**1. Введение.**

Цель данного анализа - симуляция атаки потенциального злоумышленника на веб-приложение, оценка уровня его защищенности, обнаружение уязвимостей, анализ и разработка рекомендаций по их устранению.

**2 Объект тестирования.**

Объектом тестирования является веб-приложение OWASP Juice Shop (OWASP-juice.shop). Приложение представляет из себя интернет магазин по продаже сока и атрибутики.

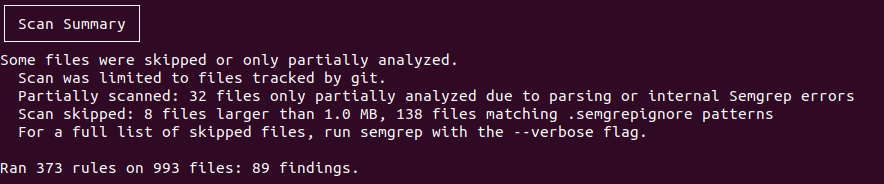
**2 Статический анализ.**

В качестве первого этапа тестирования был проведен статический анализ исходного кода приложения. Анализ производился с использованием инструмента Semgrep.

git clone <https://github.com/juice-shop/juice-shop.git>

cd ./juice-shop

semgrep scan --json-output=semgrep.json –-config “p/default”



К отчету приложен полный журнал сканирования **semgrep.sarif**

**3. Обнаруженные уязвимости.**

В результате статического анализа исходного кода приложения были обнаружены следующие уязвимости.

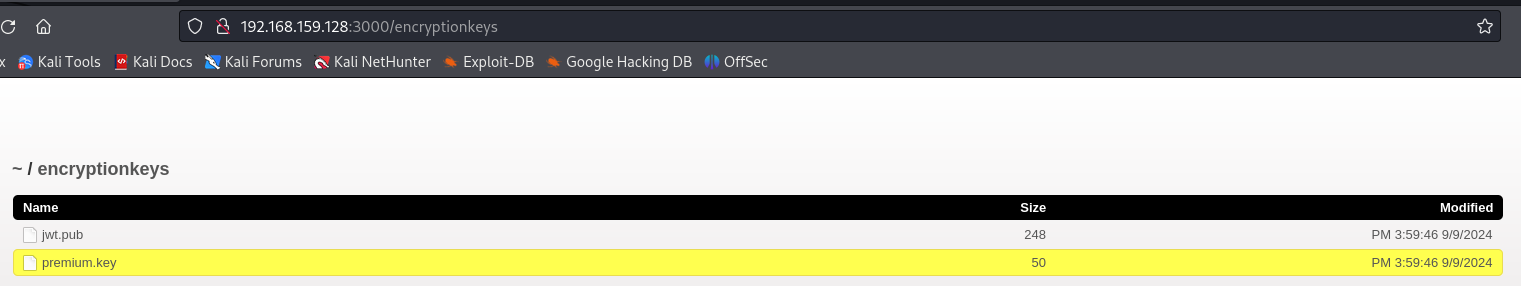
|  |  |  |
| --- | --- | --- |
| **Тип** | **Описание** | **Ссылка на классификацию** |
| **Injection** | Приложение генерирует SQL-запрос на основе пользовательских данных без соответствующего санитайзинга.  В случае эксплуатации данная уязвимость может привести к утечке данных.  Приложение не обеспечивает достаточный санитайзинг пользовательского ввода, который в последующем используется в рендеринге веб-страницы. Это может привести к возникновению уязвимости типа DOM XSS, проэксплуатировав которую злоумышленник сможет выполнить на веб-странице произвольный скрипт. | CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')  <https://cwe.mitre.org/data/definitions/89.html>  CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')  <https://cwe.mitre.org/data/definitions/79.html>  OWASP-A03:2021 – Injection  <https://owasp.org/Top10/A03_2021-Injection/> |
| **Broken Access Control** | В приложении активирован листинг директорий, в том числе тех, что содержат чувствительные данные.    Приложение не обеспечивает корректную авторизацию пользовательских запросов, что может позволить злоумышленнику получить доступ к данным другому пользователя, используя уязвимость типа IDOR | CWE-548: Exposure of Information Through Directory Listing  <https://cwe.mitre.org/data/definitions/548.html>  CWE-639: Authorization Bypass Through User-Controlled Key  <https://cwe.mitre.org/data/definitions/639.html>  OWASP-A01:2021 - Broken Access Control  <https://owasp.org/Top10/A01_2021-Broken_Access_Control/> |
| Identification and Authentication Failures | Несколько “секретов” хранятся в исходном коде приложения. Данный подход может привести к утечке чувствительных данных, которыми сможет воспользоваться внутренний или внешний злоумышленник. | CWE-798: Use of Hard-coded Credentials  <https://cwe.mitre.org/data/definitions/798.html>  OWASP-A07:2021 - Identification and Authentication Failures  <https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/> |
| Server-Side Request Forgery | Приложение генерируют путь до файлов из потенциально недоверенных данных, что может привести к уязвимости типа path traversal. Злоумышленник может проэксплуатировать данную уязвимости, чтобы получить доступ к данным файлам. В случае недостаточной валидации и санитайзинга злоумышленник также может получить возможность изменять или создавать файлы. | CWE-918: Server-Side Request Forgery (SSRF)  <https://cwe.mitre.org/data/definitions/918.html>  OWASP-A10:2021 - Server-Side Request Forgery (SSRF)  <https://owasp.org/Top10/A10_2021-Server-Side_Request_Forgery_%28SSRF%29/> |
| Software and Data Integrity Failures | Недоверенный пользовательский ввод может быть использован для создания HTTP запроса, который приведет к возникновению уязвимости типа SSRF. Эксплуатация данной уязвимости позволит злоумышленнику произвольный запрос от сервера к другим внутренним или внешним системам, что может привести к несанкционированному доступу к чувствительным данным. | CWE-502: Deserialization of Untrusted Data  <https://cwe.mitre.org/data/definitions/502.html>  OWASP-A08:2021 - Software and Data Integrity Failures  <https://owasp.org/Top10/A08_2021-Software_and_Data_Integrity_Failures/> |

**4. Демонстрация эксплуатации уязвимости.**

В данном разделе представлены процесс и результаты эксплуатации обнаруженных уязвимостей.

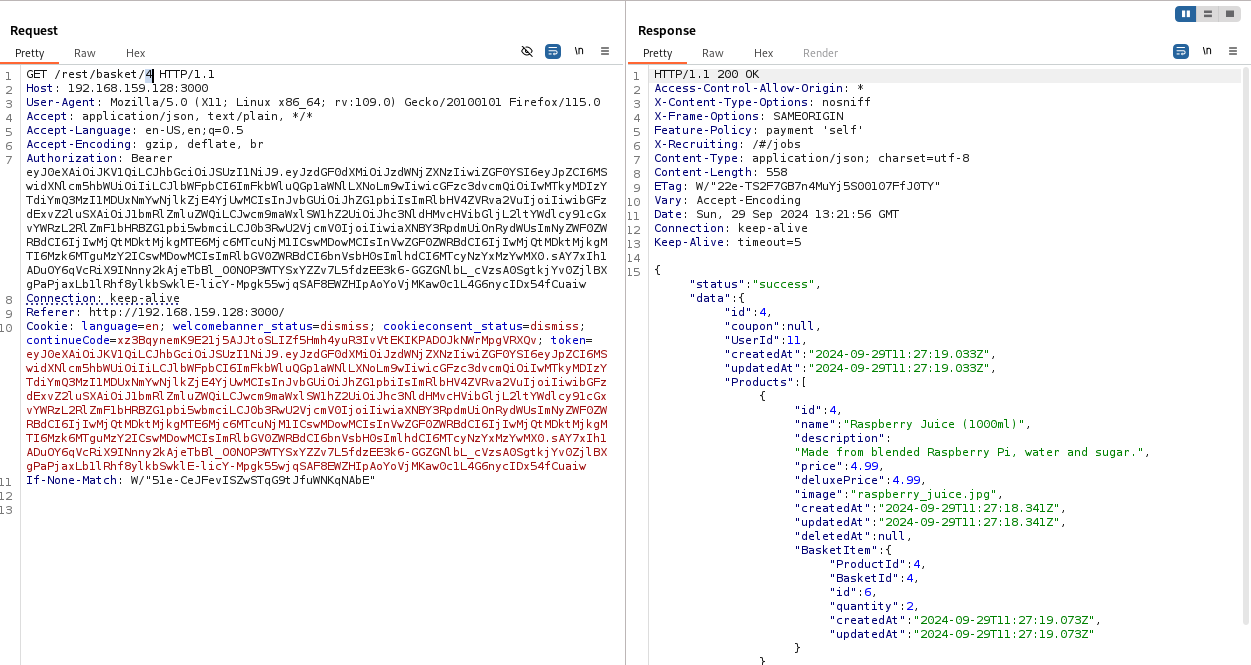
**Broken Access Control (directory listing)**

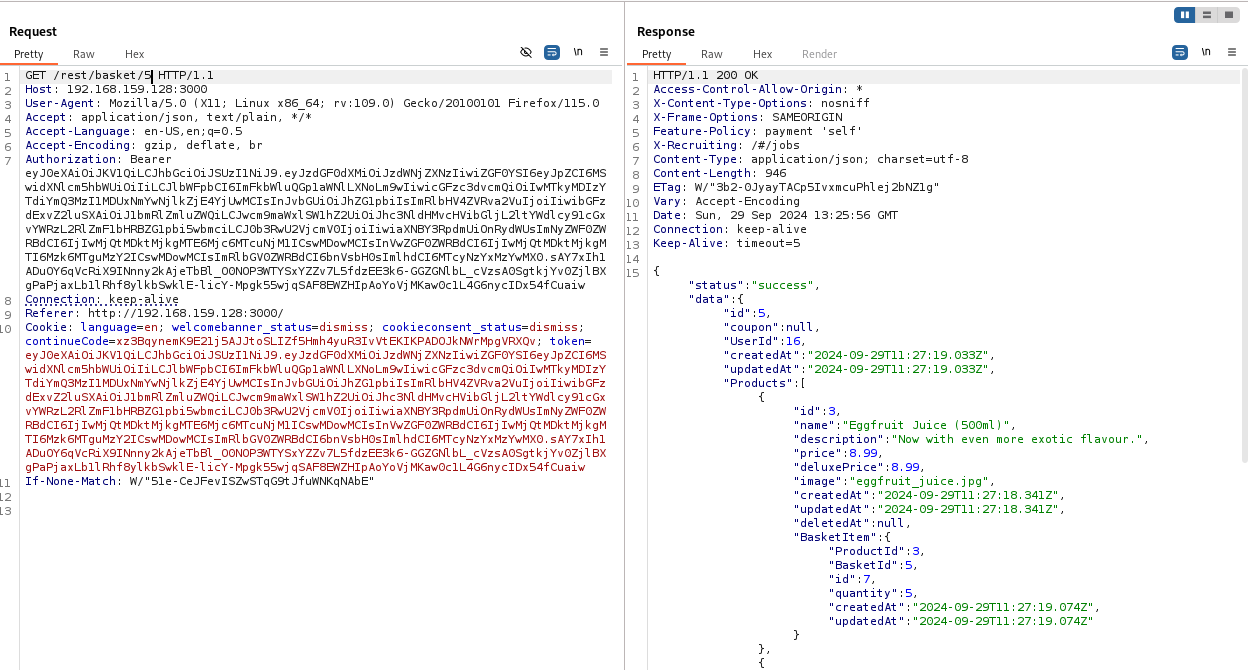
Обратившись напрямую к директории /encryptionkeys, мы можем скачать закрытый ключ администратора.



**Broken Access Control (IDOR)**

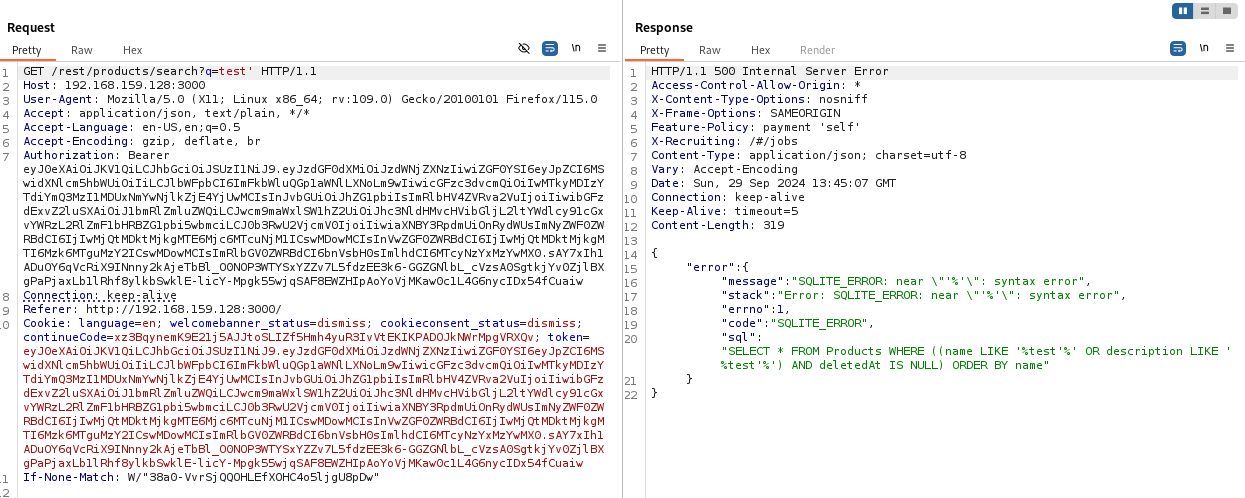
Путем подмены индекса корзины в GET запросе пользователь может получить информацию о содержимом корзины других пользователей.





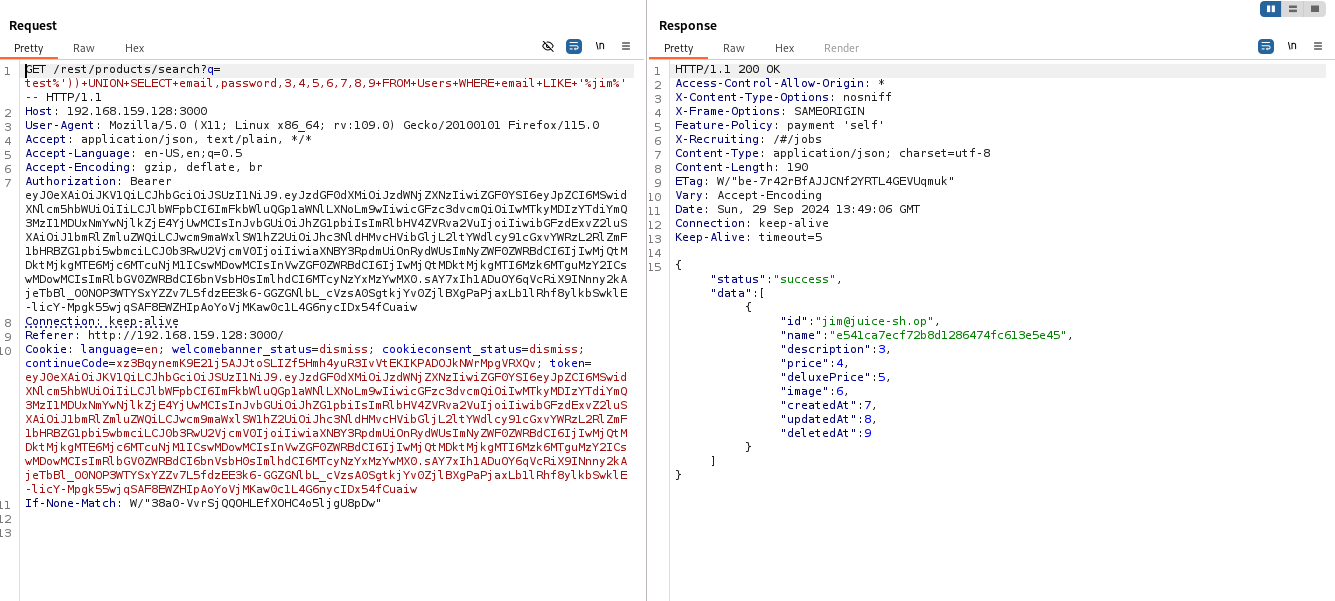
**Injection (SQL)**

Воспользуемся инструментом поиска на сайте. Для начала введем не валидные данные.



Теперь, зная структуру SQL-запроса, получим аутентификационные данные пользователя Jim.

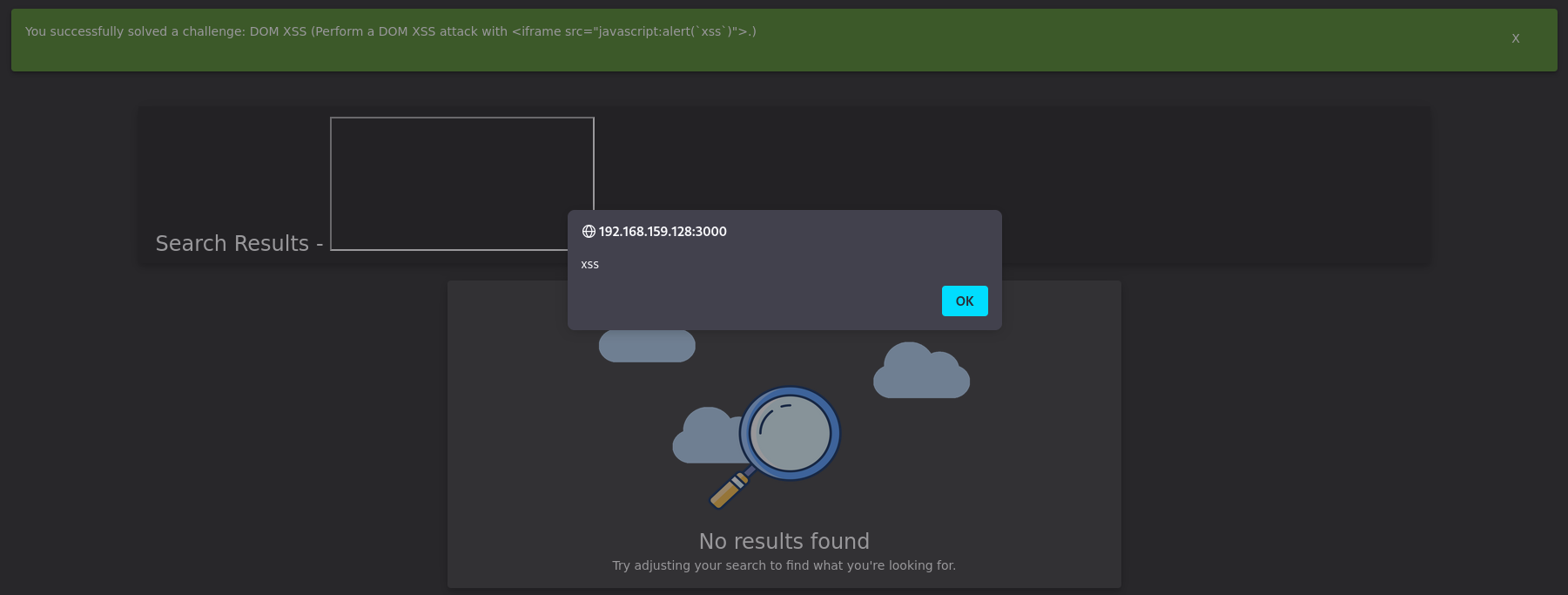
Запрос: **GET /rest/products/search?q=test%'))+UNION+SELECT+email,password,3,4,5,6,7,8,9+FROM+Users+WHERE+email+LIKE+'%jim%'--**



**Injection (DOM XSS)**

Воспользуемся инструментом поиска на сайте и в качестве входных данных используем скрипт. Как видно их скриншотов, скрипт выполняется.

**<iframe src="javascript:alert(`xss`)">**



**<iframe width="100%" height="166" scrolling="no" frameborder="no" allow="autoplay" src="https://w.soundcloud.com/player/?url=https%3A//api.soundcloud.com/tracks/771984076&color=%23ff5500&auto\_play=true&hide\_related=false&show\_comments=true&show\_user=true&show\_reposts=false&show\_teaser=true"></iframe>**



**5. Рекомендации по устранению уязвимостей.**

В данном разделе представлены рекомендации по устранению обнаруженных уязвимостей.

|  |  |
| --- | --- |
| **Тип** | **Рекомендация** |
| CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') | Рассмотрите возможность использования объектно-реляционных (ORM) фреймворков для работы с более безопасными абстракциями.  Используйте параметризованные запросы, в которых команды SQL и пользовательские данные строго разделены. |
| CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') | Экранируйте и валидируйте пользовательский ввод, прежде чем внедрять его в HTML-страницу.  Старайтесь избегать использования недоверенных данных в скриптах и обработчиках событий. Если это необходимо, используйте лучшие практики: <https://cheatsheetseries.owasp.org/cheatsheets/DOM_based_XSS_Prevention_Cheat_Sheet.html> |
| CWE-548: Exposure of Information Through Directory Listing | Используете ограниченный доступ к важным директориям и файлам.  Отключайте такие функции, как Автоматический Листинг Директорий. |
| CWE-639: Authorization Bypass Through User-Controlled Key | Избегайте использования идентификаторов в URL и POST-запросах. Вместо этого получайте информацию о пользователе из данных активной сессии.  Применяйте проверку контроля доступа для объектов, которым пользователь пытается получить доступ. |