:2023 Некорректная авторизация на уровне объекта
2. API2:2023 Некорректная аутентификация пользователей
3. API3:2023 Нарушенная авторизация на уровне свойств объекта
4. API4:2023 Неограниченное потребление ресурсов
5. API5:2023 Нарушенная авторизация на уровне функции
6. API6:2023 Неограниченный доступ к конфиденциальным бизнес-потокам
7. API7:2023 Подделка запроса на стороне сервера
8. API8:2023 Ошибки настроек безопасности
9. API9:2023 Ненадлежащее управление активами
10. API10:2023 Небезопасное использование API

API1:2023 Некорректная авторизация на уровне объекта

- **Авторизация на уровне объекта** — это механизм, который используется для проверки права доступа пользователя к определенному объекту. При отсутствии проверки на уровне объекта злоумышленники могут использовать конечные точки API, работающие с ID объекта, и получить доступ к критическим данным.

- OWASP рекомендует применять надежный механизм авторизации на основе политики пользователей и иерархии в каждой функции, использующей клиентский ввод для доступа к базе данных. Также следует использовать случайные и непредсказуемые значения для ID записей.

API2:2023 **Некорректная аутентификация пользователей**

- Механизм проверки подлинности часто становится целью кибератак ввиду своей общедоступности. Некорректные механизмы аутентификации в числе прочего
  - Позволяют генерировать ненадежные пароли,
  - Не проверяют подлинность токенов,
  - Позволяют менять почтовый адрес, пароль и другие данные, не запрашивая пароль,
  - Принимают не подписанные JWT-токены или токены с ненадежной подписью,
  - Не блокируют учетную запись и не выводят капчу в случае атаки грубой силы на одну и ту же учетную запись,
  - Используют ненадежные ключи шифрования и не хешируют пароли.
- Применение ненадежных механизмов аутентификации может привести к тому, что злоумышленники получат полный контроль над учетной записью пользователя.
- Необходимо провести тщательное исследование, выявить все возможные риски всех способов проверки подлинности и учесть их при разработке механизмов аутентификации. Также рекомендуется по возможности использовать двухфакторную аутентификацию.

API3:2023 Нарушенная авторизация на уровне свойств объекта

- API (особенно REST API) часто предоставляют конечные точки, возвращающие данные обо всех свойствах объекта. Конечная точка API считается уязвимой, если она
- Предоставляет доступ ко всем свойствах объекта, даже конфиденциальным данным, которые не должны быть доступны для чтения,
- Позволяет пользователю изменять, добавлять или удалять значения свойств объекта, которые не должны были быть доступны.
- Для минимизации этого риска необходимо всегда проверять, есть ли у пользователя право на доступ к свойствам объекта, передаваемым через конкретную конечную точку API. Кроме того, следует уменьшить количество возвращаемых данных, насколько позволяет бизнес-логика.

API4:2023 **Неограниченное потребление ресурсов**

- Обработка запросов к API связана с потреблением ресурсов, таких как пропускная способность сети, CPU, память и хранилище данных. В некоторых случаях ресурсы, необходимые для обработки запроса (например, при обращении к базе данных), предоставляются по модели «оплата за запрос» (pay per request).

- Часто API не ограничивают клиентские взаимодействия и потребление ресурсов, что может привести к отказу в обслуживании (DoS) или увеличению операционных расходов.

- Следовательно, разработчикам необходимо установить ограничения на потребление ресурсов, например, установить лимит потребления памяти, размер передаваемых файлов, количество операций в одном клиентском API-запросе, количество или временной диапазон запросов от одного клиента/пользователя API и так далее.

API5:2023 Нарушенная авторизация на уровне функции

- Современные приложения могут включать сложную иерархию пользователей и большое количество ролей и групп, в результате чего разделение между административными и обычными функциями может быть нечетким. При эксплуатации таких уязвимостей злоумышленник может отправлять вызовы к конечной точке API, к которой у него не должно быть доступа, и выполнять критичные действия (создание, обновление, удаление) путем изменения HTTP метода (GET на PUT, например).

- Рекомендуется проанализировать конечные точки API на предмет изъянов авторизации на уровне функций. Административные контроллеры и административные функции внутри обычных контроллеров должны выполнять проверку авторизации на основании ролей/групп пользователей. Для доступа к каждой функции должна быть определена конкретная роль.

API6:2023 Неограниченный доступ к конфиденциальным бизнес-потокам

- Конечные точки API часто предоставляют доступ к бизнес-потокам, но при разработке конечных точек не всегда учитываются риски, связанные с чрезмерным доступом к тому или иному потоку. Например, воспользовавшись потоком «покупка продукта», злоумышленник может написать скрипт для автоматической покупки, купить весь запас товара, пользующегося высоким спросом, а затем продать его на другой платформе по более высокой цене.

- Эти уязвимости не всегда связаны с ошибками проектирования, и степень риска может варьироваться в зависимости от бизнеса. Соответственно, нужно сперва идентифицировать, какие потоки критичны для конкретной компании, а затем разработать механизмы их защиты, включающие, помимо прочего, обнаружение действий, похожих на автоматические, использование капчи или других методов подтверждения личности и т.д.

API7:2023 Подделка запроса на стороне сервера

- Подделка запроса на стороне сервера (SSRF) — это уявимость, при которой API не проверяет URL, предоставленный пользователем. В результате злоумышленники могут заставить серверное приложение отправить запрос на этот URL и получить информацию о внутренней сети (например, открытых портах) или другие критичные данные.

- Риск SSRF невозможно полностью устранить, но его можно уменьшить посредством проверки всех данных, отправленных пользователем, отключения переадресации HTTP и создания списка разрешений (принимаемые типы данные, URL-схемы и порты, удаленные источники данных).

API8:2023 **Ошибки настроек безопасности**

- Распространенные ошибки настроек безопасности API включают помимо прочего:
  - Недостаточно сильную защиту на любом уровне системы,
  - Незакрытые CVE,
  - Отсутствие TLS (или устаревшую версию сертификата),
  - Отсутствие или неправильную настройку политики разделения ресурсов между источниками,
  - Сообщения об ошибках, содержащие стек-трейсы или другую критичную информацию.
- Злоумышленники часто ищут незакрытые уязвимости, ненадежные настройки по умолчанию, незащищенные файлы и репозитории и другие ошибки конфигурации безопасности, чтобы использовать их для получения доступа к системе и конфиденциальным данным.
- Для надлежащей защиты API в течение всего жизненного цикла рекомендуется реализовать непрерывный процесс по усилению безопасности, включающий своевременное внедрение патчей, обзор и обновление настроек, автоматический процесс проверки эффективности настроек безопасности.
- Это касается всех компонентов и технологий в составе корпоративной ИТ-инфраструктуры.

API9:2023 **Ненадлежащее управление активами**

- Использование нескольких версий API увеличивает риски безопасности, поскольку старые версии API могут не содержать некоторые патчи или новый, более безопасный функционал. Отсутствующая или ненадлежащий инвентаризация активов приводит к тому, что в компании продолжают использоваться устаревшие API, которые злоумышленникам гораздо легче использовать.

- Налаженная система документации хостов и используемых версий API облегчит управление активами. Необходимо документировать все аспекты API: аутентификацию, перенаправления, конечные точки с параметрами, запросами и ответами и т.д. Кроме того, в случае внедрения улучшений безопасности в новые версии API следует провести анализ рисков в отношении старых версий и определить последующие действия: бэкпортировать улучшения, вывести старые версии из эксплуатации или применить к ним иные меры по повышению безопасности.

API10:2023 **Небезопасное использование** API

- Разработчики часто не проверяют вводные данные, полученные от сторонних API, особенно если те используются известными компаниями. Однако злоумышленники могут скомпрометировать сторонние API для получения доступа к целевым API, что может привести к утечке данных, инъекциям или DoS.

- Для повышения безопасности коммуникации со сторонними API необходимо внедрить проверку всех поступающих данных, обеспечивать соединение по безопасному протоколу (TLS), а также создать перечень ссылок, на которые стороннему API позволено перенаправлять ваш API.

OWASP Top Ten https://owasp.org/www-project-top-ten/

OWASP PROJECTS CHAPTERS EVENTS ABOUT Member Login Store Donate Join

OWASP Top Ten

Main | Translation Efforts | Sponsors | Data 2020

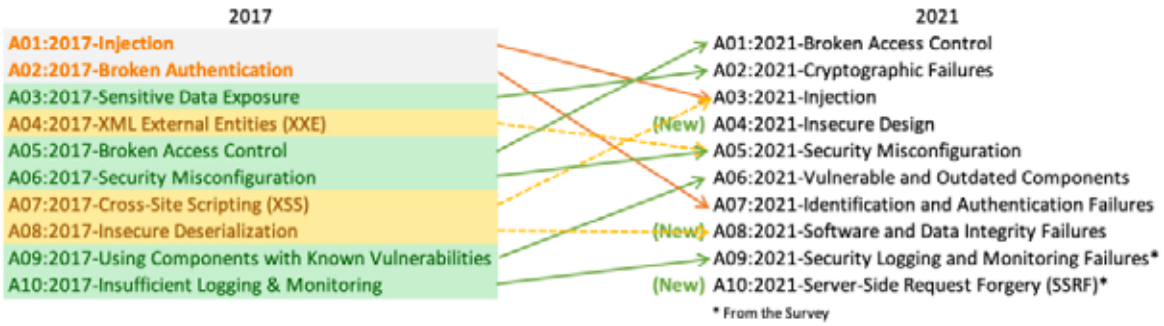
The OWASP Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.

> Globally recognized by developers as the first step towards more secure coding.

Companies should adopt this document and start the process of ensuring that their web applications minimize these risks. Using the OWASP Top 10 is perhaps the most effective first step towards changing the software development culture within your organization into one that produces more secure code.

Top 10 Web Application Security Risks

There are three new categories, four categories with naming and scoping changes, and some consolidation in the Top 10 for 2021.



Watch 281 Star 984

The OWASP® Foundation works to improve the security of software through its community-led open source software projects, hundreds of chapters worldwide, tens of thousands of members, and by hosting local and global conferences.

Project Information

- OWASP Top 10:2021
- Making of OWASP Top 10
- OWASP Top 10:2021 - 20th Anniversary Presentation (PPTX)

Flagship Project
Documentation
Builder
Defender

- Previous Version (2017)

Downloads or Social Links

- OWASP Top 10 2017
- Other languages → tab 'Translation Efforts'

Social

Twitter

Code Repository

repo

10 основных уязвимостей OWASP 2013 и 2017 годов

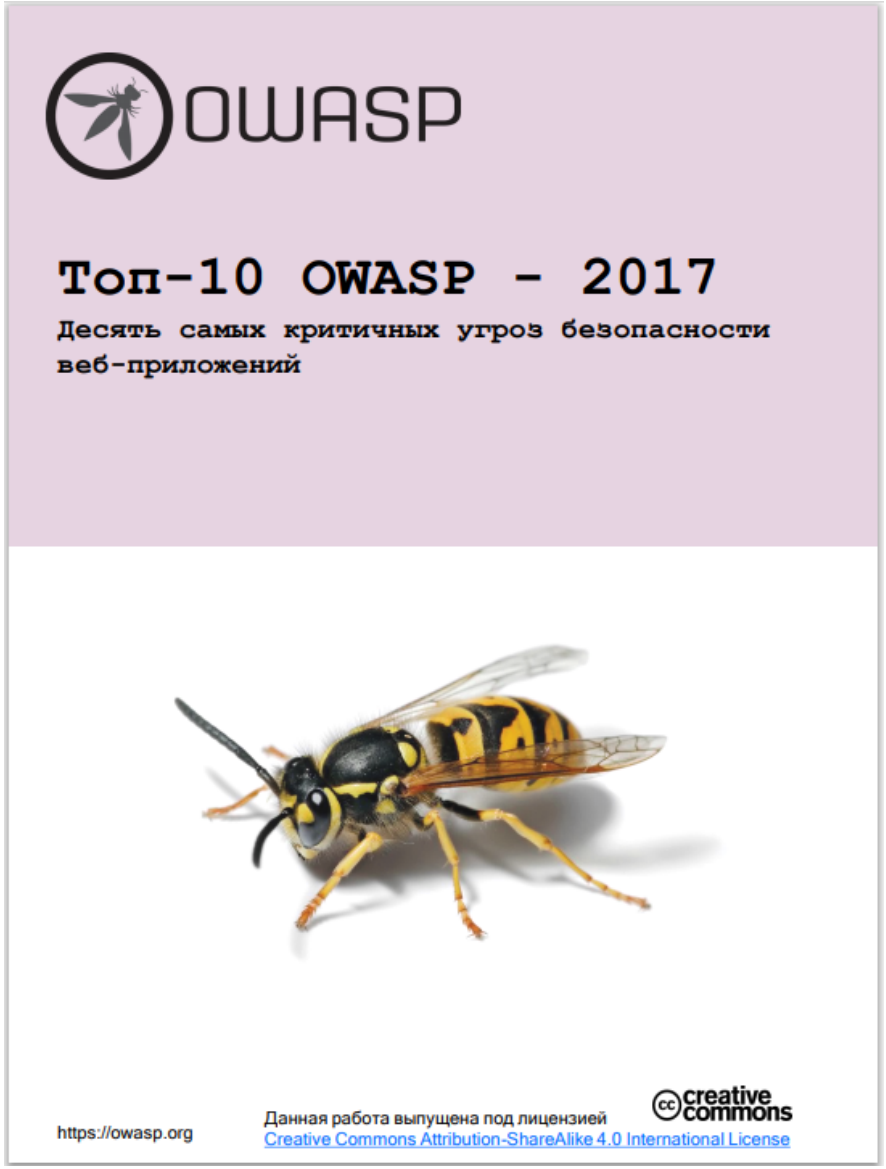
OWASP Top 10 – 2013	→	OWASP Top 10 – 2017
A1 – Внедрение кода	→	A1 – Внедрение кода
A2 – Некорректная аутентификация и управление сессией	→	A2 – Некорректная аутентификация
A3 – Межсайтовый скриптинг	↘	A3 – Утечка чувствительных данных
A4 – Небезопасные прямые ссылки на объекты (объединено с A7)	U	A4 – Внедрение внешних XML-сущностей (новый)
A5 – Небезопасная конфигурация	↘	A5 – Нарушение контроля доступа (объединено)
A6 – Утечка чувствительных данных	↗	A6 – Небезопасная конфигурация
A7 – Отсутствие контроля доступа к функциональному уровню (объединено с A4)	U	A7 – Межсайтовый скриптинг
A8 – Подделка межсайтовых запросов	☒	A8 – Небезопасная десериализация (новый)
A9 – Использование компонентов с известными уязвимостями	→	A9 – Использование компонентов с известными уязвимостями
A10 – Непроверенные перенаправления и переходы	☒	A10 – Отсутствие журналирования и мониторинга (новый)

10 основных уязвимостей OWASP в 2017 года

- **A1:2017-Injection**- Инъекционные атаки
- **A2:2017-Broken Authentication** Нарушенная аутентификация
- **A3:2017-Sensitive Data Exposure** - Незащищённость критичных данных
- **A4:2017-XML External Entities (XXE)** - Внешние объекты XML (XXE)
- **A5:2017-Broken Access Control** - Нарушение контроля доступа
- **A6:2017-Security Misconfiguration** - Небезопасная конфигурация
- **A7:2017-Cross-Site Scripting XSS** - Межсайтовый скриптинг (XSS)
- **A8:2017-Insecure Deserialization** - Небезопасная десериализация
- **A9:2017-Using Components with Known Vulnerabilities** - Использование компонентов с известными уязвимостями
- **A10:2017-Insufficient Logging & Monitoring** - Неэффективный мониторинг

Top 10 Web Application Security Risks
https://owasp.org/www-project-top-ten/
https://owasp.org/www-project-top-ten/2017/

ТОП-10 уязвимостей OWASP 2017 года

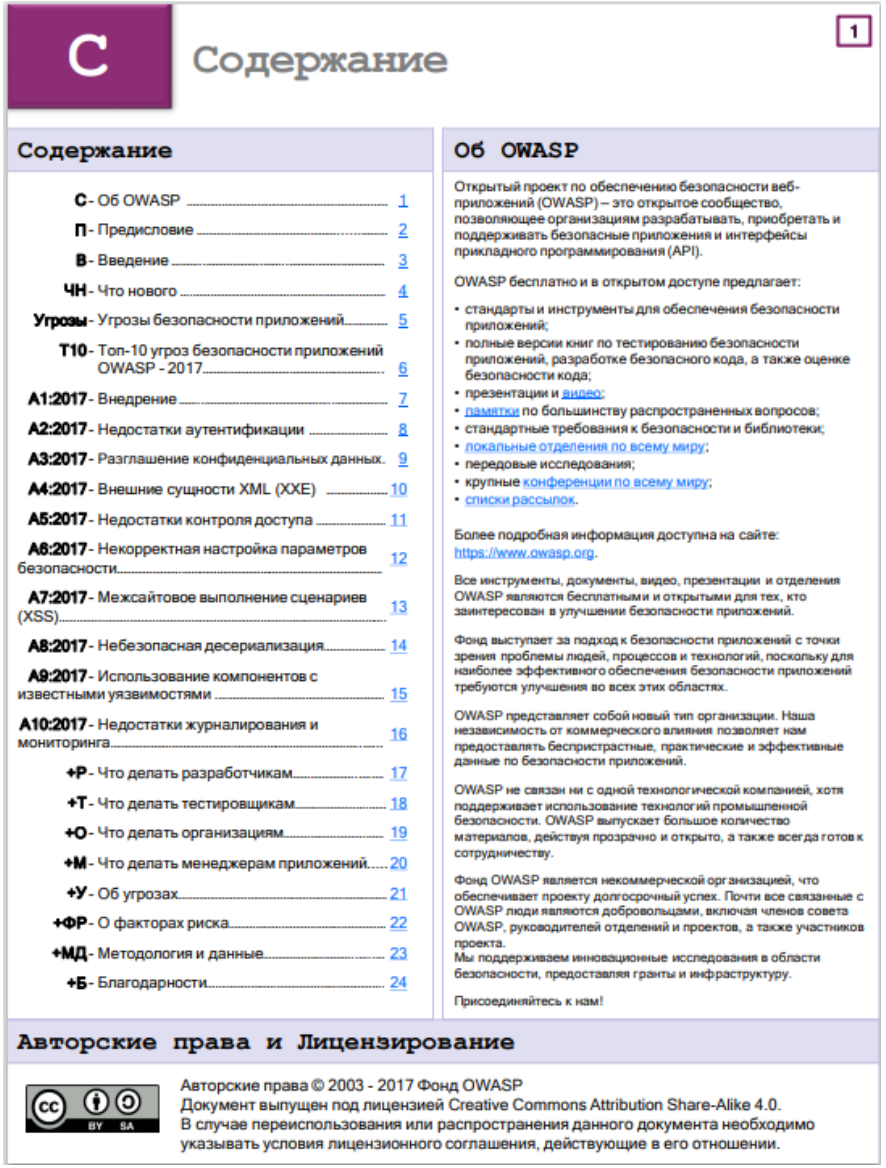


OWASP

Топ-10 OWASP - 2017

Десять самых критичных угроз безопасности веб-приложений

https://owasp.org



С Содержание

1

Содержание

Об OWASP

Открытый проект по обеспечению безопасности веб-приложений (OWASP) – это открытое сообщество, позволяющее организациям разрабатывать, приобретать и поддерживать безопасные приложения и интерфейсы прикладного программирования (API).

OWASP бесплатно и в открытом доступе предлагает:

- стандарты и инструменты для обеспечения безопасности приложений;
- полные версии книг по тестированию безопасности приложений, разработке безопасного кода, а также оценке безопасности кода;
- презентации и видео;
- памятки по большинству распространенных вопросов;
- стандартные требования к безопасности и библиотеки;
- локальные отделения по всему миру;
- передовые исследования;
- крупные конференции по всему миру;
- списки рассылок.

Более подробная информация доступна на сайте: https://www.owasp.org.

Все инструменты, документы, видео, презентации и отделения OWASP являются бесплатными и открытыми для тех, кто заинтересован в улучшении безопасности приложений.

Фонд выступает за подход к безопасности приложений с точки зрения проблемы людей, процессов и технологий, поскольку для наиболее эффективного обеспечения безопасности приложений требуются улучшения во всех этих областях.

OWASP представляет собой новый тип организации. Наша независимость от коммерческого влияния позволяет нам предоставлять беспристрастные, практические и эффективные данные по безопасности приложений.

OWASP не связан ни с одной технологической компанией, хотя поддерживает использование технологий промышленной безопасности. OWASP выпускает большое количество материалов, действуя прозрачно и открыто, а также всегда готов к сотрудничеству.

Фонд OWASP является некоммерческой организацией, что обеспечивает проекту долгосрочный успех. Почти все связанные с OWASP люди являются добровольцами, включая членов совета OWASP, руководителей отделений и проектов, а также участников проекта.
Мы поддерживаем инновационные исследования в области безопасности, предоставляя гранты и инфраструктуру.

Присоединяйтесь к нам!

Авторские права и Лицензирование

https://radware.pro/downloads/OWASP_Top_10-2017-ru.pdf

10 основных уязвимостей OWASP 2017 и 2021 годов

2017			2021	
A1	Внедрение кода		A1	Нарушение контроля доступа
A2	Некорректная аутентификация и управление сессией		A2	Сбои в криптографии
A3	Утечка чувствительных данных		A3	Внедрение кода
A4	Внедрение внешних XML- сущностей (XXE)	(Новая)	A4	Небезопасный дизайн
A5	Нарушение контроля доступа		A5	Небезопасная конфигурация
A6	Небезопасная конфигурация		A6	Уязвимые и устаревшие компоненты
A7	Межсайтовый скриптинг		A7	Ошибки идентификации и аутентификации
A8	Небезопасная десериализация	(Новая)	A8	Нарушение целостности данных и программного обеспечения
A9	Использование компонентов с известными уязвимостями		A9	Журнал безопасности и сбои мониторинга
A10	Отсутствие журналирования и мониторинга	(Новая)	A10	Подделка запросов со стороны сервера (SSRF)

10 основных уязвимостей OWASP в 2021 года

A01:2021	Нарушение контроля доступа	Broken Access Control
A02:2021	Сбои в криптографии	Cryptographic Failures
A03:2021	Внедрение кода	Injection
A04:2021	Небезопасный дизайн	Insecure Design
A05:2021	Неправильная конфигурация	Security Misconfiguration
A06:2021	Уязвимые и устаревшие компоненты	Vulnerable and Outdated Components
A07:2021	Ошибки идентификации и аутентификации	Identification and Authentication Failures
A08:2021	Нарушение целостности данных и программного обеспечения	Software and Data Integrity Failures
A09:2021	Журнал безопасности и сбои мониторинга	Security Logging and Monitoring Failures
A10:2021	Подделка запросов со стороны сервера	SSRF (Server-Side Request Forgery)

A01:2021- **Нарушение контроля доступа (англ.** Broken Access Control)

- **A01:2021- Нарушение контроля доступа (англ. Broken Access Control)** переместился с пятой позиции в категорию с самым серьезным риском для безопасности веб-приложений; согласно внесенным данным, в среднем 3,81% протестированных приложений имели одно или несколько перечислений общих слабых мест (CWE), при этом в данной категории риска было обнаружено более 318 тыс. случаев CWE. 34 CWE, сопоставленные со сломанным контролем доступа, встречались в приложениях чаще, чем в любой другой категории.

A02:2021-**Сбои в криптографии (англ.** Cryptographic Failures)

- **A02:2021-Сбои в криптографии (англ. Cryptographic Failures)** переместился на одну позицию выше и занял второе место, ранее известный как **A3:2017- Утечка чувствительных данных** (англ. Sensitive Data Exposure), который был скорее симптомом, чем главной причиной. Обновленное название фокусируется на сбоях, связанных с криптографией, как это подразумевалось ранее. Эта категория часто приводит к утечке конфиденциальных данных или к нарушению целостности системы.

A03:2021- **Внедрение кода (англ.** Injection)

- **A03:2021- Внедрение кода (англ. Injection)** опускается на третью позицию. 94% приложений были протестированы на наличие той или иной формы инъекций с максимальным уровнем инцидентов 19%, средним уровнем инцидентов 3,37%, а 33 CWE, отнесенные к этой категории, занимают второе место по количеству инцидентов в приложениях - 274 тыс. инцидентов. Межсайтовый скриптинг (англ. Cross-site Scripting (XSS)) теперь является частью этой категории в данном выпуске.

A04:2021- **Небезопасный дизайн (англ.** Insecure Design)

- **A04:2021- Небезопасный дизайн (англ. Insecure Design)** - это новая категория для 2021 года, в которой основное внимание уделяется рискам, связанным с недостатками проектирования. Если мы ходим двигаться в ином направлении, нам нужно больше моделирования угроз, моделей и принципов безопасного проектирования и эталонных архитектур. Небезопасный дизайн не может быть исправлен идеальной реализацией, поскольку по определению необходимые средства контроля безопасности никогда не создавались для защиты от конкретных атак.

A05:2021- **Неправильная конфигурация (англ. Security** Misconfiguration)

- **A05:2021- Неправильная конфигурация (англ. Security Misconfiguration)** безопасности поднялась с 6 на 5 место; 90% приложений были проверены на наличие той или иной формы неправильной конфигурации, со средним коэффициентом встречаемости 4,5%, и более 208 тысяч случаев CWE были отнесены к этой категории риска. Поскольку все большее число программ переходит на высококонфигурируемое программное обеспечение, неудивительно, что эта категория повысилась. Бывшая категория **A4:2017- Внедрение внешних XML сущностей** (англ. XML External Entities (XXE)) теперь является частью этой категории риска.

A06:2021-**Уязвимые и устаревшие компоненты (англ.** Vulnerable and Outdated Components)

- **A06:2021-Уязвимые и устаревшие компоненты (англ. Vulnerable and Outdated Components)** ранее называлась **A09:2017 Использование компонентов с известными уязвимостями**, она занимает второе место в Топ-10 по результатам опроса сообщества, но также имеет достаточно данных для попадания в Топ-10 по результатам наших анализов. Эта категория поднялась с № 9 в 2017 году и является известной проблемой, которую мы с трудом тестируем и оцениваем его риски. Это единственная категория, в которой нет общих уязвимостей (CVE), сопоставленных с включенными в нее CWE, поэтому в их оценках по умолчанию учитывается эксплойт и вес воздействия 5,0.

A07:2021- **Ошибки идентификации и аутентификации (англ.** Identification and Authentication Failures)

- **A07:2021- Ошибки идентификации и аутентификации (англ. Identification and Authentication Failures)** ранее называлась **A02:2017 Некорректная аутентификация** (англ. Broken Authentication) и опустилась со второго места на седьмое место, теперь она включает в себя CWE, которые больше связаны с ошибками идентификации. Эта категория все еще является неотъемлемой частью Топ-10, но увеличение количества стандартизированных систем, похоже, способствует уменьшению некорректной аутентификации.

A08:2021- **Нарушение целостности данных и программного обеспечения (англ. Software** and Data Integrity Failures)

- **A08:2021- Нарушение целостности данных и программного обеспечения (англ. Software and Data Integrity Failures)** - новая категория для 2021 года, посвященная принятию предположений, связанных с обновлениями программного обеспечения, критическими данными и CI/CD конвейерами без проверки на целостность. Одно из самых высоких по весу воздействий из данных Common Vulnerability and Exposures/Common Vulnerability Scoring System (CVE/CVSS) сопоставлено с 10 CWE в этой категории. **A8:2017- Небезопасная десериализация** (англ. Insecure Deserialization) теперь является частью этой более крупной категории.

A09:2021- **Журнал безопасности и сбои мониторинга (англ.** Security Logging and Monitoring Failures)

- **A09:2021- Журнал безопасности и сбои мониторинга (англ. Security Logging and Monitoring Failures)** ранее была **A10:2017- Отсутствие журналирования и мониторинга** (англ. Insufficient Logging & Monitoring) и добавлена из опроса сообщества Топ-10 (#3), поднявшись с десятого на девятое место. Эта категория расширена и включает больше типов сбоев, ее сложно тестировать, и она не очень хорошо представлена в данных CVE/CVSS. Однако сбои в этой категории могут напрямую влиять на видимость, оповещение об инцидентах и проведение экспертизы.

A10:2021- **Подделка запросов со стороны сервера или же SSRF (англ. Server**-Side Request Forgery)

- **A10:2021- Подделка запросов со стороны сервера или же SSRF (англ. Server-Side Request Forgery)** добавлена в Топ-10 опроса сообщества (#1). Данные показывают относительно низкий уровень инцидентов при охвате тестирования выше среднего, а также оценки выше среднего по потенциалу применения и воздействия. Эта категория представляет собой сценарий, в котором члены сообщества безопасности говорят нам, что это важно, хотя на данный момент это не показано в данных.

Банк данных угроз безопасности информации

Типовые уязвимости веб-приложений

https://bdu.fstec.ru/webvulns

Банк данных угроз безопасности информации

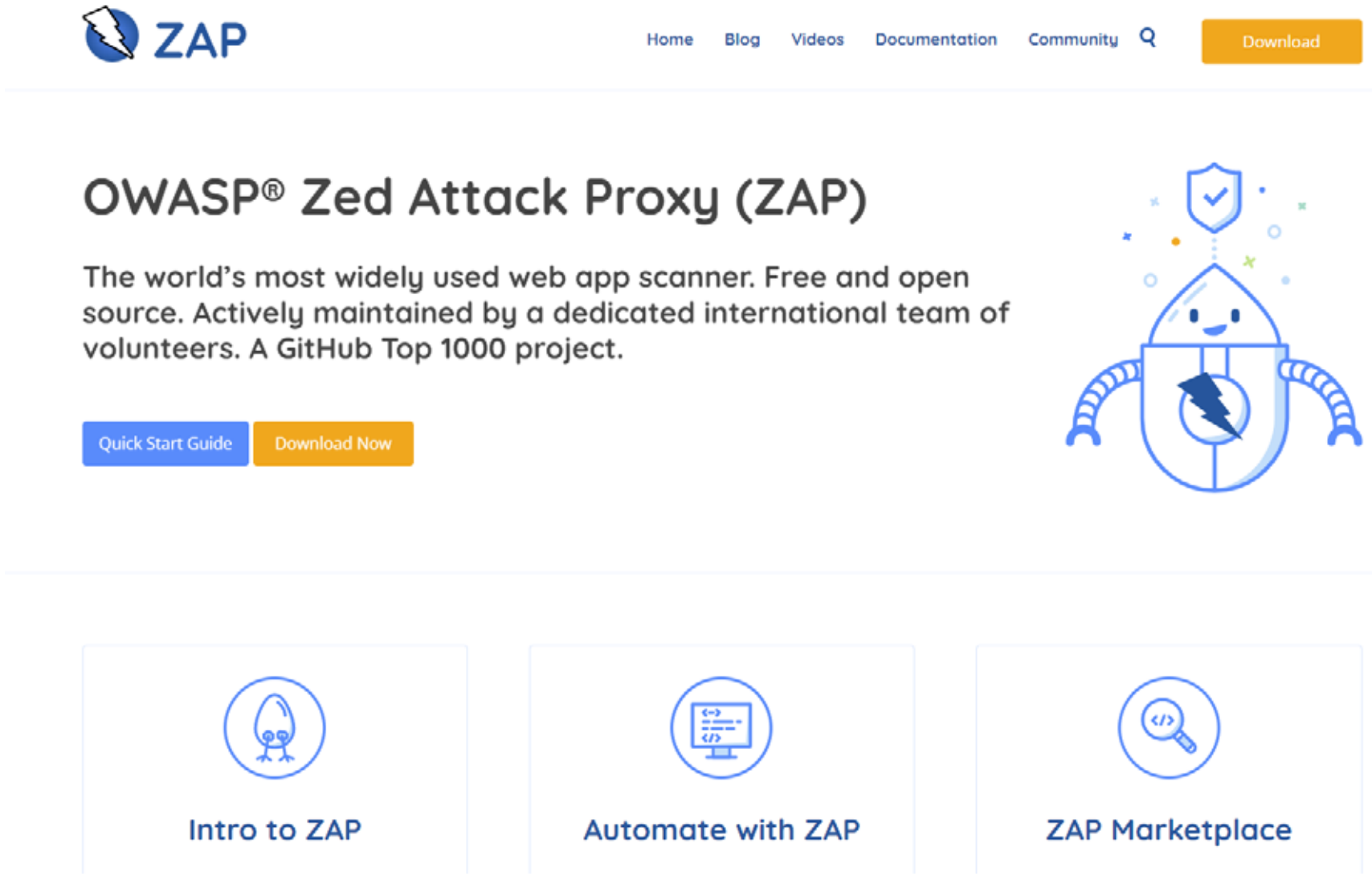
- Типовые уязвимости веб-приложений

- https://bdu.fstec.ru/webvulns

Программное обеспечение для поиска уязвимостей в веб-приложениях

OWASP ZAP www.zaproxy.org

Сканер для поиска уязвимостей в веб-приложениях



OWASP ZAP

- **OWASP ZAP — сканер веб-приложений, основанный на методике DAST (Dynamic Application Security Testing).**
- В русском варианте этот метод принято называть методом тестирования «черного ящика». Методика позволяет обнаруживать проблемы безопасности в работающем приложении или веб-сайте при помощи их сканирования на известные уязвимости. К таким уязвимостям можно отнести SQL-инъекции, межсайтовый скриптинг (XSS), Clickjacking и т.д.

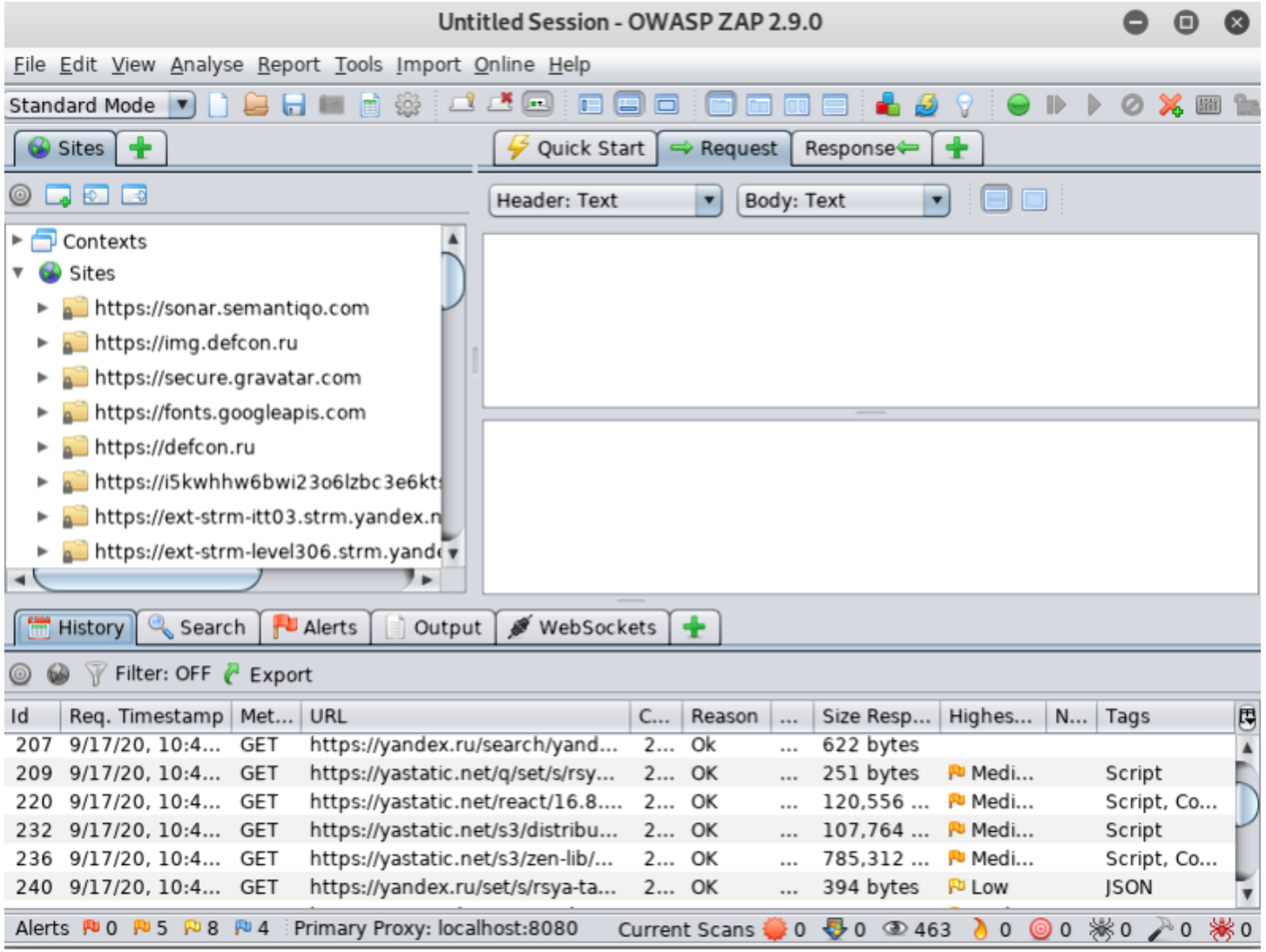
OWASP ZAP. Преимущества

- Кроссплатформенность — поддержка всех основных ОС (Windows, Linux, MacOS);
- Бесплатный проект с открытым исходным кодом;
- Поддержка плагинов для расширения функциональности;
- Возможность работы как через графический интерфейс (GUI), так и через интерфейс командной строки;
- Обширный набор функций — от активного/пассивного сканирования и до сканирования API и AJAX;
- Простота использования. Идеально подходит и для начинающих специалистов в ИБ и для профессионалов.

OWASP ZAP. Принцип работы

- При сканировании ZAP создает собственный прокси-сервер, через который обрабатываются все запросы на сканирование. ZAP включает в себя специальные поисковые роботы (краулеры), которые выполняют идентификацию уязвимостей. Прокси-сервер располагается между браузером пользователя и конечным веб-приложением. Схема работы изображена ниже.

OWASP ZAP

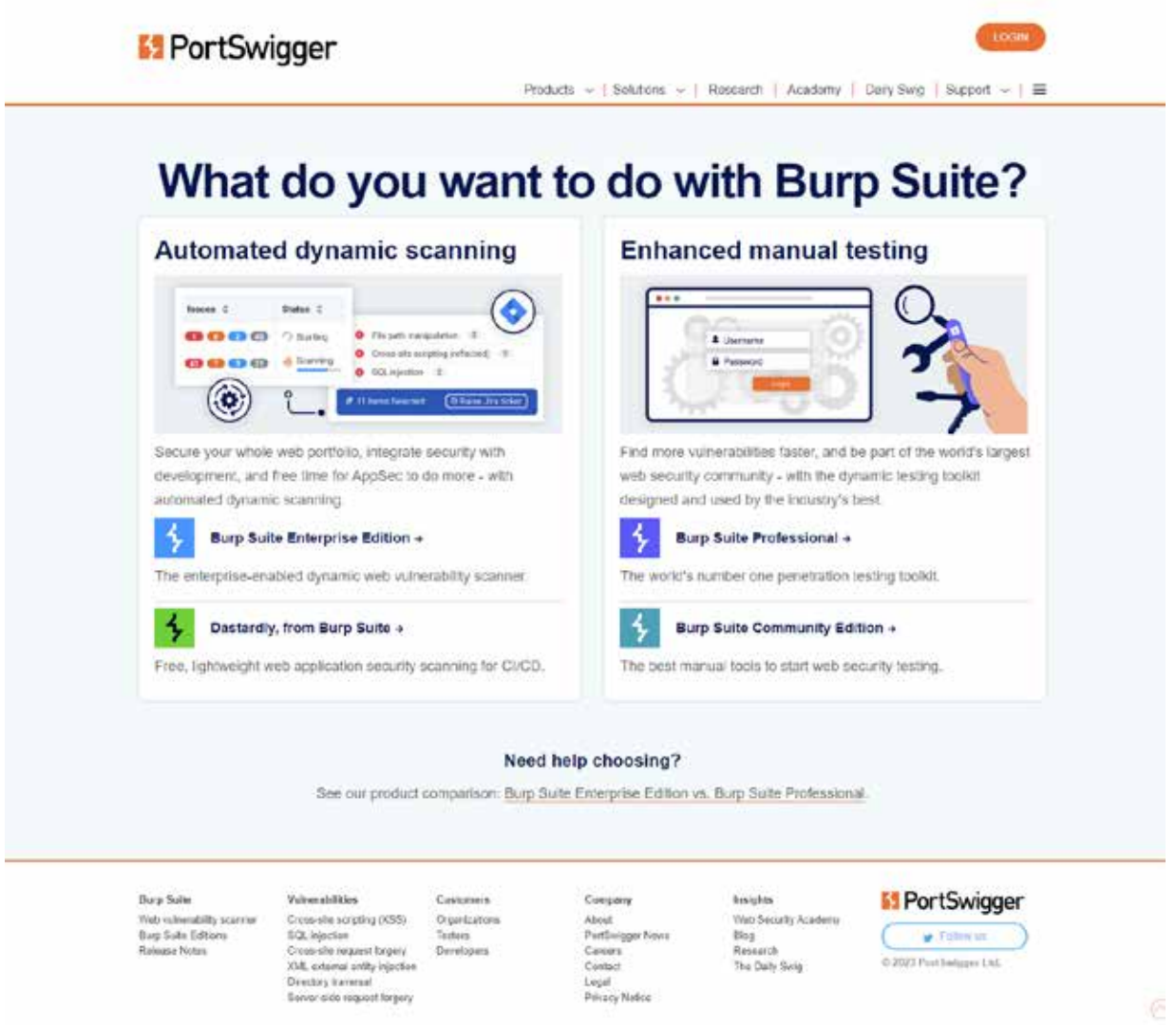

Untitled Session - OWASP ZAP 2.9.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Text
Contexts
Sites
https://sonar.semantiqo.com
https://img.defcon.ru
https://secure.gravatar.com
https://fonts.googleapis.com
https://defcon.ru
https://i5kwhhw6bwi23o6lzbc3e6kt
https://ext-strm-itt03.strm.yandex.n
https://ext-strm-level306.strm.yand
History
Search
Alerts
Output
WebSockets
Filter: OFF
Export
Id Req. Timestamp Met... URL C... Reason ... Size Resp... Highes... N... Tags
207 9/17/20, 10:4... GET https://yandex.ru/search/yand... 2... Ok ... 622 bytes
209 9/17/20, 10:4... GET https://yastatic.net/q/set/s/rsy... 2... OK ... 251 bytes Medi... Script
220 9/17/20, 10:4... GET https://yastatic.net/react/16.8.... 2... OK ... 120,556 ... Medi... Script, Co...
232 9/17/20, 10:4... GET https://yastatic.net/s3/distribu... 2... OK ... 107,764 ... Medi... Script
236 9/17/20, 10:4... GET https://yastatic.net/s3/zen-lib/... 2... OK ... 785,312 ... Medi... Script, Co...
240 9/17/20, 10:4... GET https://yandex.ru/set/s/rsya-ta... 2... OK ... 394 bytes Low JSON
Alerts 0 5 8 4 Primary Proxy: localhost:8080 Current Scans 0 0 463 0 0 0 0 0

OWASP ZAP

- OWASP ZAP – Официальный сайт https://www.zaproxy.org
- GitHub OWASP ZAP https://github.com/zaproxy
- Обзор OWASP ZAP. Сканер для поиска уязвимостей в веб-приложениях. Инструкция по работе https://habr.com/ru/company/first/blog/709586/
- Пентест вебсайта с помощью Owasp Zap https://habr.com/ru/company/alexhost/blog/530110/

Burp Suite

https://portswigger.net/burp

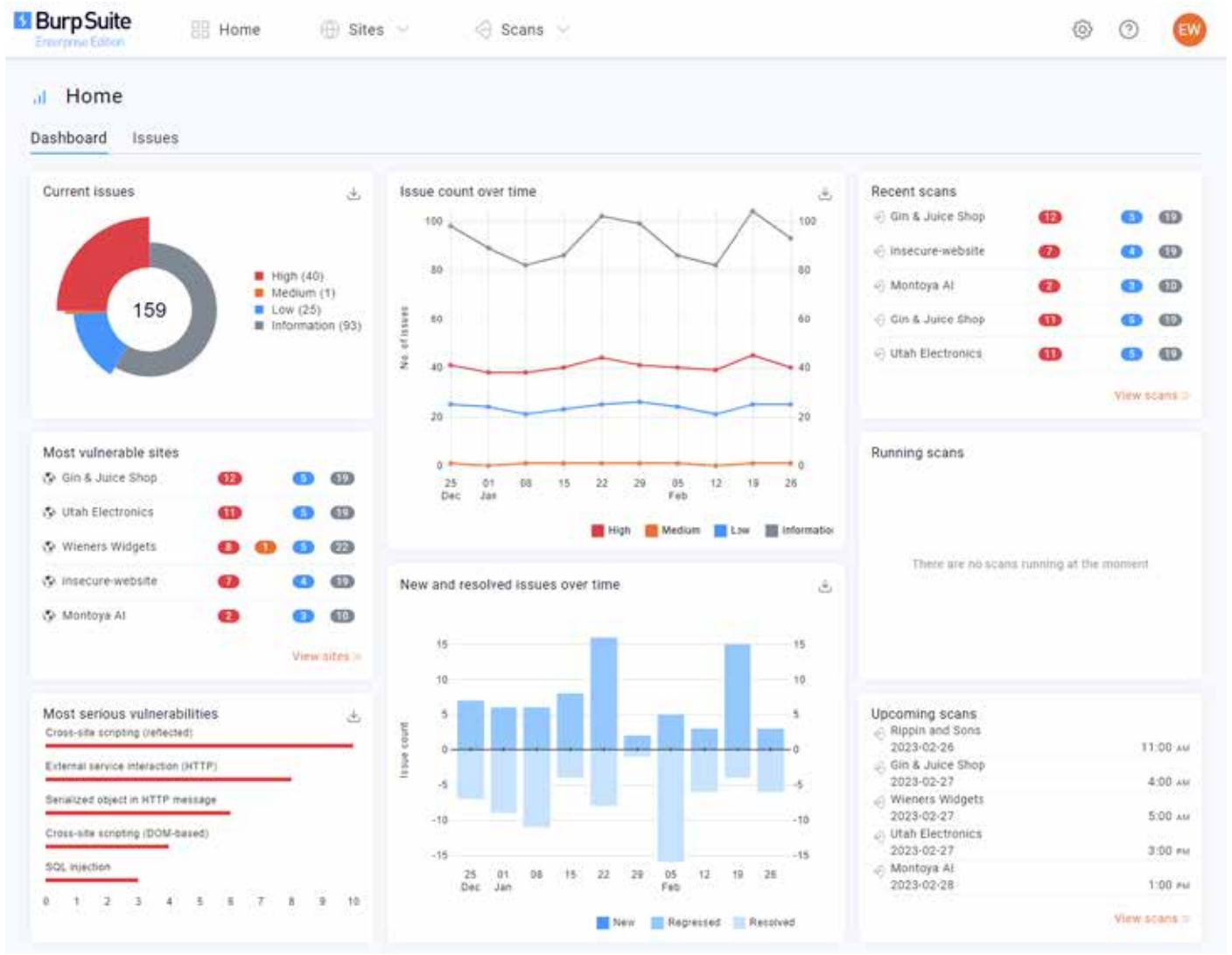


Burp Suite

- **Burp Suite** – это мультитул для проведения аудита безопасности веб-приложений. Содержит инструменты для составления карты веб-приложения, поиска файлов и папок, модификации запросов, фаззинга, подбора паролей и многое другое. Также существует магазин дополнений BApp store, содержащий дополнительные расширения, увеличивающие функционал приложения. Стоит отметить и появление в последнем релизе мобильного помощника для исследования безопасности мобильных приложений — MobileAssistant для платформы iOS.

- **Burp Suite** — это интегрированная платформа, предназначенная для проведения аудита веб-приложения, как в ручном, так и в автоматических режимах. Содержит интуитивно понятный интерфейс со специально спроектированными табами, позволяющими улучшить и ускорить процесс атаки. Сам инструмент представляет из себя проксирующий механизм, перехватывающий и обрабатывающий все поступающие от браузера запросы. Имеется возможность установки сертификата burp для анализа https соединений.

Burp Suite


Burp Suite
Enterprise Edition
Home
Sites
Scans
EW
Home
Dashboard
Issues
Current issues
159
High (40)
Medium (1)
Low (25)
Information (93)
Issue count over time
No. of issues
25 Dec
01 Jan
08
15
22
29
05 Feb
12
19
26
High
Medium
Low
Recent scans
Gin & Juice Shop
insecure-website
Montoya AI
Gin & Juice Shop
Utah Electronics
View scans
Most vulnerable sites
Gin & Juice Shop
Utah Electronics
Wieners Widgets
insecure-website
Montoya AI
View sites
Running scans
There are no scans running at the moment
New and resolved issues over time
Issue count
New
Regressed
Resolved
Most serious vulnerabilities
Cross-site scripting (reflected)
External service interaction (HTTP)
Serialized object in HTTP message
Cross-site scripting (DOM-based)
SQL injection
Upcoming scans
Rippin and Sons
2023-02-26
11:00 AM
Gin & Juice Shop
2023-02-27
4:00 AM
Wieners Widgets
2023-02-27
5:00 AM
Utah Electronics
2023-02-27
3:00 PM
Montoya AI
2023-02-28
1:00 PM
View scans

Burp Suite

- Burp Suite
  https://portswigger.net/burp
- Burp Suite Enterprise Edition DEMO
  https://enterprise-demo.portswigger.net
- Burp Suite: швейцарский армейский нож для тестирования веб-приложений
  https://habr.com/ru/post/328382/
- Telegram: BurpSuite (not official)
  https://t.me/burpsuite

Полезная нагрузка (payload)

Payloads All The Things — один из наиболее популярных репозиториев полезных нагрузок, содержит следующие категории (которых хватит практически на «все случаи жизни» тестирования веб-приложений) https://github.com/swisskyrepo/PayloadsAllTheThings

SecLists — еще одна популярная «сборка» пейлоадов, предназначенная как для тестирования веб-приложений, так и сетевой инфраструктуры. https://github.com/danielmiessler/SecLists

Intruder Payloads — репозиторий пейлодов от автора BruteX, Sn1per и Findsploit. Помимо листов фаззинга содержит атакующие пейлоды для различных инструментов (repeater, intruder) утилиты BurpSuite https://github.com/1N3/IntruderPayloads

Полезная нагрузка (payload)

Fuzzdb содержит списки различных пейлоадов. Для тестирования веб-приложений полезным будет список «attack»
https://github.com/fuzzdb-project/fuzzdb/tree/master/attack

Foospidy содержит список пейлодов для выявления критичных файлов и папок на веб-сервере
https://github.com/foospidy/payloads/tree/master/owasp/dirbuster

Защита информации

Тема: Защита internet ресурсов, сайтов

Благодарю за внимание

КУТУЗОВ Виктор Владимирович

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»
Республика Беларусь, Могилев, 2024

Список использованных источников

1. Рабочая программа дисциплины «Защита информации» / Кутузов В.В. – Могилев : Белорусско-Российский университет, 2019
2. Фотографии и картинки взяты с сайтов Яндекс.Картинки и Гугл.Картинки, иконки с flaticon.com