ВОКРУГ РБПО ЗА 25 ВЕБИНАРОВ

ГОСТ Р 56939-2024

Вебинар 19. Нефункциональное тестирование



PVS-Studio

УЧЕБНЫЙ ЦЕНТР МАСКОМ

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Нефункциональное тестирование

- **Нефункциональное тестирование** – проверка реализуемости нефункциональных требований:
  - Производительность (нагрузочное и стресс-тестирование)
  - Удобство использования.
  - Масштабируемость.
  - Отсутствие ошибок при установке.
  - Безопасность.
  - Переносимость/портируемость.
  - Надежность.
  - Отказоустойчивость.
  - Совместимость с другим ПО.

Атаки и уязвимости

ГОСТ Р 56939-2024 п 3.20:

- **уязвимость программы**: Недостаток программы, который может быть использован для реализации угроз безопасности информации.

- *Примечание - Уязвимость программы может быть результатом ее разработки без учета требований по обеспечению безопасности информации, вследствие наличия ошибок проектирования или реализации или в результате применения небезопасных инструментальных средств и языков программирования, имеющих ошибки проектирования, реализации или конфигурирования.*

Атаки и уязвимости

ГОСТ Р 51275-2006, п 3.11, ГОСТ Р 56939-2024 п.3.7:

- **компьютерная атака**: Целенаправленное несанкционированное воздействие на информацию, на ресурс автоматизированной информационной системы или получение несанкционированного доступа к ним с применением программных или программно-аппаратных средств.

Атаки и уязвимости

ГОСТ Р 56546-2015 «Защита информации. Уязвимости информационных систем. Классификация уязвимостей информационных систем»

- По области происхождения
- По типу недостатка ИС
- По месту возникновения/проявления

Атаки и уязвимости

По области происхождения:

- 3.8 уязвимость кода: Уязвимость, появившаяся в процессе разработки программного обеспечения.
- 3.9 уязвимость конфигурации: Уязвимость, появившаяся в процессе задания конфигурации (применения параметров настройки) программного обеспечения и технических средств информационной системы.
- 3.10 уязвимость архитектуры: Уязвимость, появившаяся в процессе проектирования информационной системы.
- 3.11 организационная уязвимость: Уязвимость, появившаяся в связи с отсутствием (или недостатками) организационных мер защиты информации в информационной системе и (или) несоблюдением правил эксплуатации системы защиты информации информационной системы, требований организационно-распорядительных документов по защите информации и (или) несвоевременном выполнении соответствующих действий должностным лицом (работником) или подразделением, ответственным за защиту информации.
- 3.12 многофакторная уязвимость: Уязвимость, появившаяся в результате наличия нескольких недостатков различных типов.

Атаки и уязвимости

По типам недостатков ИС: недостатки, связанные с

- неправильной настройкой параметров ПО
- неполнотой проверки вводимых (входных) данных
- возможностью прослеживания пути доступа к каталогам
- возможностью перехода по ссылкам
- возможностью внедрения команд ОС
- межсайтовым скриптингом
- внедрением интерпретируемых операторов языков программирования или разметки
- произвольного кода
- переполнением буфера памяти

Подробнее и с примечанием – ссылка на ГОСТ

Атаки и уязвимости

(продолжение) По типам недостатков ИС: недостатки, связанные с

- неконтролируемой форматной строкой
- вычислениями (некорректная обработка диапазонов данных, знака числа и др.)
- приводящие к утечке/раскрытию информации ограниченного доступа
- управлением полномочиями (учетными данными)
- управлением разрешениями, привилегиями и доступом
- аутентификацией
- криптографическими преобразованиями
- подменой межсайтовых запросов
- приводящие к "состоянию гонки«
- управлением ресурсами
- иные типы недостатков

Подробнее и с примечанием – ссылка на ГОСТ

Атаки и уязвимости

По месту возникновения/проявления

- уязвимости в общесистемном (общем) ПО
  - уязвимости ОС: уязвимости ФС, загрузки, связанные с наличием средств разработки и отладки ПО и др.
  - уязвимости систем управления базами данных: уязвимости серверной, клиентской частей СУБД, инструментов и утилит, исполняемых объектов баз данных (SQL-процедуры, триггеры) и др.
  - уязвимости иных типов общесистемного (общего) ПО

Атаки и уязвимости

По месту возникновения/проявления (продолжение)

- в прикладном ПО
- в специальном ПО
- в технических средствах*
- в портативных технических средствах**
- в сетевом оборудовании
- в средствах защиты информации

* интегральные схемы, средства в/в, контроллеры, их микропрограммное обеспечение

** мобильные устройства

Подробнее - ссылка

Атаки и уязвимости

Базы/классификаторы уязвимостей:

- Common Vulnerabilities and Exposures (CVE)
- Common Weakness Enumeration (CWE)
- Mitre
- Common Attack Pattern Enumeration and Classification (CAPEC)
- ФСТЭК ИСПДн

Прочая классификация уязвимостей:

- По угрозам: связанные с нарушением целостности, доступности и конфиденциальности
- По субъективному индексу критичности
  - Common Vulnerability Scoring System (CVSS) (см. далее

Common Vulnerability Scoring System (CVSS)

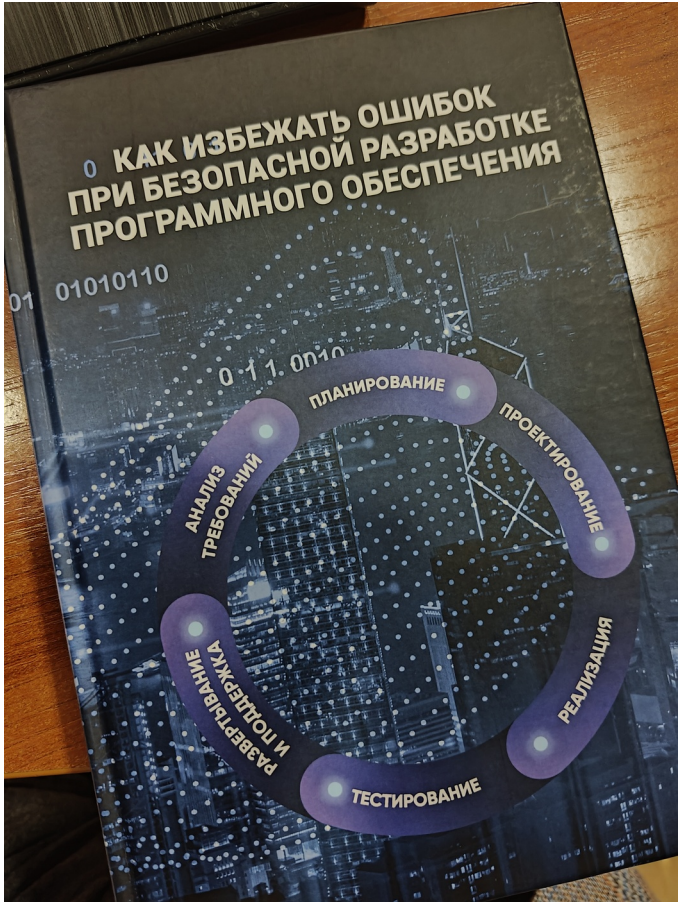
- Спецификация
- Онлайн-калькулятор NIST

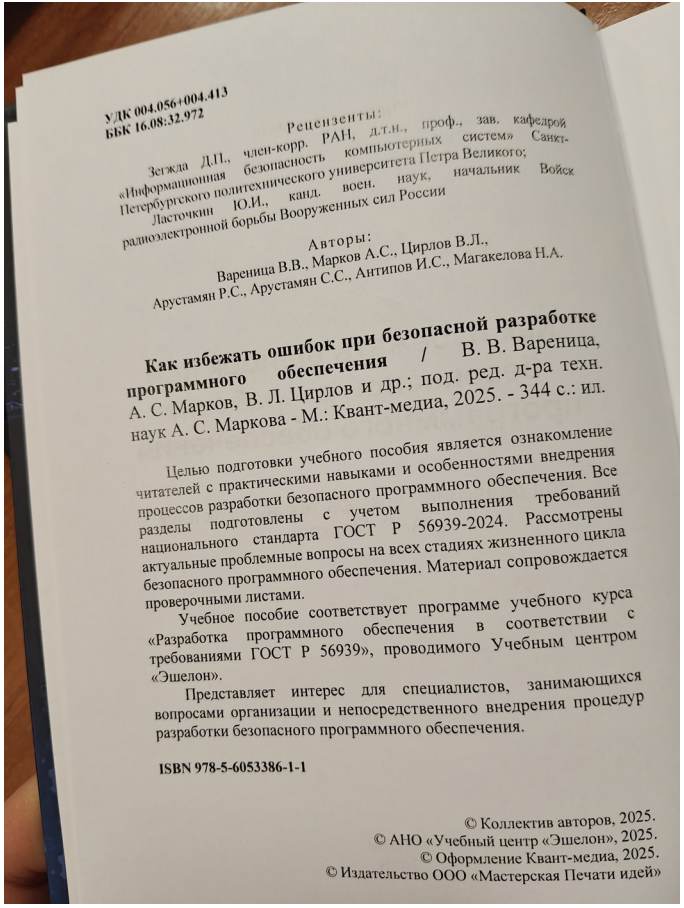
CVSS v2.0 Ratings	
Low	0.0-3.9
Medium	4.0-6.9
High	7.0-10.0

CVSS v3.0 Ratings	
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Источник мудрости по РБПО





УДК 004.056+004.413
ББК 16.08:32.972

Рецензенты:
Зегжда Д.П., член-корр. РАН, д.т.н., проф., зав. кафедрой «Информационная безопасность компьютерных систем» Санкт-Петербургского политехнического университета Петра Великого;
Ласточкин Ю.И., канд. воен. наук, начальник Войск радиоэлектронной борьбы Вооруженных сил России

Авторы:
Вареница В.В., Марков А.С., Цирлов В.Л.,
Арустамян Р.С., Арустамян С.С., Антипов И.С., Магакелова Н.А.

Как избежать ошибок при безопасной разработке программного обеспечения / В. В. Вареница, А. С. Марков, В. Л. Цирлов и др.; под. ред. д-ра техн. наук А. С. Маркова - М.: Квант-медиа, 2025. - 344 с.: ил.

Целью подготовки учебного пособия является ознакомление читателей с практическими навыками и особенностями внедрения процессов разработки безопасного программного обеспечения. Все разделы подготовлены с учетом выполнения требований национального стандарта ГОСТ Р 56939-2024. Рассмотрены актуальные проблемные вопросы на всех стадиях жизненного цикла безопасного программного обеспечения. Материал сопровождается проверочными листами.

Учебное пособие соответствует программе учебного курса «Разработка программного обеспечения в соответствии с требованиями ГОСТ Р 56939», проводимого Учебным центром «Эшелон».

Представляет интерес для специалистов, занимающихся вопросами организации и непосредственного внедрения процедур разработки безопасного программного обеспечения.

ISBN 978-5-6053386-1-1

© Коллектив авторов, 2025.
© АНО «Учебный центр «Эшелон», 2025.
© Оформление Квант-медиа, 2025.
© Издательство ООО «Мастерская Печати идей»

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Процесс № 19: Нефункциональное тестирование

— часть жизненного цикла безопасной разработки ПО

— регламентирован ГОСТ Р 56939-2024

— направлен на выявление уязвимостей, связанных с безопасностью, надёжностью и устойчивостью функционирования ПО

— включает тестирование, имитирующее действия потенциального нарушителя

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Цель процесса

Согласно ГОСТ Р 56939-2024, цель процесса № 19 —

— подтверждение соответствия поведения ПО при атаке, модели угроз и архитектуры требованиям безопасности,

— обнаружение недостатков (уязвимостей) путём выполнения нефункциональных тестов,

— в том числе с имитацией действий потенциального нарушителя.

Этот процесс относится к этапу тестирования жизненного цикла разработки ПО.

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Терминология: два подхода

Общеупотребительное понимание (международное):

— нефункциональное тестирование = проверка качества ПО по характеристикам, отличным от функциональности

— включает: удобство использования, производительность, совместимость, мобильность, безопасность и др.

Термин в ГОСТ Р 56939-2024 (2024 г.):

— строго ограничивает процесс проверкой безопасности, надёжности и устойчивости ПО

— фактически сводится к систематическому тестированию на проникновение в рамках безопасного жизненного цикла

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Чем не является нефункциональное тестирование по ГОСТ Р 56939-2024?

— Это НЕ тестирование удобства интерфейсов

— Это НЕ нагрузочное/стресс-тестирование ради производительности

— Это НЕ проверка совместимости версий

По новому стандарту — это именно проверка стойкости ПО к вредоносному воздействию, в том числе при активных атаках.

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Основные проверки в рамках процесса № 19

Стандарт предписывает тестирование следующих аспектов:

— стойкость к локальным и сетевым взаимодействиям
— безопасность работы с конфиденциальными данными
— корректность файловых операций
— защита двоичных файлов и система управления секретами
— безопасность сетевых протоколов
— механизмы аутентификации и авторизации
— обработка данных от потенциального нарушителя
— реализация мер по снижению критичности выявленных угроз

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Источники для формирования тест-кейсов

Для разработки тестов рекомендуется использовать:

— проект архитектуры ПО (включая сторонние компоненты)

— результаты моделирования угроз

— данные статического анализа исходного кода

— данные экспертной оценки кода

— результаты динамического анализа

Это обеспечивает целевую и рискоориентированную проверку.

Методики нефункционального тестирования

Наиболее авторитетные источники:

1. OWASP Testing Guide (2020)
2. PTES (Penetration Testing Execution Standard, 2017)
3. NIST SP 800-115 (2008)
4. OSSTMM (2010)
5. PCI DSS Penetration Testing Guidance (2015)
6. FedRAMP Penetration Test Guidance (2017)
7. ISSAF (2006)
8. Penetration Testing Framework (PTF, 2016)

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Типовые уязвимости веб-приложений (примеры)

VUL P1 — XSS (межсайтовый скриптинг)

VUL P2 — CSRF (межсайтовая подделка запроса)

VUL P3 — SQL-инъекции

VUL P4 — переполнение буфера

VUL P5 — отказ в обслуживании (DoS)

VUL P6 — path traversal (обход каталогов)

Эти примеры используются при составлении проверочного перечня.

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Документирование: паспорт уязвимости

Каждая выявленная уязвимость оформляется в виде паспорта, включающего:

— наименование и идентификатор
— описание и класс
— CVSS-оценка критичности
— статус (подтверждена/опровергнута)
— наличие эксплойта
— рекомендации по устранению
— оценку экспертной лаборатории

Формат регламентирован ГОСТ Р 56545-2015.

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Пример паспорта уязвимости

Наименование: XSS (Cross-Site Scripting)

Класс: уязвимость кода

Описание: возможность выполнения произвольного кода в браузере пользователя

CVSS 3.1: 8.6 (High) — AV:N/AC:L/PR:H/UI:R/S:C/C:H/I:H/A:H

Рекомендации: экранирование, фильтрация, использование whitelist

Статус: подтверждена в ходе испытаний

Проверочный лист (чек-лист) по ГОСТ Р 56939-2024

Обязательные требования:

✔️ Проведение нефункционального тестирования

✔️ Наличие регламента тестирования (сценарии, роли, описание нарушителя)

✔️ Тестирование локальных и сетевых интерфейсов

✔️ Выполнение тестов с имитацией действий нарушителя

✔️ Корректировка модели угроз и архитектуры по результатам

УЧЕБНЫЙ ЦЕНТР МАСКОМ

Интеграция в жизненный цикл разработки

Нефункциональное тестирование:

— проводится на этапе тестирования,

— но использует артефакты с ранних этапов (архитектура, модель угроз),

— результаты влияют на дальнейшую доработку ПО,

— является обязательной частью подтверждения соответствия требованиям безопасности.

Заключение

— Нефункциональное тестирование по ГОСТ Р 56939-2024 — это не «опциональный QA», а систематическая проверка стойкости ПО к атакам.

— Процесс тесно связан с моделированием угроз и анализом кода.

— Требует применения отраслевых методик (OWASP, PTES и др.).

— Результаты должны быть оформлены и использованы для улучшения архитектуры и кода.

Безопасность — не фича, а процесс. И процесс №19 — его неотъемлемая часть.

Сделай свой проект
чистым и безопасным
вместе с PVS-Studio

VOKRUG_RBPO25

PVS-Studio

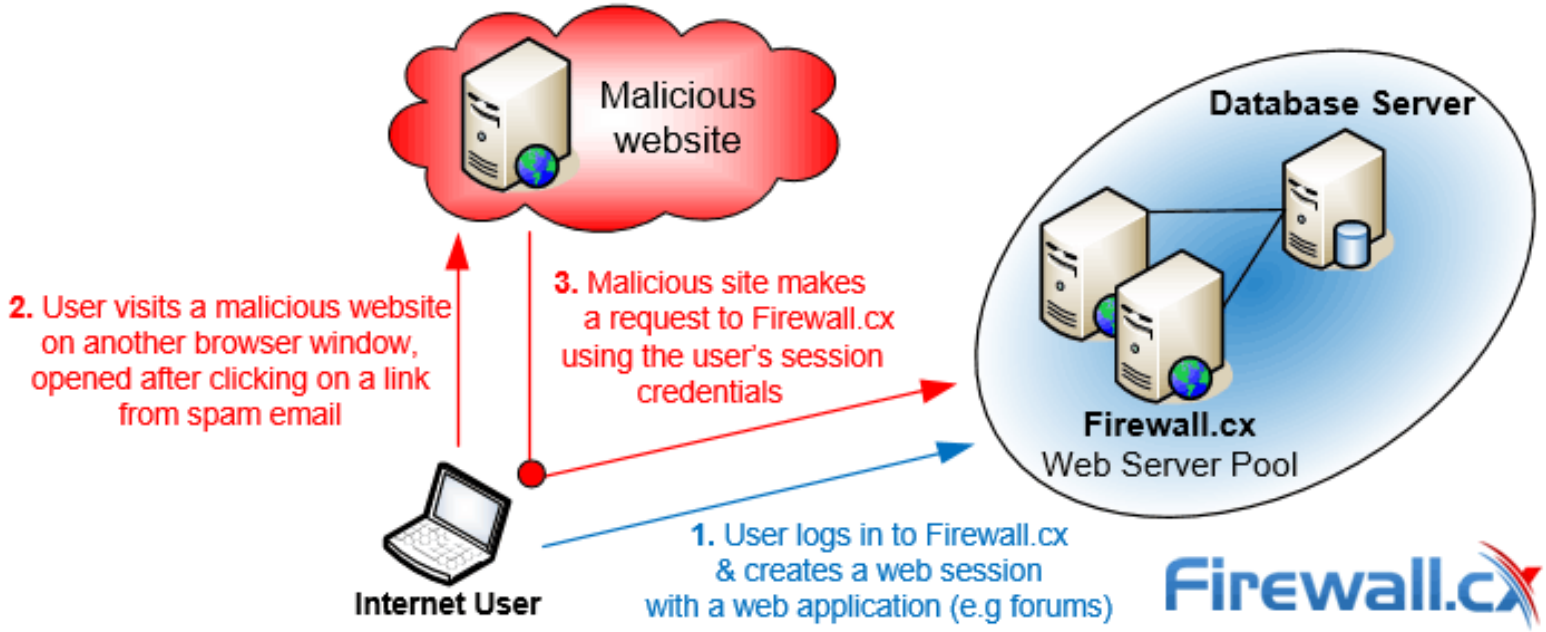
Получи 10% скидку
на курсы «М БРПО»
в Учебном Центре «МАСКОМ»

VOKRUG_RBPO25

MACKOM
УЧЕБНЫЙ ЦЕНТР

Примеры уязвимостей

CSRF (Cross-Site Request Forgery), подделка межсайтовых запросов



Примеры уязвимостей

Злоумышленник побуждает пользователя-жертву выполнить действия, которые он не намеревался совершать (но не может получить или проанализировать ответ сервера)

Атака возможна благодаря механике Cookies браузера

Для совершения атаки:

1. …

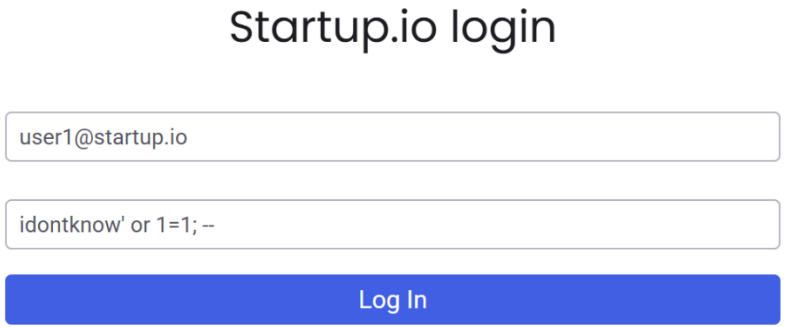
Защита:

- добавление в cookies защищаемого сайта специально сгенерированного ключа/токена, с опцией SameSite=Strict.
- добавление специально сгенерированного ключа/токена и в Cookie, и в HTML.

Примеры уязвимостей

SQL Injection

```
sqlQuery = "SELECT email FROM credentials " +
           "WHERE email='" + req.getParameter("email") + "' " +
           "AND password='" + req.getParameter("password") + "'";
```




```
sqlQuery = "SELECT email FROM credentials " +
           "WHERE email='user1@startup.io " +
           "AND password='idontknow' OR 1=1";
---
TRUE
```

Примеры уязвимостей

SQL Injection. Интерактивные примеры

- https://www.w3schools.com/sql/sql_injection.asp
- https://learn.snyk.io/lesson/sql-injection/

Защита – фильтрация/очистка параметров по заранее определённой схеме

Примеры уязвимостей

XSS (Cross-site Scripting) – межсайтовый скриптинг, или JS-инъекции

Злоумышленник выполняет произвольный JavaScript код в браузере пользователя-жертвы

Защита:

- Строгая проверка кодировки
- Content Security Policy
- др.

Защита веб-приложений на уровне развёртывания

Меры защиты

- На уровне архитектуры и инфраструктуры
  - Изоляция компонентов (виртуализация, контейнеризация)
  - Внесение в инфраструктуру дополнительных компонентов, повышающих безопасность
- На уровне разработки
  - Выбор языков и фреймворков
  - Безопасная разработка
  - Менеджмент уязвимостей
- На уровне развёртывания и настройки
  - Пользователи и права
  - Внутрисистемный межсетевой экран (файрвол)
- На организационном уровне
  - Регламент для коммуникации, работы с документами, данными
  - Обучение сотрудников

Защита веб-приложений на уровне инфраструктуры

Защита путём внесения в инфраструктуру информационной системы дополнительных компонентов, повышающих защищённость основных/целевых/функциональных компонентов этой системы:

- Программные и аппаратные сетевые экраны
- WAF (Web Application Firewall)
- SIEM (Security Information and Event Management)

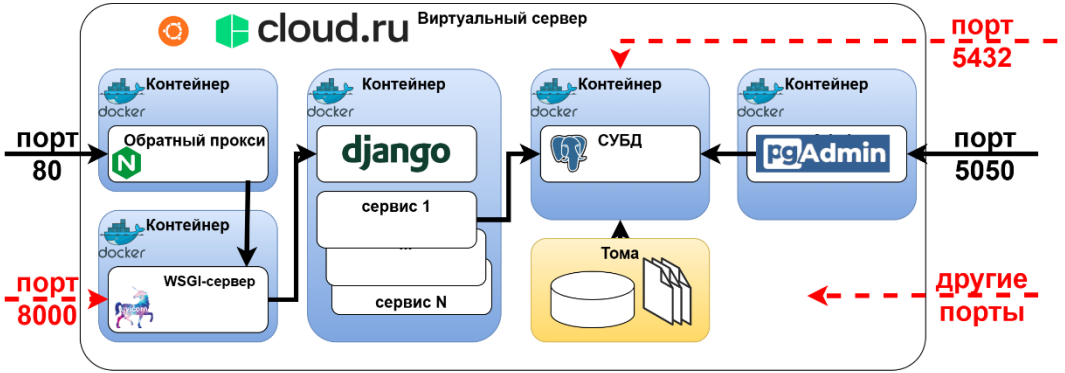
- Обзор WAF

Характеристика	ModSecurity	Coraza WAF	NAXSI	Cloudflare WAF
Фильтрация HTTP трафика	Да	Да	Да	Да
Защита от XSS и SQL инъекций	Да	Да	Да	Да
Доступность настройки	Высокая	Высокая	Средняя	Средняя
Интеграция с CDN	Нет	Да	Нет	Да
Тип лицензии	Apache-2.0	Apache-2.0	GPL	GPL
Стоимость лицензии	Бесплатно	Бесплатно	Бесплатно	Платно

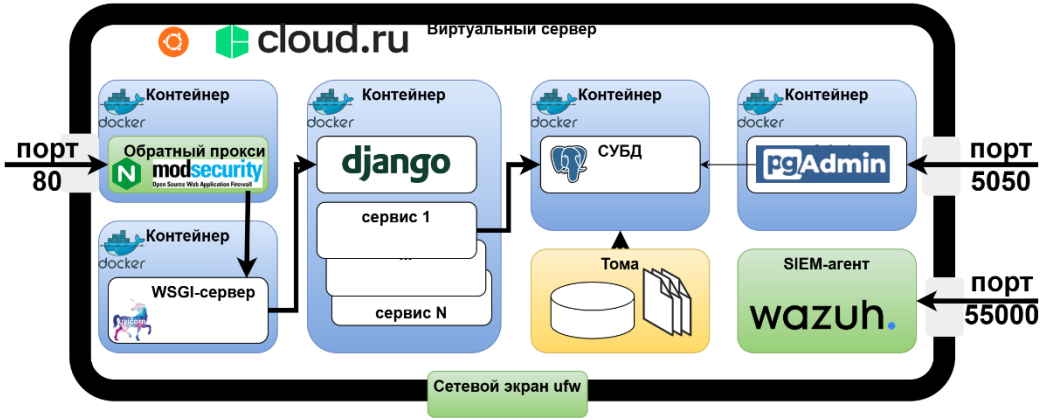
УЧЕБНЫЙ ЦЕНТР МАСКОМ

Защита веб-приложений

Незащищённое приложение

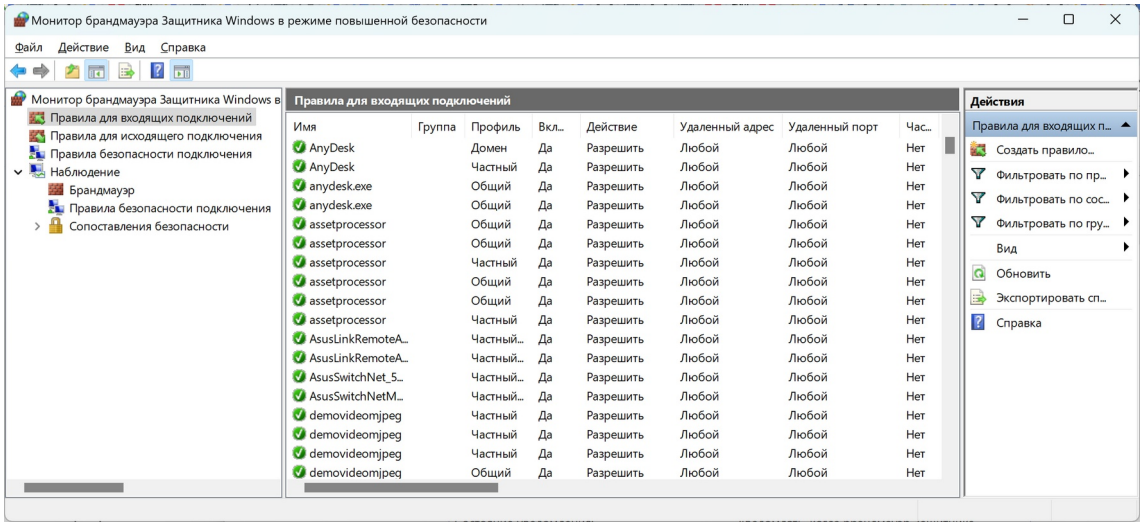


Элементы защиты в инфраструктуре



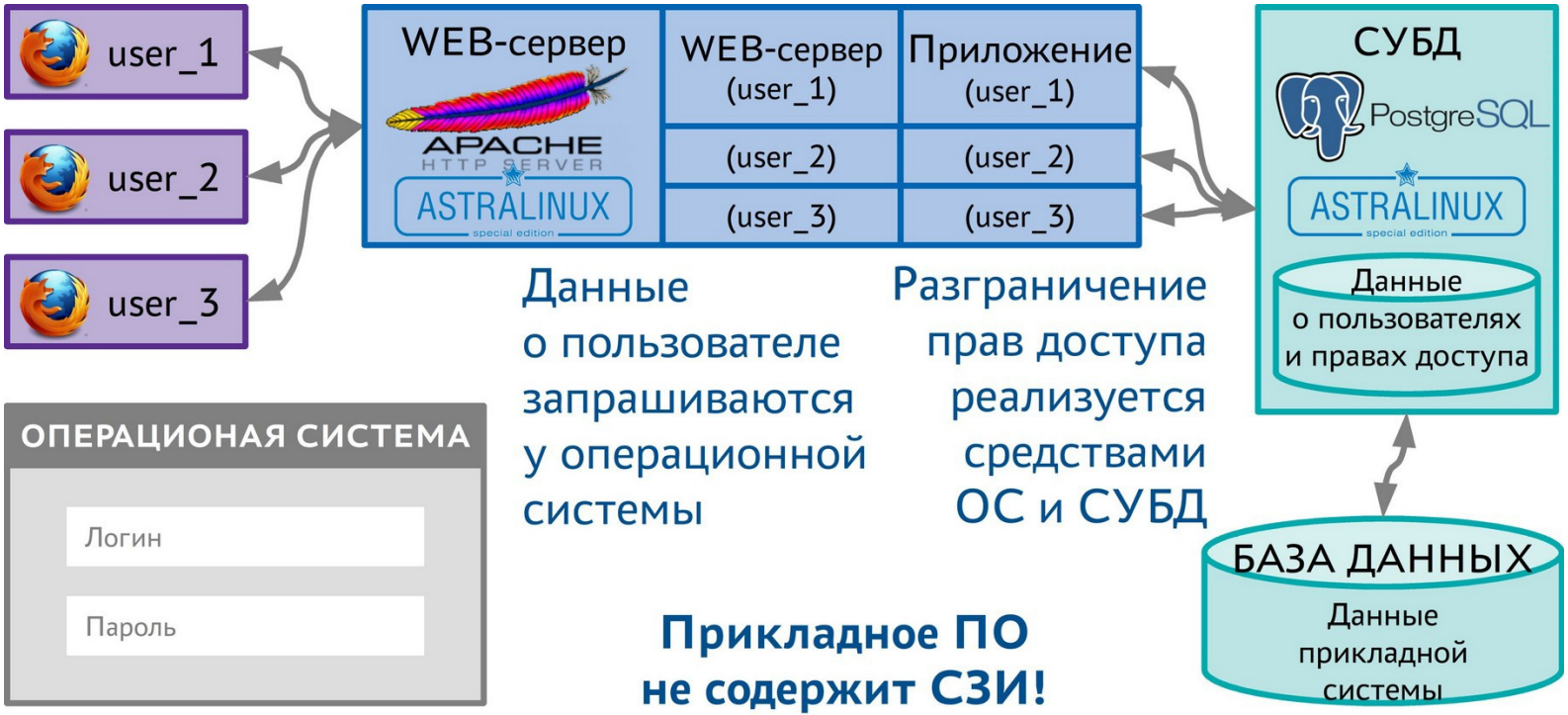
Пример: настройка сетевых экранов Linux и Windows

```
sudo apt install ufw
sudo ufw deny from 192.168.1.0/24
sudo ufw deny 1234/tcp
```



Защита веб-приложений

Пример: Astra Linux («Описание контрольного примера»)



Защита веб-приложений

- Защита от атак на цепочку поставки
  - Сторонние компоненты только с открытым исходным кодом
  - Аудит исходного кода сторонних компонентов
  - Сборка всех артефактов в минималистичных контейнерах на собственной инфраструктуре, исключение попадания сторонних бинарных артефактов

Тестирование на проникновение

ГОСТ Р 56939-2016, п. 3.16: **тестирование на проникновение**: Вид работ по выявлению (подтверждению) уязвимостей программы, основанный на моделировании (имитации) действий потенциального нарушителя.

Пентест (**PenTest**, англ. **penetration testing**) – процесс взлома или поиска уязвимостей ИС, санкционированный заказчиком

Несмотря на название, данный тип тестирования часто направлен на поиск любых уязвимостей

Тестирование на проникновение

Bug bounty – форма тестирования в виде соревнования, с регламентом, согласно которому

- заказчик объявляет награду за найденные уязвимости, наибольшее число уязвимостей
- участники должны раскрыть найденные уязвимости заказчику

Баг-хантер (англ. **bug hunter**) – часто используемое название участника таких соревнований. Также – white hat («этичный хакер»).

Также **Red Team:**

- команда, выполняющая аналогичную работу в ходе киберучений (**Blue Team** – те, кто им противостоит)
- подразделение компании; специально нанятые на длительной основе сотрудники, с большими полномочиями, чем у сторонних пентестеров ссылка

Тестирование на проникновение

Онлайн-ресурсы:

- Платформы для проведения Bug bounty:
  - https://www.hackerone.com/
  - https://bugbounty.ru/
  - https://bugbounty.standoff365.com
  - https://bugbounty.bi.zone/
- Корпоративные программы
  - https://yandex.com/bugbounty/index
  - https://bugbounty.vk.company.ru/
- Тренировочные площадки
  - https://www.hackthebox.com/
  - https://tryhackme.com/
  - https://www.vulnhub.com/
  - https://codeby.games/
  - https://range.standoff365.com/#battlesInProgress
  - https://www.root-me.org/
  - https://portswigger.net/web-security/all-labs

Тестирование на проникновение

Приложения для демонстрации уязвимостей и тренировки:

- OWASP Juice Shop ссылка1, ссылка2
- OWASP Mutillidae II ссылка 2025
- OWASP Broken Web Applications Project ссылка 2015
- OWASP Samurai Web Training Framework ссылка 2023
- OWASP VulnerableApp ссылка1, ссылка2
- OWASP Webgoat
- Metasploitable3 ссылка
- Damn Vulnerable Web Application (DVWA) – «чертовски уязвимое веб-приложение»
  - edgeNEXUS DVWA Ссылка (облачный образ, $0.375/hour)
  - Docker-образ ссылка, ссылка2, сайт разработчика, исходный код
- Buggy Web Application (BWAPP, 2014) ссылка, исходный код,
- Damn Vulnerable iOS App (DVIA)
- Web Security Dojo (WSD) ссылка 2020
- Damn Vulnerable Web Services ссылка 2016, ссылка

Специализированные дистрибутивы

Название	Ссылка	Примечание
Дистрибутив Kali Linux	https://www.kali.org/	Наиболее известный специализированный дистрибутив Linux для пентеста
OWASP Samurai Web Training Framework	https://www.samurai-wtf.org/	Специализированный дистрибутив Linux, включает уязвимые приложения для тренировки
Parrot OS	https://www.parrotsec.org/	Специализированный дистрибутив Linux
BlackArch Linux	https://blackarch.org/	Специализированный дистрибутив Linux
Pentoo	https://www.pentoo.ch/	Специализированный дистрибутив Linux
Network Security Toolkit	https://www.networksecuritytoolkit.org/	

Инструменты пентеста

- Отдельные инструменты тестирования

Название
Burp Suite
John the ripper
Nmap
OWASP ZAP

Инструменты пентеста

Ещё инструменты (https://kali.tools/?p=1744), Web Application Attack Tool (https://www.edgenexus.io/ru/dvwa/)

looxy.io

w3af

sqlmap

arachni

OWASP Skavenger

OWASP Dirbuster

Paros

Webscarab

Ratproxy

skipfish

websecurify

davtest

J-Baah

JBroFuzz

Watobo

RATS

Инструменты пентеста

- [практический пример пентеста – уязвимое приложение + сканер]
  - Metasploitable2 https://hub.docker.com/r/kirscht/metasploitable3-ub1404, https://blog.securelayer7.net/metasploitable-3-walkthrough-part-2/
  - Подборка https://medium.com/@muchina/what-is-docker-2a21077b98df
  - Juice Shop https://dev.to/hax/owasp-juice-shop-set-up-with-docker-2php

Сканеры уязвимостей

Выявление уязвимостей:

- **Пентест** – тестирование методом «чёрного» или «серого» ящика, чаще всего - динамическое
- **Сканирование уязвимостей** – тестирование методом «белого» или «серого» ящика, чаще всего - статическое

Сделай свой проект чистым
и безопасным вместе
с PVS-Studio

VOKRUG_RBPO25

PVS-Studio

Получи 10% скидку
на курсы «М БРПО»
в Учебном Центре
«МАСКОМ»

VOKRUG_RBPO25

MACKOM
УЧЕБНЫЙ ЦЕНТР