

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Информационная безопасность

Работа 4

Анализ уязвимостей веб-приложения с помощью OWASP ZAP

Группа: Р3416

Выполнил:

Сиразетдинов Азат Ниязович

г. Санкт-Петербург

2025

Выполнение

Выполняем сканирование с помощью OWASP ZAP. Для сканирования был выбран сайт <http://testphp.vulnweb.com/>

Идентификатор	Источник	Время Ответа времени	Метод	URL	Тип
32	Прокси-серв...	08.10.2025, 00:23:21	GET	http://	Form, Password, Objec...
33	Прокси-серв...	08.10.2025, 00:23:29	POST	http://	Comment
35	Прокси-серв...	08.10.2025, 00:23:29	GET	http://	Form, Password, Objec...
36	Прокси-серв...	08.10.2025, 00:23:33	GET	http://	Form, Password, Objec...
37	Прокси-серв...	08.10.2025, 00:23:37	POST	http://	Form, Password, Objec...
38	Прокси-серв...	08.10.2025, 00:23:38	GET	http://	Form, Password, Objec...
40	Прокси-серв...	08.10.2025, 00:23:42	POST	http://	Form, Password, Objec...
41	Прокси-серв...	08.10.2025, 00:23:42	GET	http://testphp.vulnweb.com/login.php	5 523 байт
42	Прокси-серв...	08.10.2025, 00:23:47	POST	http://testphp.vulnweb.com/userinfo.php	327 мс
44	Прокси-серв...	08.10.2025, 00:23:48	GET	http://testphp.vulnweb.com/hacked	301 Moved Permanen...
46	Прокси-серв...	08.10.2025, 00:23:49	GET	https://www.kalkikravdna.com/hacked	769 мс
57	Прокси-серв...	08.10.2025, 00:23:50	GET	https://www.kalkikravdna.com/cdn-cgi/challenge...	302 Found
62	Прокси-серв...	08.10.2025, 00:23:50	GET	https://www.kalkikravdna.com/cdn-cgi/challenge...	200 OK
63	Прокси-серв...	08.10.2025, 00:23:50	GET	https://www.kalkikravdna.com/kalki.php	200 OK
69	Прокси-серв...	08.10.2025, 00:23:50	POST	https://www.kalkikravdna.com/cdn-cgi/challenge...	200 OK
75	Прокси-серв...	08.10.2025, 00:23:57	GET	http://testphp.vulnweb.com/car.php	58 мс
75	Прокси-серв...	08.10.2025, 00:23:58	GET	http://testphp.vulnweb.com/questbook.php	183 мс
77	Прокси-серв...	08.10.2025, 00:23:59	GET	http://testphp.vulnweb.com/AJAX/index.php	164 мс
79	Прокси-серв...	08.10.2025, 00:23:59	GET	http://testphp.vulnweb.com/AJAX/styles.css	162 мс
80	Прокси-серв...	08.10.2025, 00:24:01	GET	http://testphp.vulnweb.com/artists.php	4 236 байт
81	Прокси-серв...	08.10.2025, 00:24:03	GET	http://testphp.vulnweb.com/index.php	200 OK
					320 мс
					164 мс
					323 мс
					5 402 байт
					5 032 байт

В процессе сканирования были выявлены уязвимости: 1 high, 8 medium, 10 low и 7 informational

Тип	Количество
SQL-инъекция – MySQL	5
CSP: script-src unsafe-eval	2
CSP: style-src небезопасный встроенный	2
CSP: директива подстановочного знака	2
CSP: скрипт-SRC небезопасный встроенный	2
Заголовок Content Security Policy (CSP) не задан	17
Межхостовая неправильная конфигурация	3
Отсутствует заголовок (Header) для защиты от кликджекинга	13
Отсутствуют токены против CSRF атак	15
Cookie No HttpOnly Flag	2
Cookie без атрибута SameSite	2
Cookie без флагка безопасности	
Cookie с атрибутом SameSite нет	
Заголовок Strict-Transport-Security не установлен	3
Заголовок X-Content-Type-Options отсутствует	16
Множественные записи заголовков Strict-Transport-Security (несовместимы)	
Раскрытие отметки времени – Unix	2
Сервер утекает информацию через поля заголовка HTTP-ответа "X-PowerBy"	
Сервер утечка информации о версии через поле заголовка HTTP-ответа «\$	
Authentication Request Identified	
Cookie с произвольным ограничением	
Session Management Response Identified	3
Атрибут элемента HTML, управляемый пользователем (потенциальный XSS)	
Несоответствие кодировки (Заголовок по сравнению с кодировкой мета-)	
Пересмотрите директивы управления кешем	
Современное веб-приложение	4

Уязвимость 1: SQL инъекция

Уровень риска

Высокий(High). Количество экземпляров – 5

Причина

Пользовательский ввод попадает в SQL запрос без экранирования. Злоумышленник может удалять/изменять/получать данные из таблицы используя фразы SQL.

Обнаружение

Чтобы обнаружить наличие SQL инъекции можно передать невалидный SQL скрипт. Одним из вариантов является одинарная кавычка. SQL вернет ошибку, что кавычка не имеет закрывающую кавычку

Пример

SQL-инъекция – MySQL

URL-адрес: <http://testphp.vulnweb.com/artists.php?artist=%27>

Риск: High

Достоверность: Medium

Параметр: artist

Атака: '

Доказательства: You have an error in your SQL syntax

CWE ID: 89

WASC ID: 19

Источник: Активная (40018 – SQL-инъекция)

Input Vector: URL Query String

Описание: SQL injection may be possible.

Дополнительно:
RDBMS [MySQL] likely, given error message regular expression [^QYou have an error in your SQL syntax\] matched by the HTML results.
The vulnerability was detected by manipulating the parameter to cause a database error message to be returned and recognised.

Решение:
Do not trust client side input, even if there is client side validation in place.
In general, type check all data on the server side.
If the application uses JDBC, use PreparedStatement or CallableStatement, with parameters passed by ?

Ссылка:
https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

При запросе <http://testphp.vulnweb.com/artists.php?artist=%27> параметр artist не экранируется при запросе в базу данных, вследствие чего одинарная кавычка вызывает ошибку SQL синтаксиса

Уязвимость 2: XSLT Инъекция

Уровень риска

Средний (Medium). Количество экземпляров – 2

Причина

Текстовый ввод попадает в XSLT-процессор без дальнейшего форматирования. Это позволяет злоумышленнику добавлять XSL трансформации и получить содержимое системных файлов, делать запросы на внутренние порты, вызывать удаленное исполнение кода

Обнаружение

Для обнаружения XSLT инъекции можно передать невалидную XSL трансформацию. Например, двойные кавычки, одинарные кавычки и угловые скобки

Пример

HTTP/1.1 200 OK
Server: nginx/1.19.0
Date: Tue, 07 Oct 2025 22:03:31 GMT
Content-Type: image/jpeg
Connection: keep-alive
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
Content-Length: 286

Warning: fopen(<xsl:value-of select="document('http://testphp.vulnweb.com:22')"/>): failed to open stream: No such file or directory in /hj/var/www/showimage.php on line 13
Warning: fpassthru() expects parameter 1 to be resource, boolean given in /hj/var/www/showimage.php on line 19

XSLT Инъекция

- URL-адрес: <http://testphp.vulnweb.com/showimage.php?file=%3Cxsl%3Avalue-of+select%3D%22document%28%27http%3A%2Ftestphp.vulnweb.com%3A22%27%29%22%2F%3E>
- Риск: Medium
- Достоверность: Medium
- Параметр: file
- Атака: <xsl:value-of select="document('http://testphp.vulnweb.com:22')"/>
- Доказательства: failed to open stream
- CWE ID: 91
- WASC ID: 23
- Источник: Активная (90017 – XSLT Инъекция)
- Input Vector: URL Query String
- Описание:
Внедрение с использованием преобразований XSL может быть возможным и может позволить злоумышленнику читать системную информацию, читать и записывать файлы или выполнять произвольный код.
- Дополнительно:
Возможно сканирование портов.

При запросе <http://testphp.vulnweb.com/showimage.php?file=%3Cxsl%3Avalue-of+select%3D%22document%28%27http%3A%2F%2Ftestphp.vulnweb.com%3A22%27%29%22%2F%3E> мы передаем XSL трансформацию

<xsl:value-of select="document('http://testphp.vulnweb.com:22')"/> что вызывает ошибку XSLT-процессора

Уязвимость 3: Отсутствует заголовок (Header) для защиты от кликджекинга

Уровень риска

Средний (Medium). Количество экземпляров – 21

Причина

Сайт не защищен от кликджекинга. Это позволяет злоумышленнику использовать невидимые iframe для совершения действий на уязвимом сайте

Обнаружение

Для обнаружения нужно проверить наличие заголовка X-Frame-Options (позволит ограничить появление сайта в iframe) или Set-Cookie (запретит использование cookie файлов на других доменах)

Пример

The screenshot shows a web-based security scanner interface. At the top, there are dropdown menus for 'Заголовок: Текст' (Header: Text) and 'Тело: Текст' (Body: Text), followed by two small icons. Below these are two large code snippets. The first snippet is the raw HTTP response header:

```
HTTP/1.1 200 OK
Server: nginx/1.19.0
Date: Tue, 07 Oct 2025 21:22:43 GMT
Content-Type: text/html; charset=UTF-8
Connection: keep-alive
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
content-length: 4958
```

The second snippet is the HTML source code of the page:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html><!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php" codeOutsideHTMLIsLocked="false" -->
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
<!-- InstanceBeginEditable name="document_title_rgn" -->
<title>Home of Acunetix Art</title>
<!-- InstanceEndEditable -->
```

Below the code snippets is a section titled 'Сканирование' (Scanning) with a '+' button. It contains the following information:

Отсутствует заголовок (Header) для защиты от кликджекинга

URL-адрес: <http://testphp.vulnweb.com/>

Риск: Medium

Достоверность: Medium

Параметр: x-frame-options

Атака:

Доказательства:

CWE ID: 1021

WASC ID: 15

Источник: Пассивный (10020 – Заголовок против кликджекинга)

Alert Reference: 10020-1

Input Vector:

Описание:

The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.

Дополнительно:

При запросе на <http://testphp.vulnweb.com/> отсутствуют заголовки, защищающие от кликджекинга