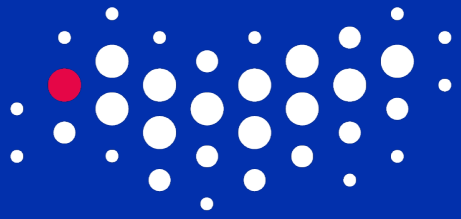


УНИВЕРСИТЕТ ИТМО

Разработка автоматизированной системы аудита веб-приложений на наличие уязвимостей OWASP TOP 10

Безручко Ярослав
Кафедра БИТ, 4 курс

Санкт-Петербург, 2017

Актуальность

- Постоянное увеличение количества веб-приложений
- Недостаточная осведомленность пользователей в вопросах ИБ
- Рост числа преступлений в сети интернет
- Отсутствие сервиса, осуществляющего аудит веб-приложений, простого в использовании для пользователей с начальным уровнем подготовки

Практическая значимость

- Автоматизация аудита веб-приложения на наличие уязвимостей
- Максимальная простота использования

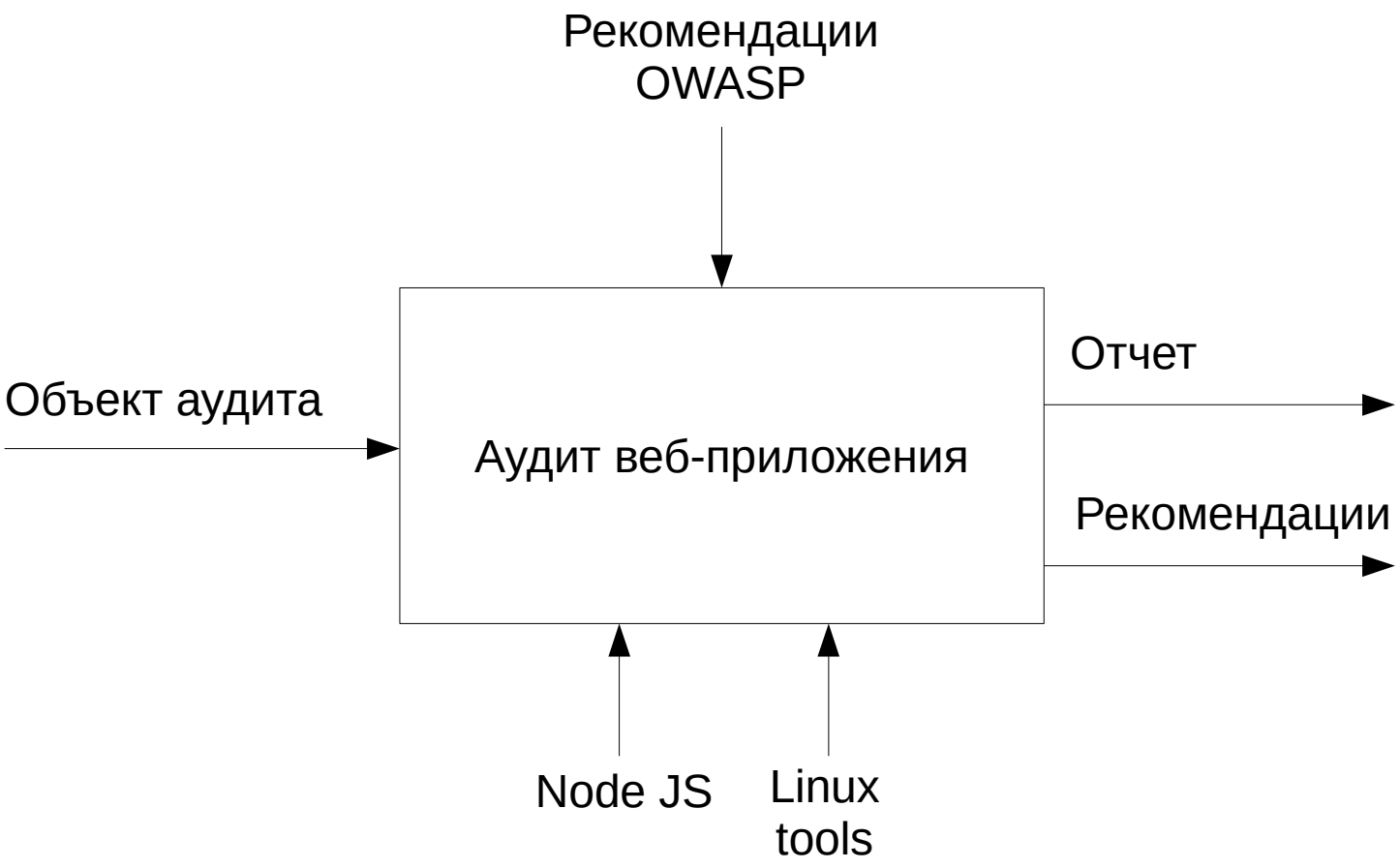
Обзор существующих решений

Tool	GUI	Price	Online-version	Ease of use (Count of acts)	Open Sourse	Engine
Acunetix	+	2500-10000 $/yr	+	>10	-	?
Arachni	-	free	-	5	+	Ruby
Burp Suite	+	350$/yr	-	>10	-	?
OWASP ZAP	+	free	-	7	+	Java
SkipFish	-	free	-	6	+	C
Vega	+	free	-	8	+	Java
W3AF	-	free	-	>10	+	Python

Цель

- Разработка автоматизированной системы аудита веб-приложений на наличие уязвимостей, которая проста в использовании для пользователей с начальным уровнем подготовки

Схема работы системы



Функции разрабатываемой системы

1. Сбор данных о веб-приложении
2. Проверка на соответствие рекомендациям OWASP
3. Анализ на наличие уязвимостей
4. Составление отчета
5. Формирование рекомендаций по повышению уровня безопасности

Сбор данных

Функция	Механизм
Составление карты сайта	Скрипт-краулер, реализованный с помощью утилит wget и sed
Сбор информации о DNS	Утилита nslookup
Сканирование сети	Утилита nmap
Сбор информации о владельце сайта и регистраторе доменного имени	Утилита whois
Получение содержания файлов robots.txt и sitemap.xml	Утилита curl
Получение http-заголовков	Http-запрос, реализованный на node JS

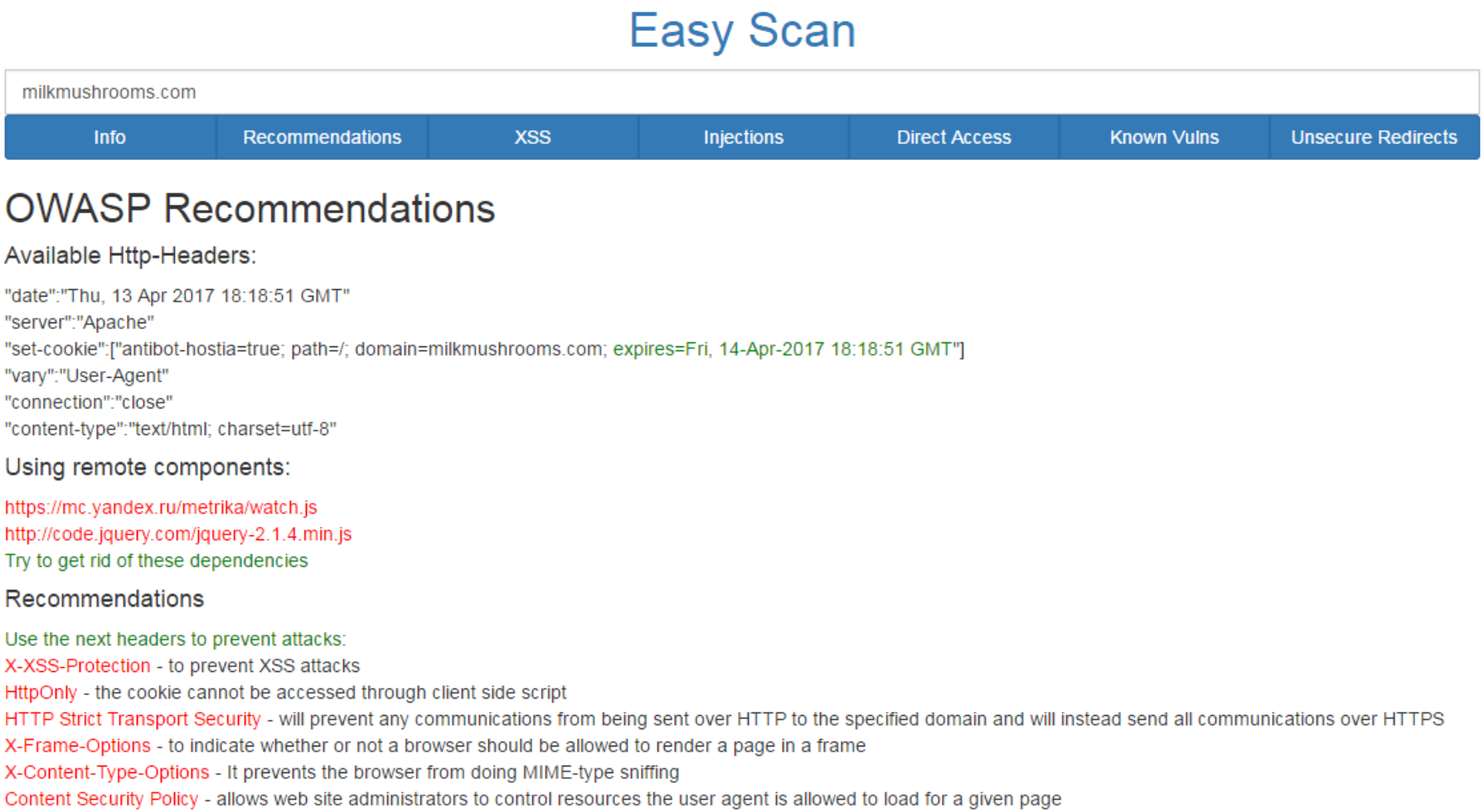
Проверка на соответствие рекомендациям OWASP

1. Проверка факта использования соответствующих http-заголовков для предотвращения XSS-атак, кликджекинга и внедрения кода
2. Проверка факта использования http-заголовков для защиты критичных данных и предупреждения подмены MIME типов
3. Проверка факта установки времени жизни сессии
4. Проверка факта отсутствия включения внешних объектов
5. Проверка факта отсутствия кэширования страниц с пользовательскими данными

Анализ веб-приложения на наличие уязвимостей

Функция	Механизм
Проверка наличия XSS-уязвимости	Node JS
Проверка наличия SQL-инъекции	Node JS + SQLmap
Проверка факта использования уязвимых компонентов	WPScan, JoomScan, DroopeScan
Проверка осуществимости небезопасного доступа к объектам	Gobuster, google dorks
Проверка наличия небезопасных перенаправлений	Node JS

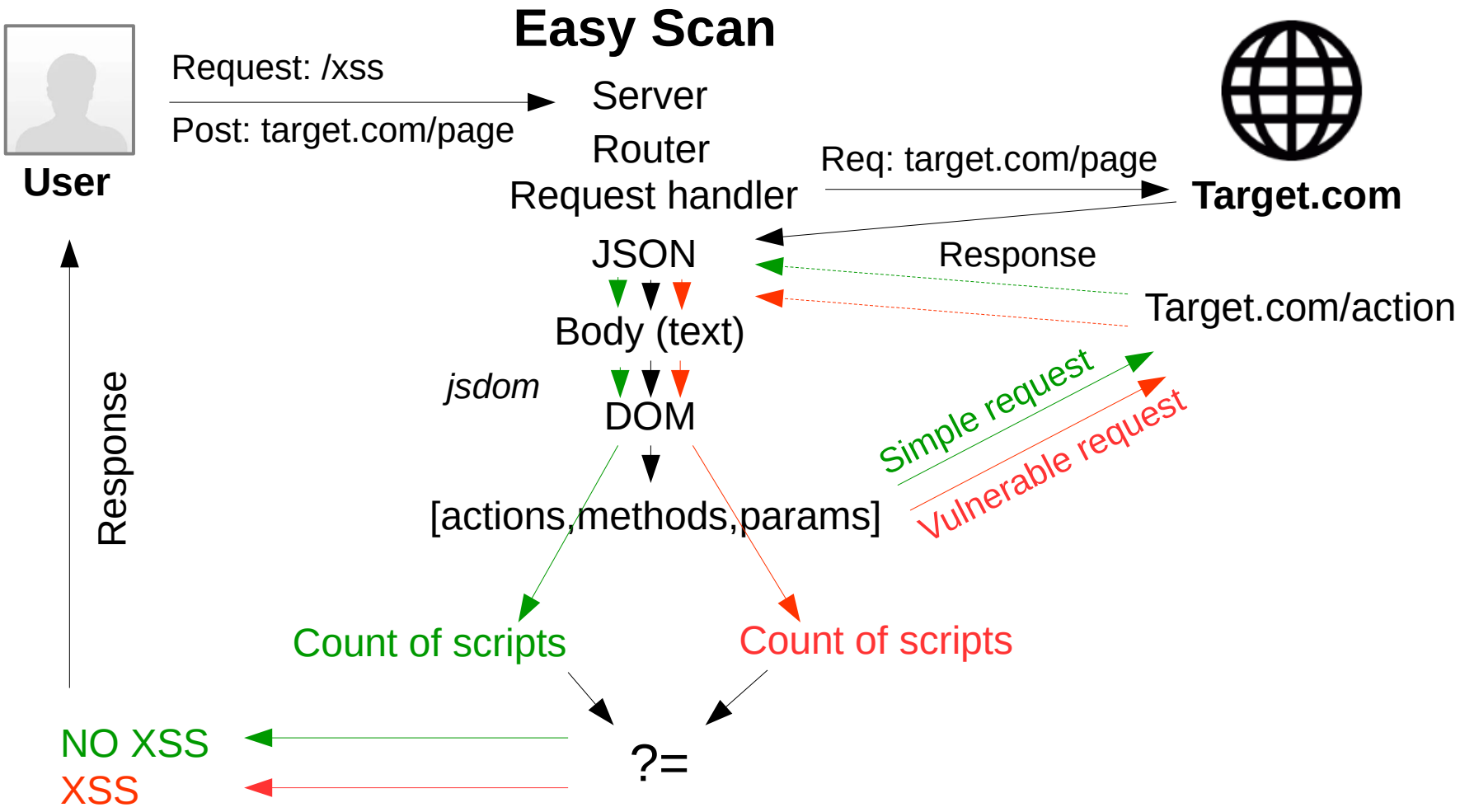
Клиентская часть



Серверная часть

```
root@kali:~/Documents/EasyScan# ls
index.js  node_modules  requestHandlers.js  router.js  server.js  static  tools
root@kali:~/Documents/EasyScan# ls tools/
checkHeaders.sh  getparams.sh  init.sh       run.sh
getinfo.sh       gobuster.sh   knownvuls.sh  wordlists
root@kali:~/Documents/EasyScan# nodejs index.js
Server has started at 8888, '192.168.1.38'.
Request for /info received.
Received POST data chunk 'target=milkmushrooms.com'.
About to route a request for /info
Request for /recomends received.
Received POST data chunk 'target=milkmushrooms.com'.
About to route a request for /recomends
{"date":"Thu, 13 Apr 2017 19:08:37 GMT","server":"Apache","set-cookie":["antibot
-hostia=true; path=/; domain=milkmushrooms.com; expires=Fri, 14-Apr-2017 19:08:3
7 GMT"],"vary":"User-Agent","connection":"close","content-type":"text/html; char
set=utf-8"}
Request for /xss received.
Received POST data chunk 'target=milkmushrooms.com'.
About to route a request for /xss
Document { location: [Getter/Setter] }
[ [ 'http://milkmushrooms.com/scripts/message.php' ],
  [ 'get' ],
  [ [ 'name', 'email', 'msg' ] ] ]
Length of Actions array: 1
```

Схема анализа наличия XSS-уязвимости



Демонстрация работы сервиса

Easy Scan

http://glassbank.ifmo.ru/rus/index.php

Info | Recommendations | XSS | Injections | Direct Access | Known Vulns | Unsecure Redirects

Cross Site Scripting (XSS)

Form handlers:

http://glassbank.ifmo.ru/rus/search.php

Methods:

get

Parametrs:

text,submit

This page IS VULNERABLE for XSS!

Try the following vulnerable links to get sure:

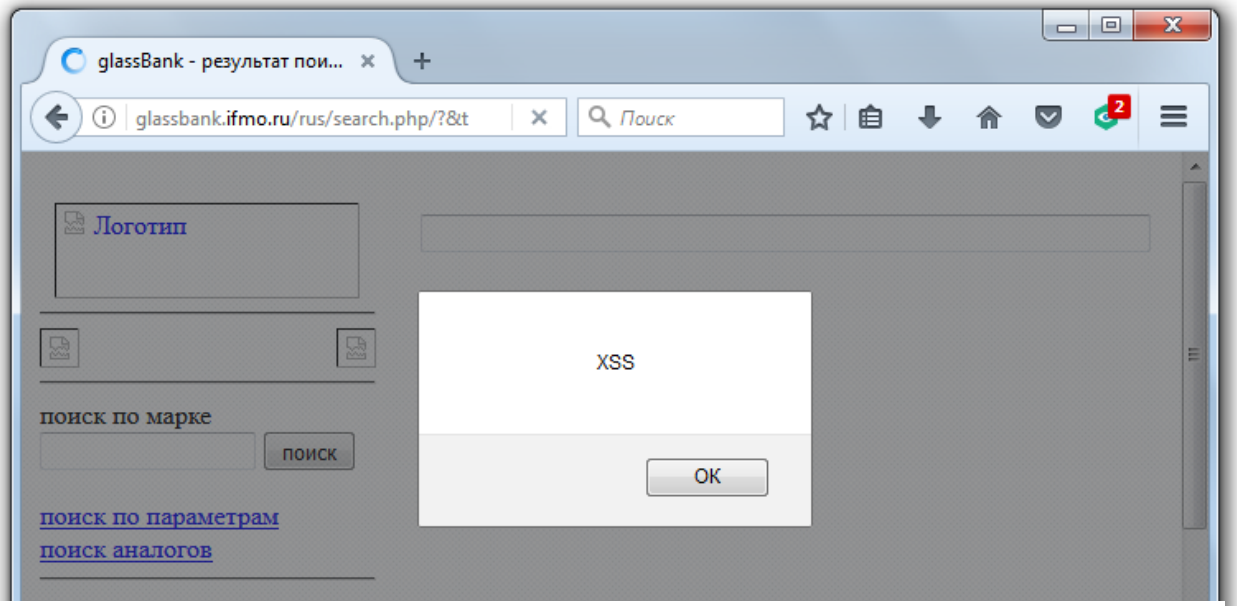
Attack with GET method

Vector 0
Vector 1
Vector 2
Vector 3

OWASP Recommendations:

Preventing XSS requires separation of untrusted data

- The preferred option is to properly escape all ur
  See the OWASP XSS Prevention Cheat Sheet fo



Здравствуйте. Сообщаю об уязвимости сайта glassbank.ifmo.ru к XSS атакам.
Пример страницы со встроенным кодом: http://glassbank.ifmo.ru/rus/search.php?text=">
<script>alert("XSS")</script>
Как видно по ссылке, уязвим параметр "text".

Выводы

- Разработанная система позволяет проводить аудит веб-приложений на наличие уязвимостей
- Система проста в использовании: требуется только указать объект аудита
- Онлайн-версия: easy-scan-prifki.c9users.io
- Открытый исходный код: github.com/Prifki/EasyScan

УНИВЕРСИТЕТ ИТМО

Спасибо за внимание!

Санкт-Петербург, 2017