**Bezpieczeństwo Aplikacji Webowych**

**OWASP Juice Shop**

Raport

Celem zadania jest znalezienie oraz exploitacja jak największej liczby podatności. W aplikacji znajdują się wyzwania, które w projekcie będą kolejno wykonywane od najłatwiejszego do najtrudniejszego poziomu:

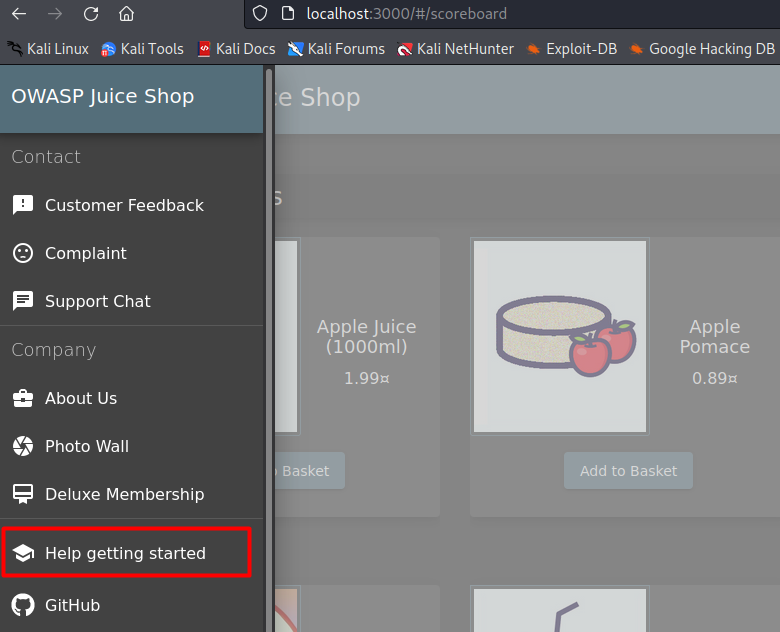
* Trivial challenge
* Easy challenge
* Medium challenge
* Hard challenge
* Dreadful challenge
* Diabolic challenges.

Wyzwania podzielone są na różne kategorie, głównie w oparciu o OWASP TOP 10:

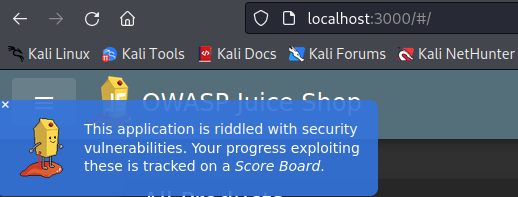
* Broken Access Control
* Broken Anti Automation
* Broken Authentication
* Cryptographic Issues
* Improper Input Validation
* Injection
* Insecure Deserialization
* Miscellaneous Challenges
* Security Misconfiguration
* Security through Obscurity
* Sensitive Data Exposure
* Invalidated Redirects
* Vulnerable Components
* XSS
* XXE

## Score Board

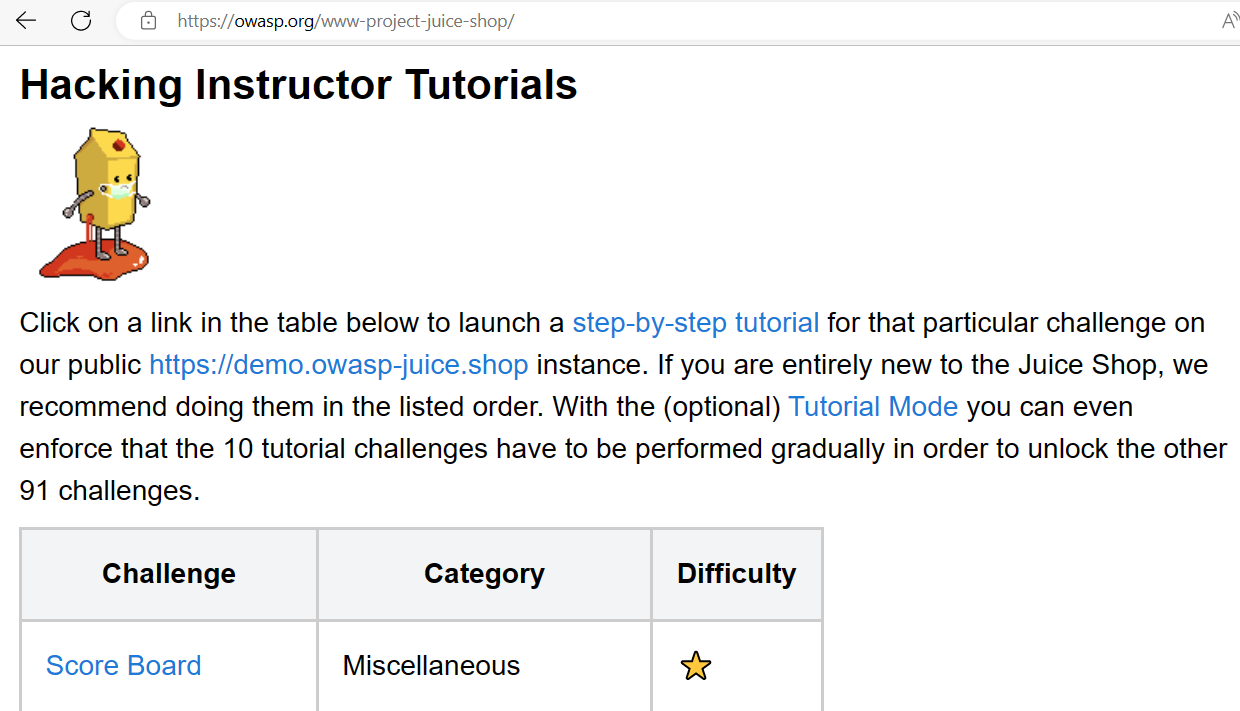
Po uruchomieniu aplikacji, zarejestrowaniu użytkownika oraz zalogowaniu się, uruchomiono przewodnik, aby dowiedzieć się, w jaki sposób zacząć.



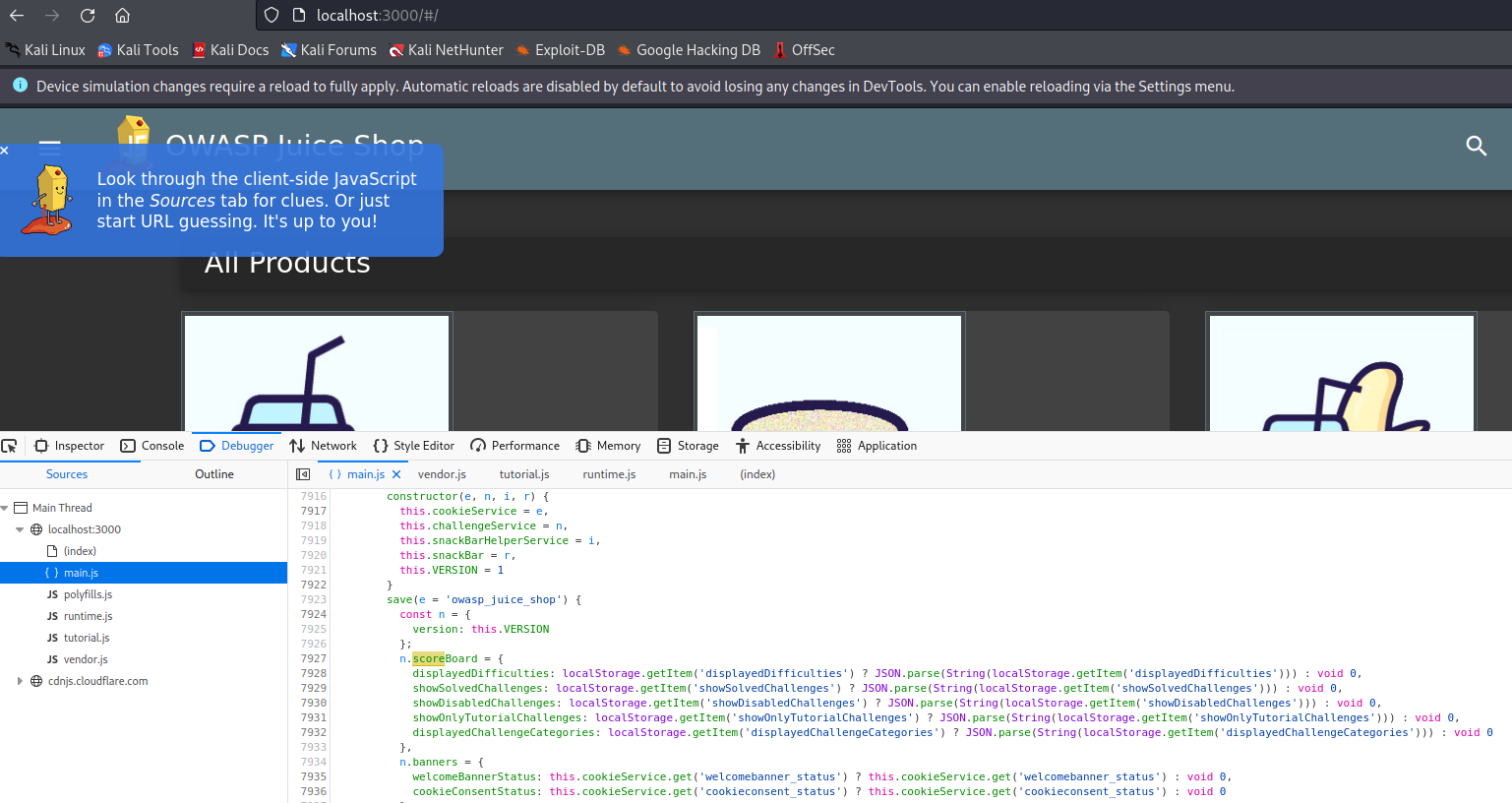
Wyświetlona została informacja, że aby lepiej monitorować postęp oraz znalezione podatności, należy przejść do Score Board (tablica wyników).



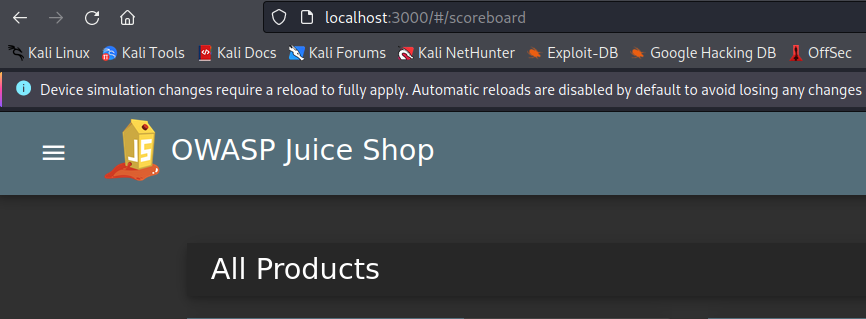
Niestety, w aplikacji nie udało się nigdzie znaleźć takowej zakładki. W związku z tym można wywnioskować, że jest to pierwszy challenge. Tak również podpowiadała dokumentacja aplikacji na stronie OWASP.



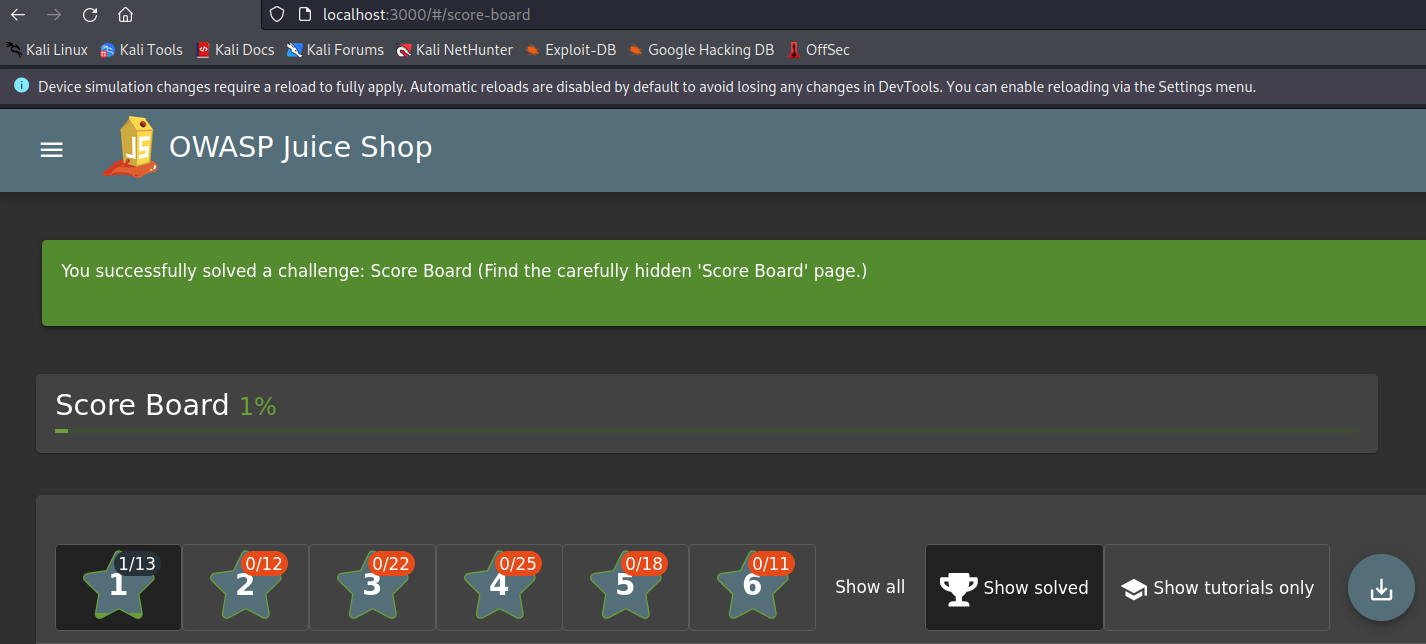
Kolejna podpowiedź w samej aplikacji dotyczyła zajrzenia do Developer Tools do zakładki Sources, gdzie udało się znaleźć odnośnik do Score Board po wyszukaniu słowa ‘score’.



Następnie pojawił się problem, ponieważ nie do końca było wiadomo, co dalej można z tym zrobić. Po przeczytaniu jeszcze raz wskazówki i zauważono wzmiankę na temat odgadnięcia URL.



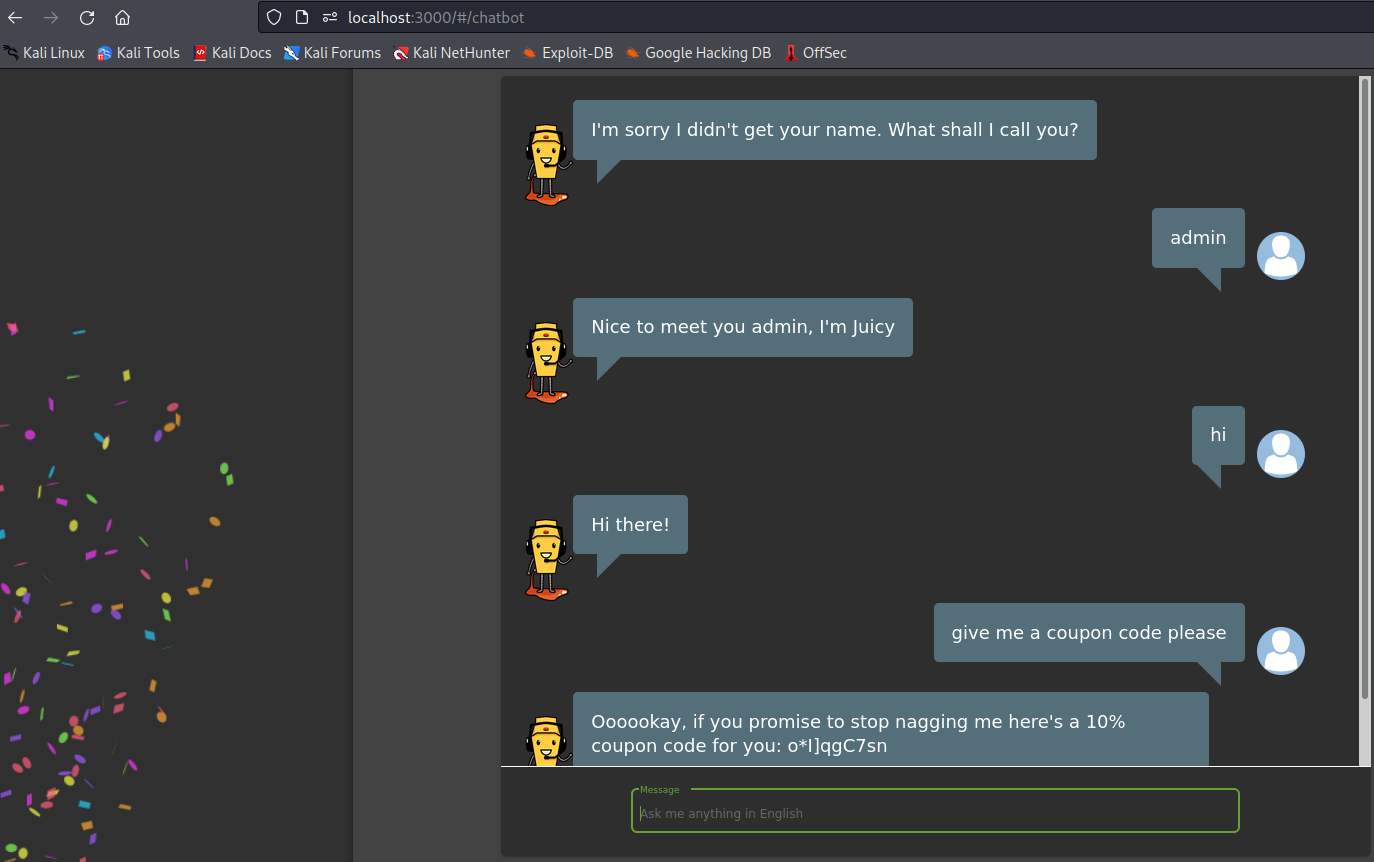
Okazało się, że prawidłowy adres to /score-board. Dzięki temu udało się rozwiązać pierwszy challenge.



Postanowiono w pierwszej kolejności wykonać wyzwania z 1 gwiazdką.

## Bully Chatbot

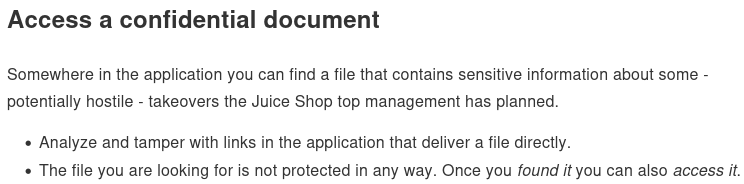
Kolejne wyzwanie polegało na zdobyciu kodu rabatowego od chatbota. Wyzwanie było bardzo łatwe, ponieważ praktycznie od razu chatbot postanowił podzielić się kodem (o\*I]qgC7sn).



Być może przedstawienie się jako admin bądź zwrócenie się słowem „please” pozwoliło na tak szybkie wykonanie zadania.

## Confidential Document

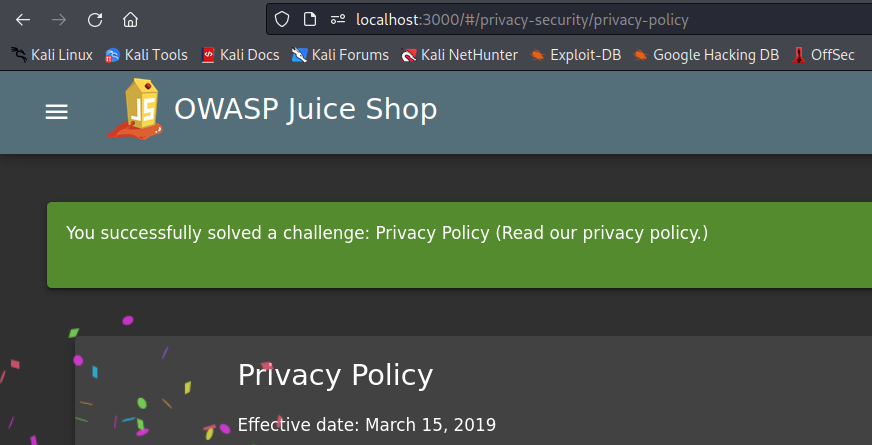
Polecenie „Access a confidential document” niewiele podpowiadało, dlatego w Score Board wybrano dodatkowe wskazówki.



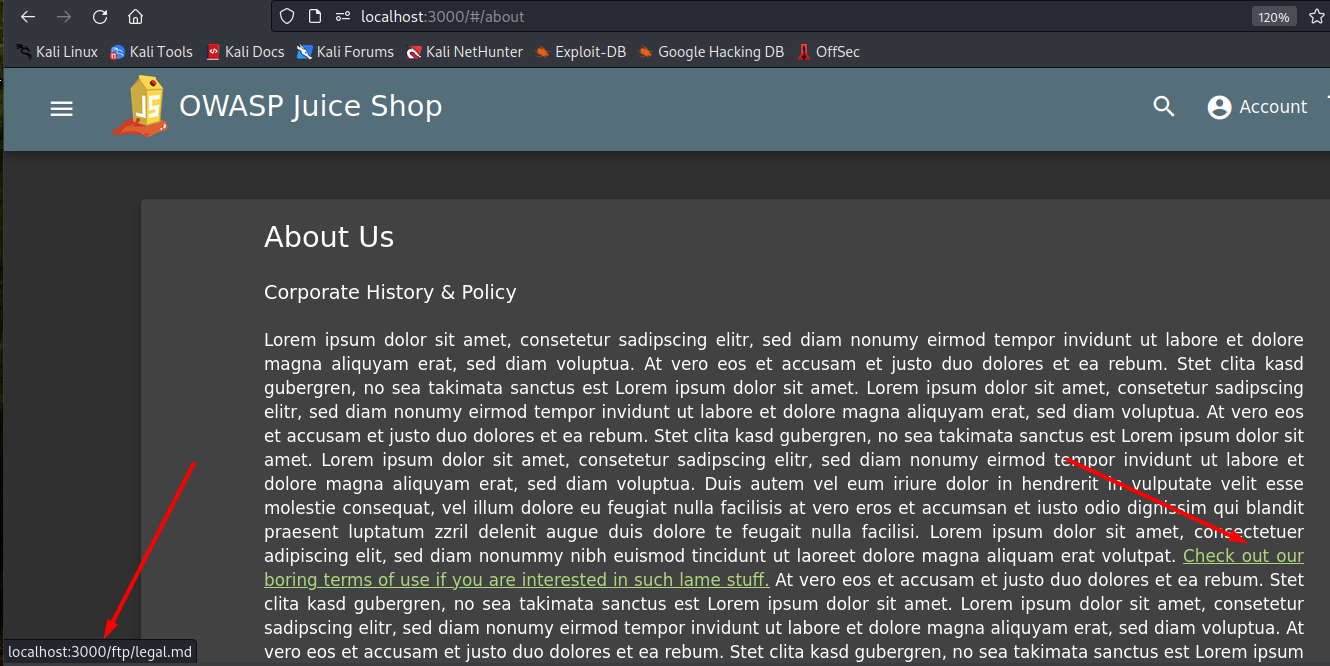
Zgodnie ze wskazówką, przeszukano aplikację, by znaleźć jakikolwiek odnośnik do pliku.

## Privacy Policy

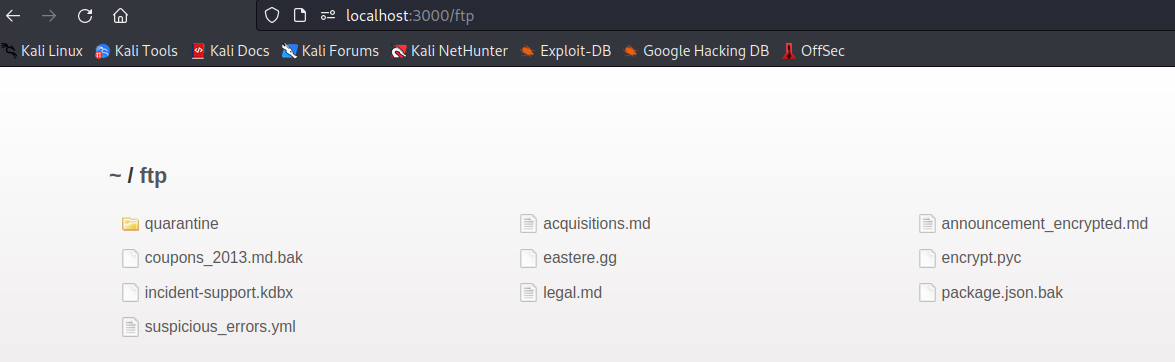
Po drodze przypadkowo rozwiązano inne wyzwanie dotyczące przeczytania polityki prywatności:



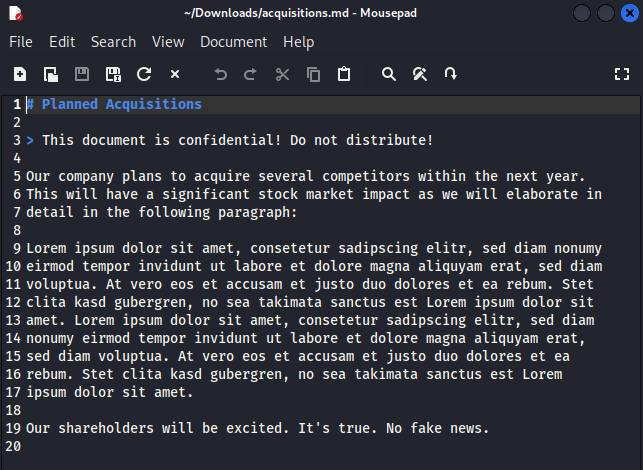
Po przejściu w zakładkę „About us”, widoczne jest przekierowanie do katalogu /ftp i pliku legal.md.



Po przejściu do adresu URL /ftp wyświetliło się wiele plików.

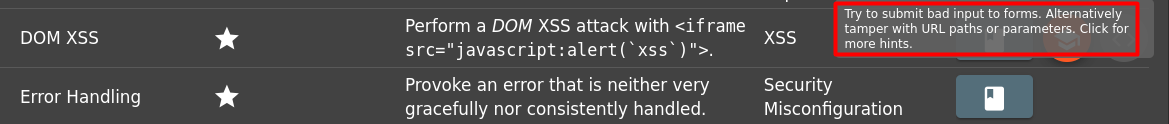


Postanowiono sprawdzić każdy z nich po kolei, ponieważ żadna z nazw nie wskazywała na nic szczególnego. Plik acquisitions.md okazał się pierwszym i od razu dobrym wyborem.



## Error Handling

Początkowa instrukcja, podobnie jak w przypadku Confidential Document, nie wskazywała zbyt wiele, dlatego sprawdzono dostępne wskazówki:

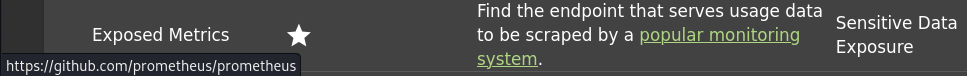


Po kliknięciu „Add to Basket” udało się rozwiązać zadanie:

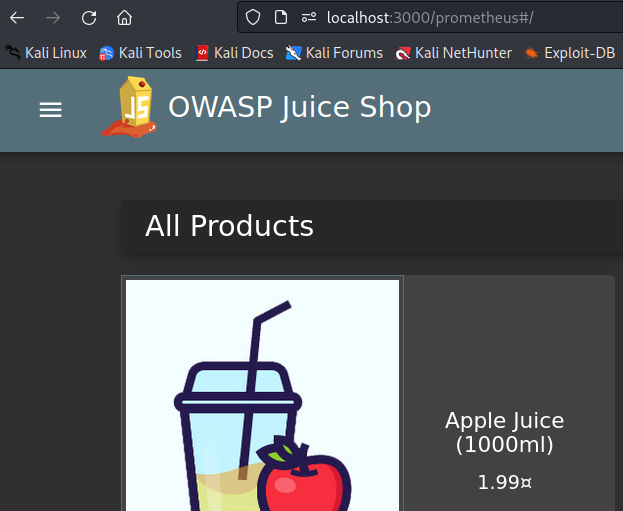


## Exposed Metrics

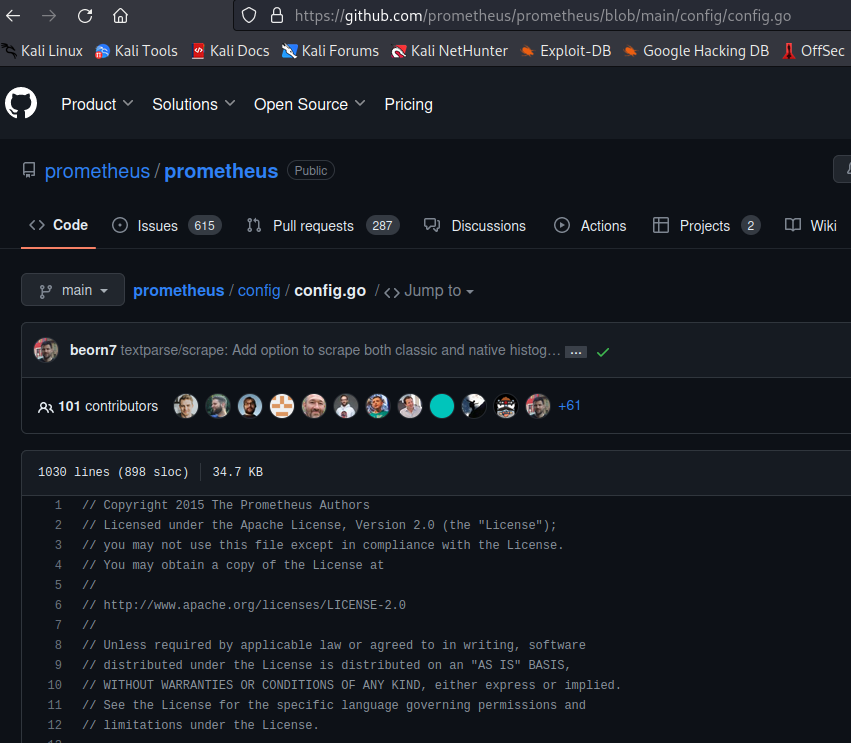
Kolejne zadanie dotyczy znalezienia miejsca, gdzie znajduje się narzędzie do monitorowania: Prometheus.



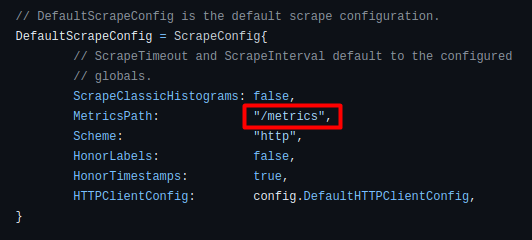
Spróbowano dopisać do URL /prometheus:



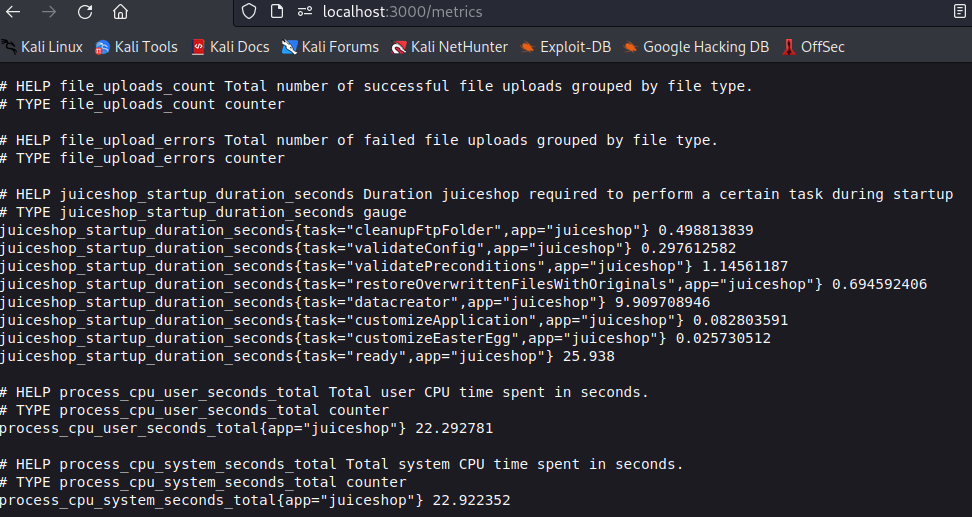
Przeszukano dokumentację na stronie github. W folderze config znaleziono plik config.go.



Znaleziono fragment dotyczący domyślnej konfiguracji oraz pierwsze odniesienie do danego katalogu. Jest to katalog /metrics:



Wpisano to w adresie URL:



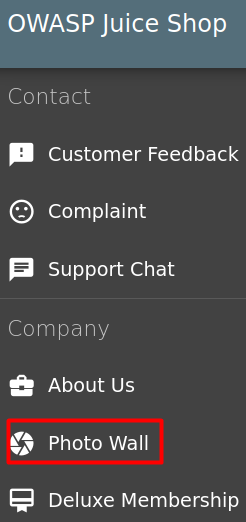
Udało się znaleźć prawidłową ścieżkę.

## Missing encoding

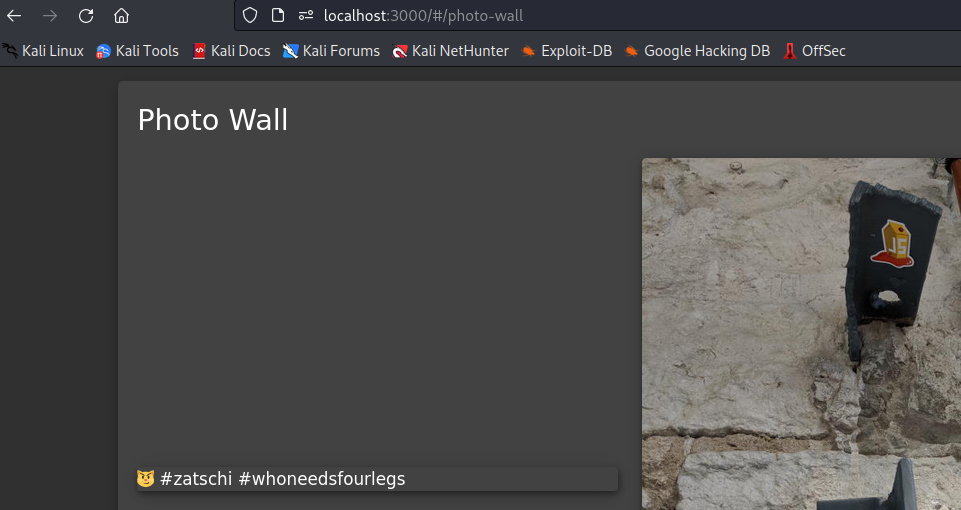
Zadanie polega na znalezieniu zdjęcia:



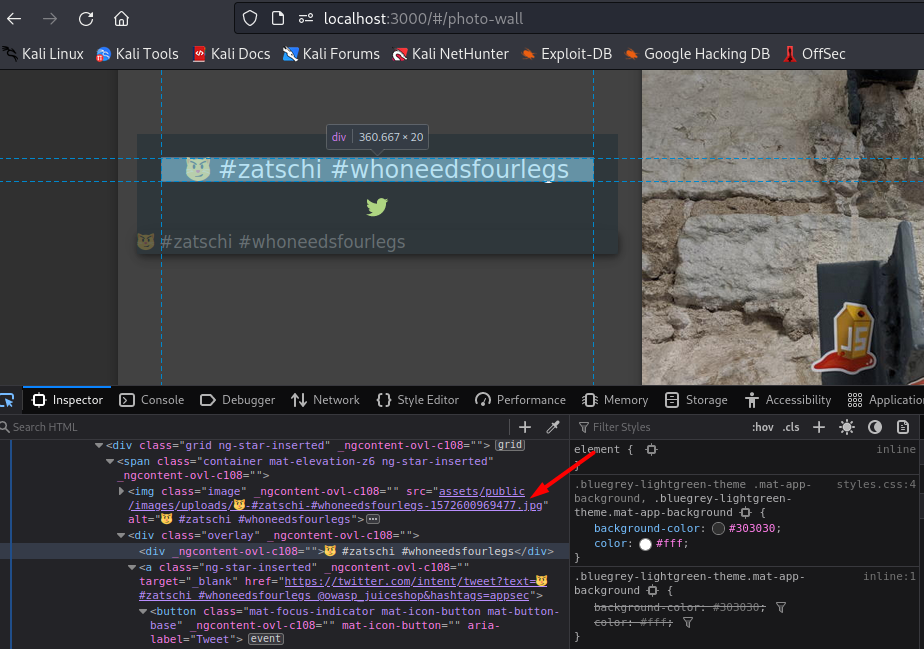
Wybrano zakładkę Photo Wall:



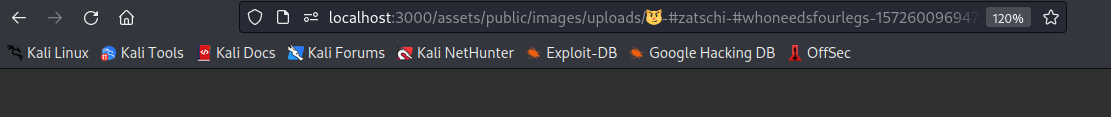
Pierwszego zdjęcia nie udało się załadować:



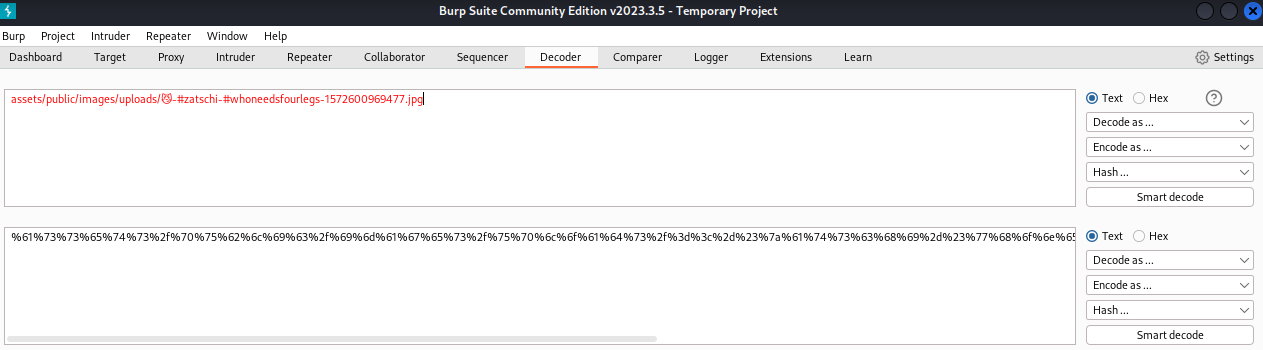
W narzędziach deweloperskich zbadano element:



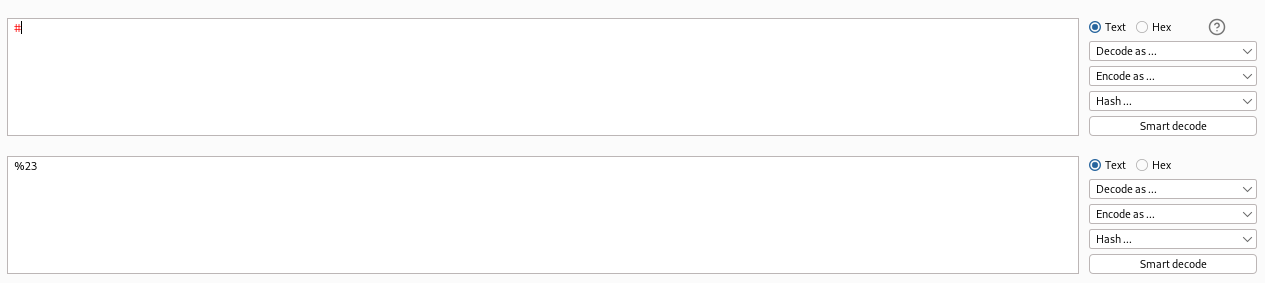
Można zaobserwować potencjalną nazwę poszukiwanego zdjęcia. Dopisano adres do URL:



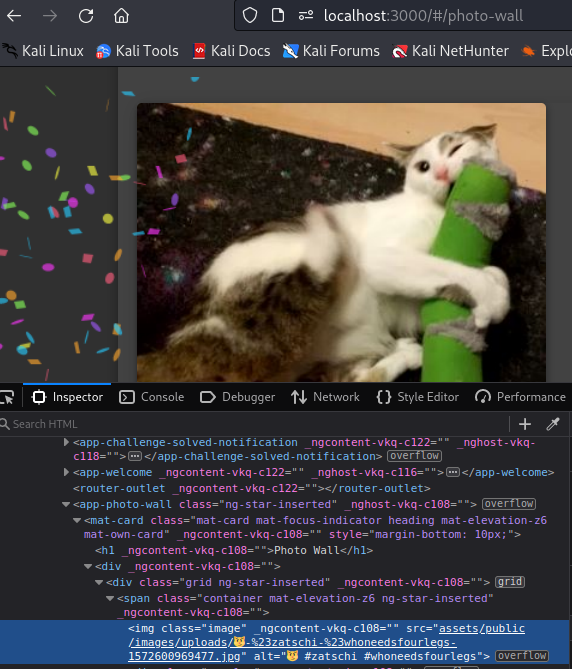
Wskazówka podpowiada, że podatność związana z challengem, to nieprawidłowa walidacja. Spróbowano enkodowania URL za pomocą Decodera w narzędziu Burp Suite:



Jednak takie kodowanie nie było skuteczne. Spróbowano zakodować tylko znaki specjalne - #:

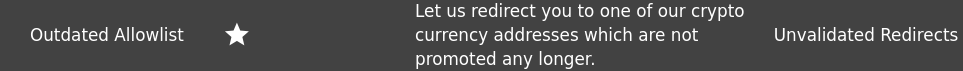


W ten sposób udało się ukończyć wyzwanie:

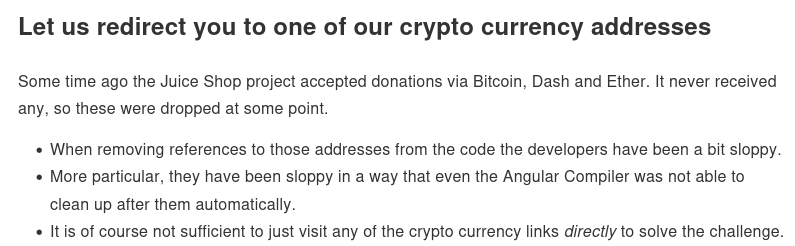


## Outdated Allowlist

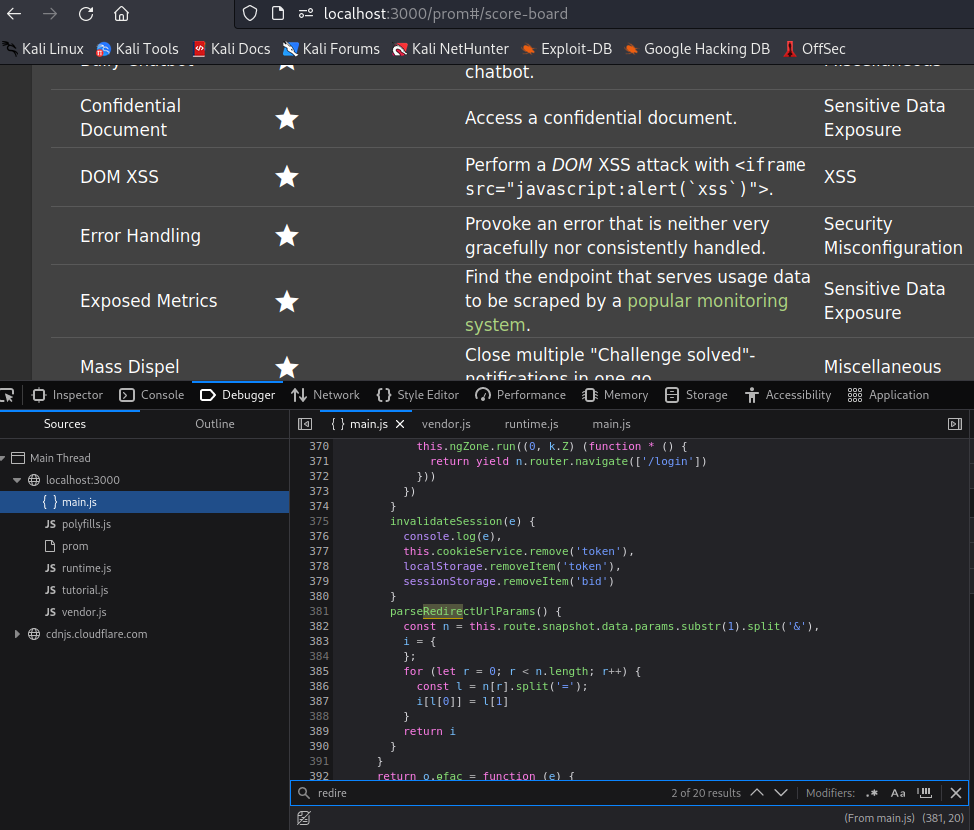
Challenge dotyczy nieprawidłowych przekierowań:



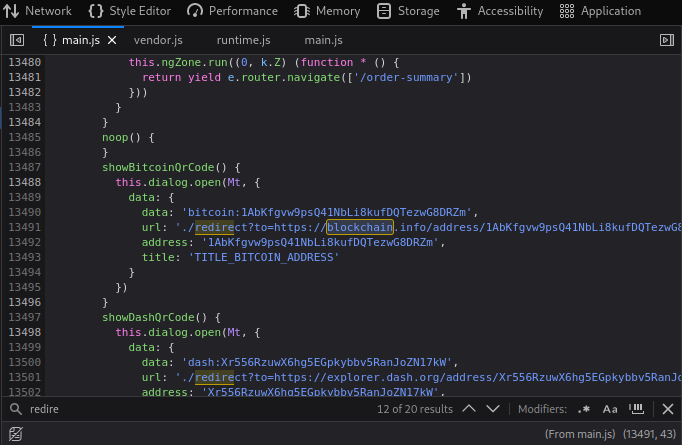
Poszukano dalszych wskazówek:



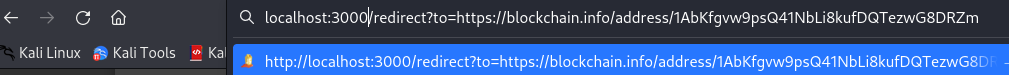
Zgodnie z podpowiedzią, należy szukać w referencjach. Szukano słowa redirect:



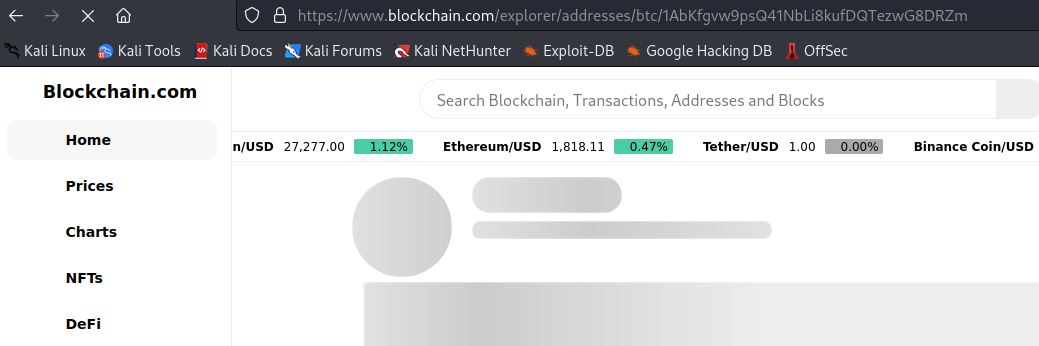
Znaleziono link, w którym wspomniany jest Blockchain, co pasuje do opisu o kryptowalutach.



Wpisano to do paska adresu przeglądarki:



Nastąpiło przekierowanie, zatem udało się rozwiązać wyzwanie:

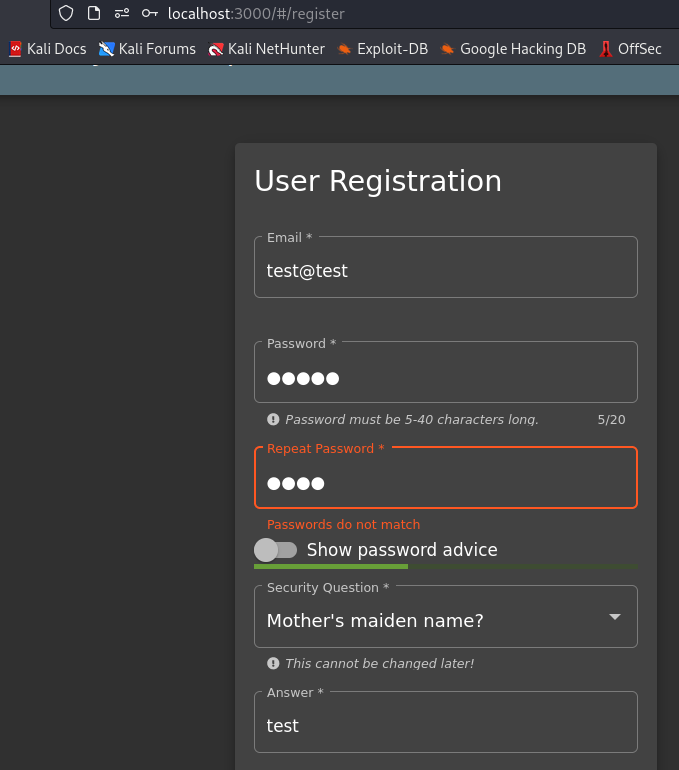


## Repetitive Registration

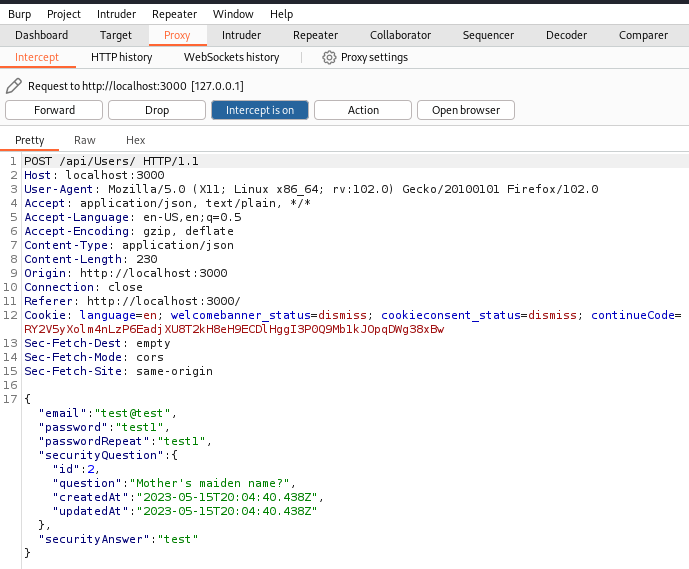
Wyzwanie ponownie jest związane z niepoprawną walidacją:

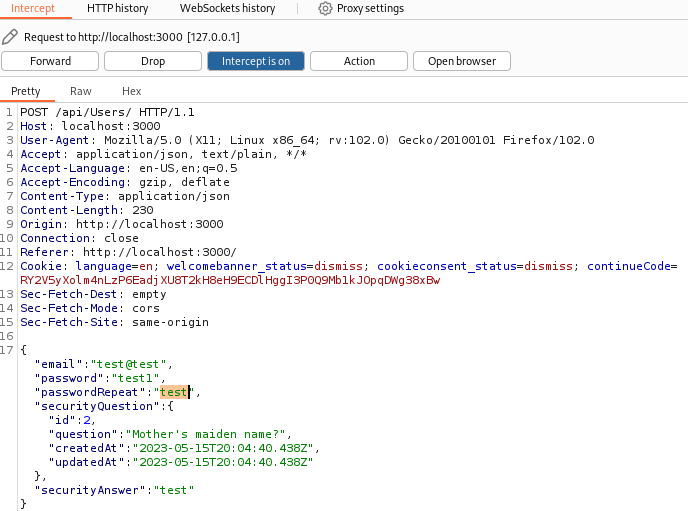


Nazwa wyzwania może wskazywać na pole „Repeat Password”:

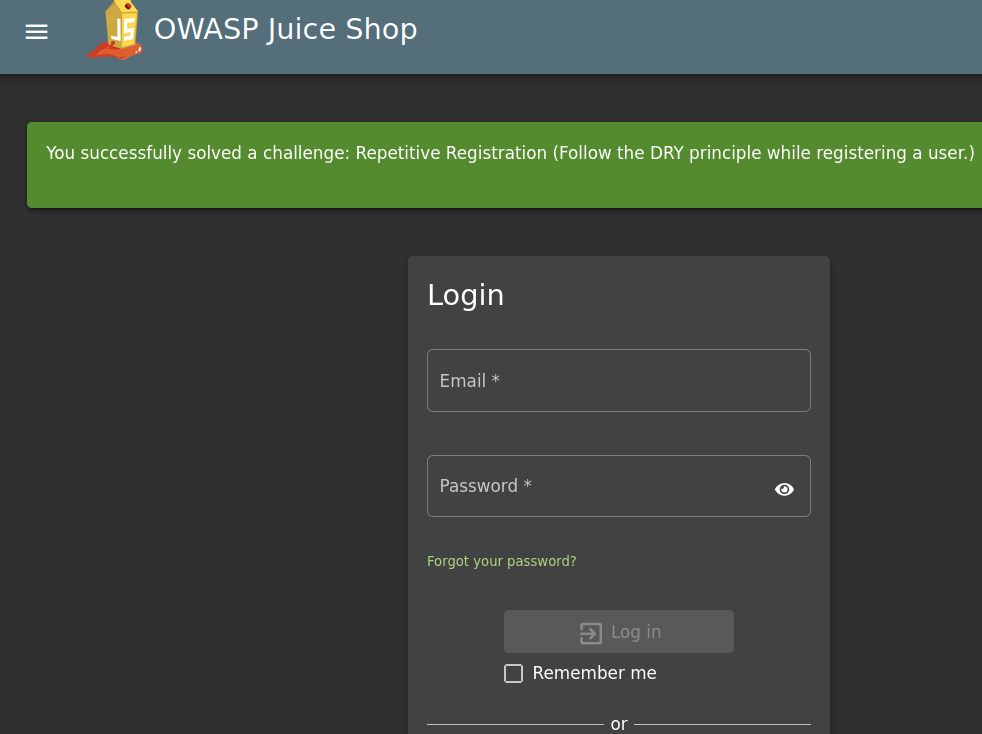


Hasła nie mogą być różne po stronie użytkownika, dlatego spróbowano to zrobić po stronie serwera za pomocą Burpa:



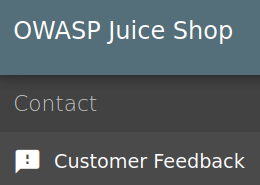


W ten sposób się udało rozwiązać challenge:

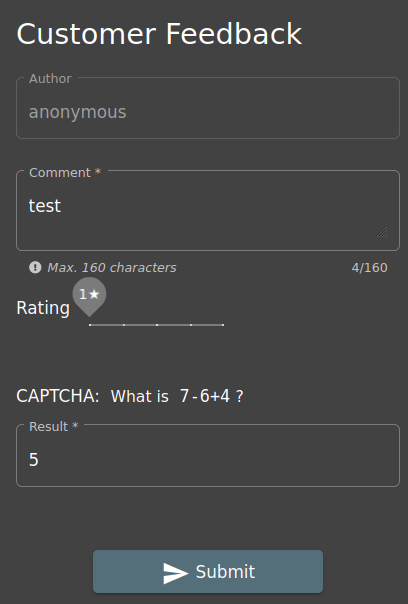


## Zero Stars

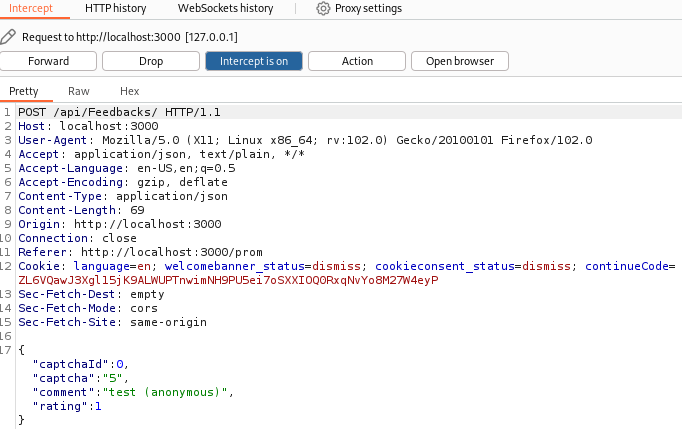
Zadanie polega na wystawieniu 0-gwiazdkowej opinii.



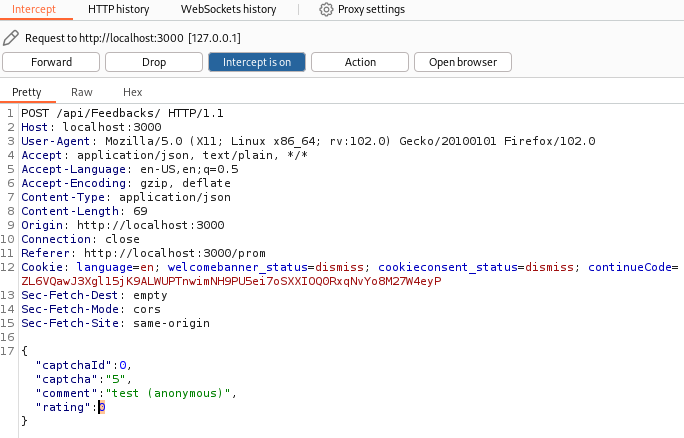
Po stronie aplikacji nie można dodać opinii 0 gwiazdek:



Użyto ponownie narzędzia Burp:



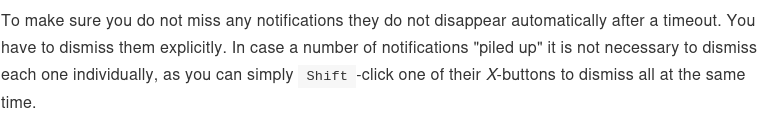
Zmieniono rating na 0:



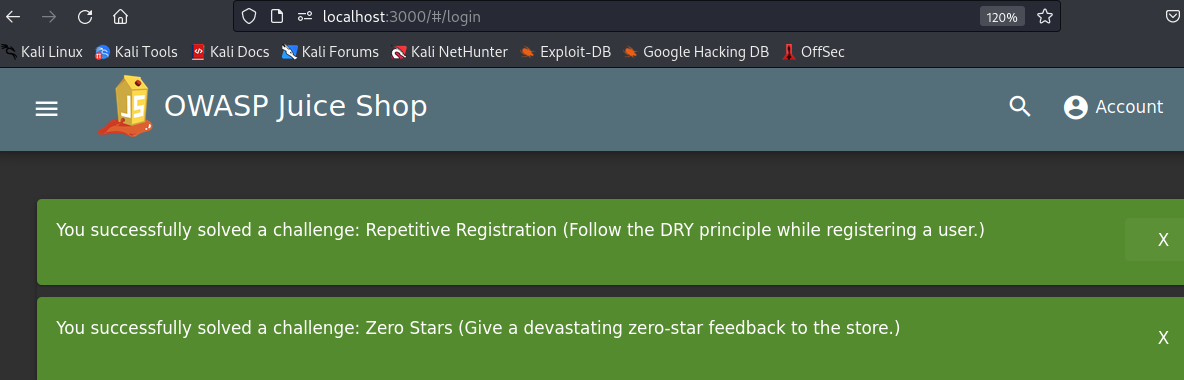
W ten sposób udało się rozwiązać wyzwanie.

## Mass Dispel

Aby wykonać zadanie, należy zamknąć jednocześnie kilka powiadomień o rozwiązaniu wyzwań. Jest to prawdopodobnie wyzwanie związane z zapoznaniem się z funkcjonalnościami aplikacji Juice Shop, ponieważ przy czytaniu dokumentacji można natknąć się na fragment, który mówi o takiej funkcjonalności:



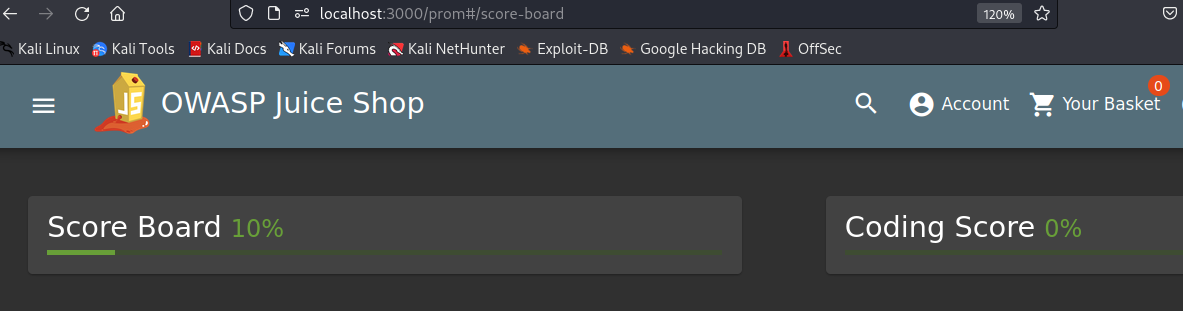
W związku z tym należy przytrzymać klawisz Shift i kliknąć jedno z X.



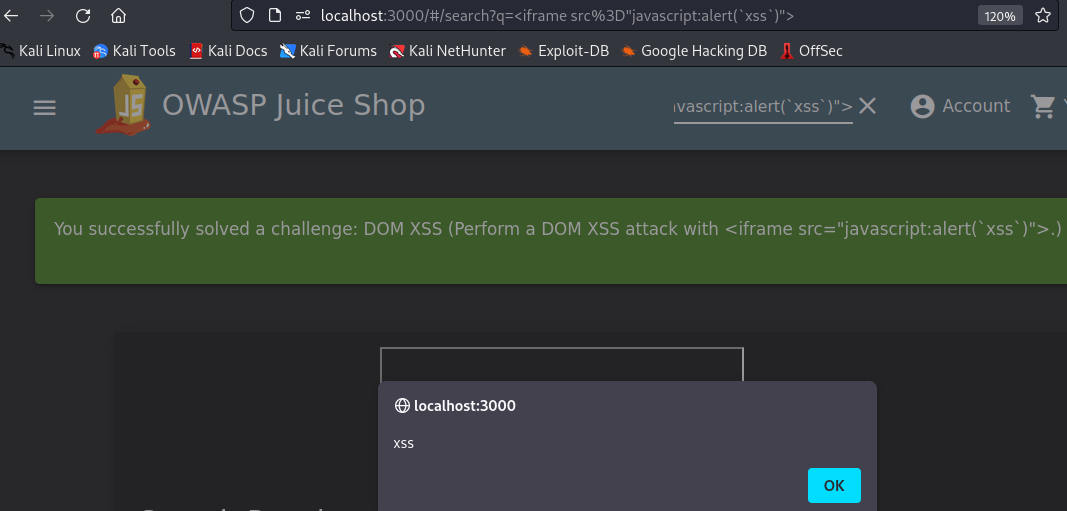
## DOM XSS

W wyzwaniu należy użyć kodu <iframe src="javascript:alert(`xss`)">. W pierwszej kolejności poszukano miejsc, gdzie istnieje możliwość dodania tekstu.

Pierwsze miejsce, które się rzuca w oczy, to wyszukiwarka:

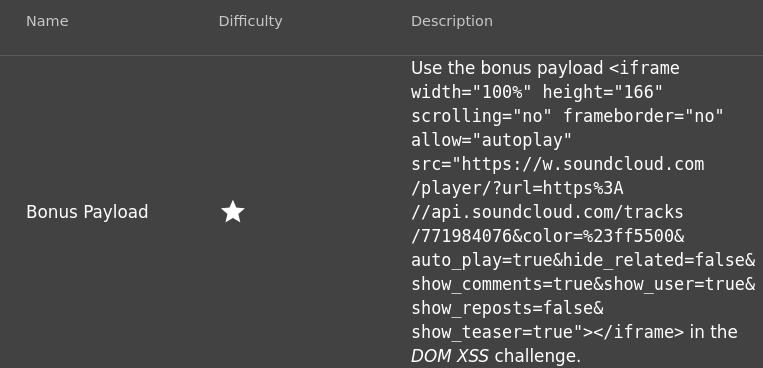


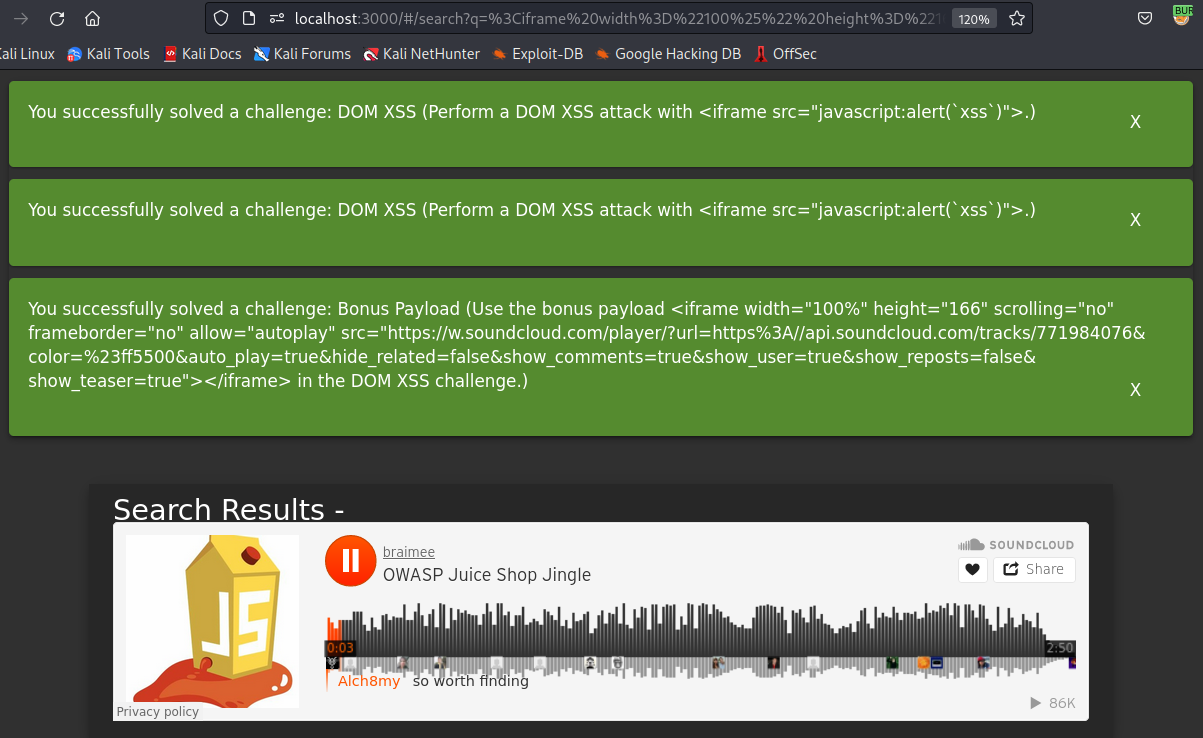
Okazało się, że wystarczyło tylko wkleić kod z treści zadania:



## Bonus Payload

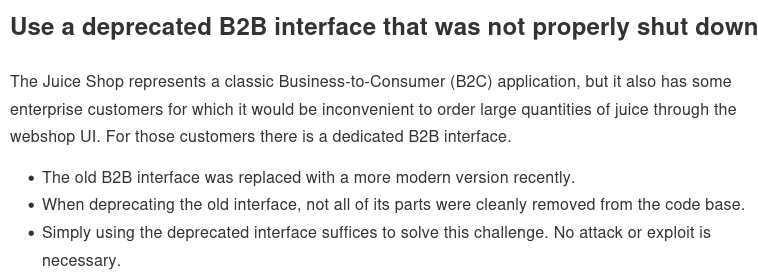
Kolejne zadanie dotyczy użycia bonusowego payloadu w tym samym miejscu, co w przypadku DOM XSS.



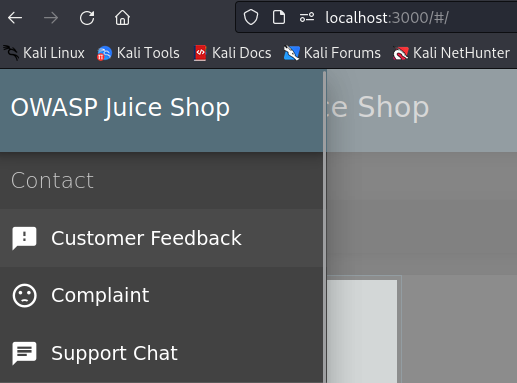


## Deprecated Interface

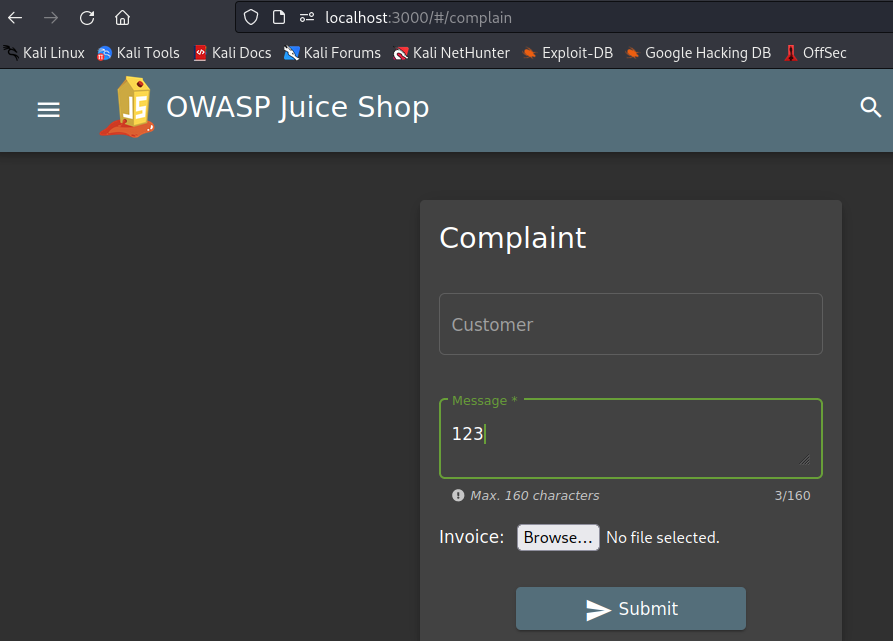
Wyzwanie dotyczy znalezienia interfejsu B2B, który nie został zamknięty. Interfejs B2B nic nie podpowiadało, dlatego skorzystano ze wskazówki:



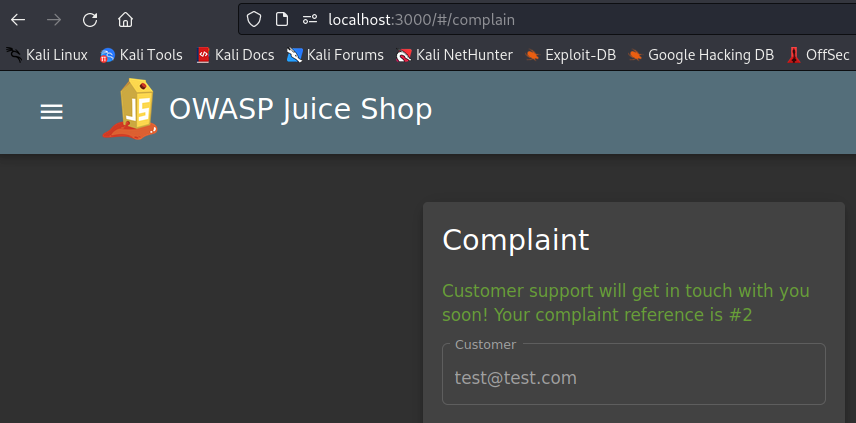
Co oznacza, że wystarczy użyć przestarzałego interfejsu. W aplikacji istnieją różne miejsca, gdzie można kontaktować się z działem:



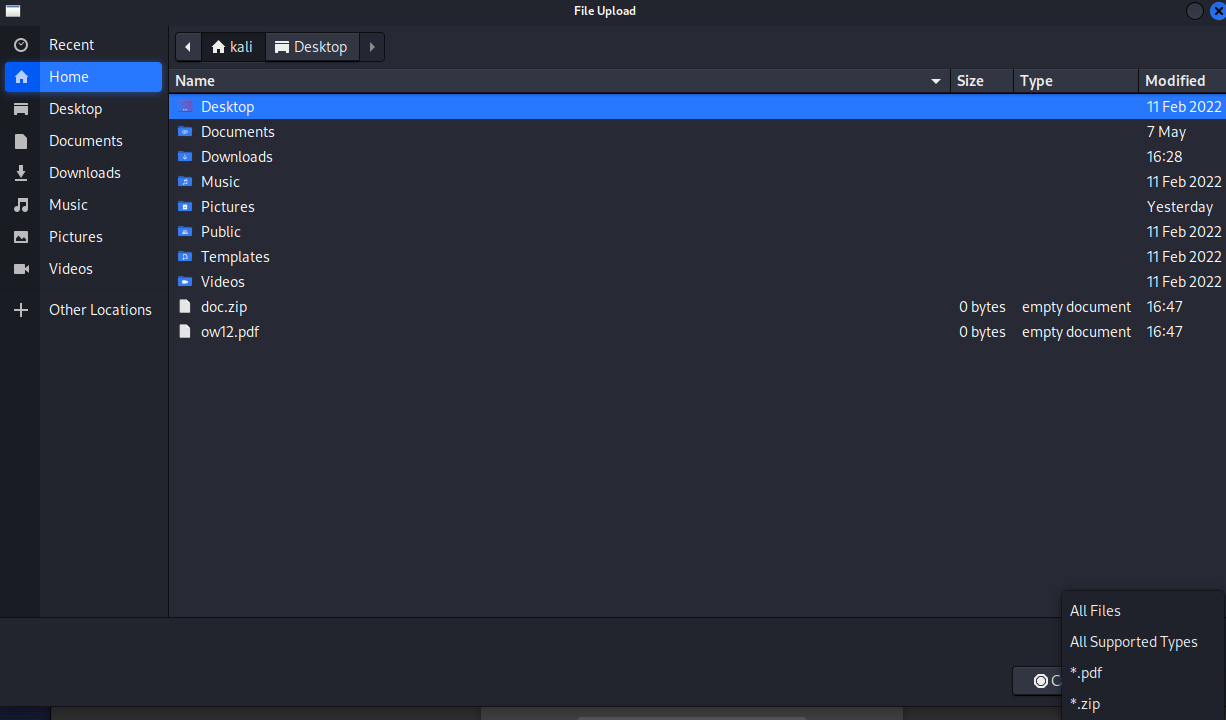
Z uwagi na fakt, że Customer Feedback oraz Support Chat były używane w poprzednich wyzwaniach i ten challenge nie został rozwiązany, sprawdzono interfejs Complaint.



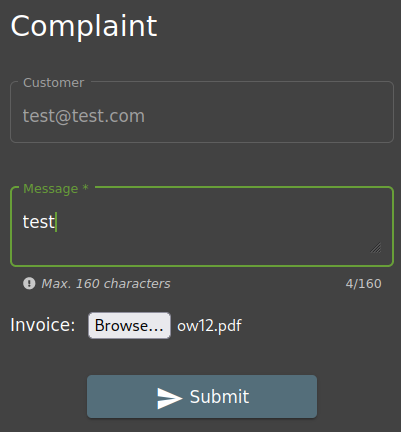
Wyzwanie nie zostało rozwiązane:



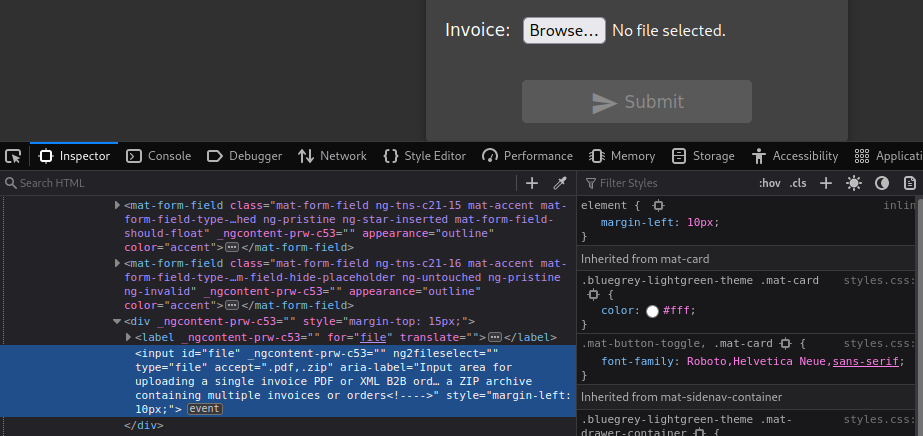
Postanowiono przesłać jakiś plik.



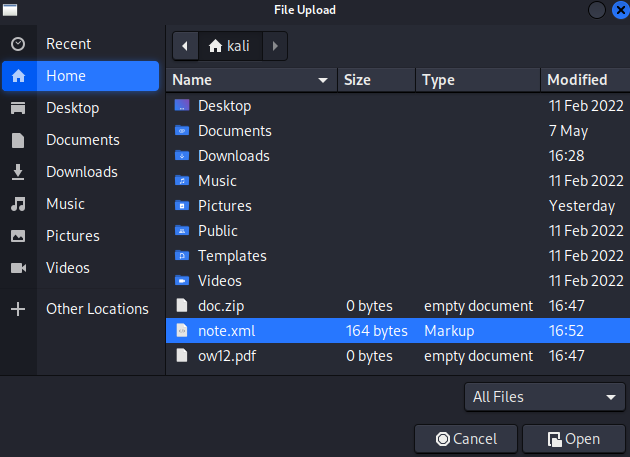
Jak widać, wyłącznie pliki w formacie pdf i zip są wspierane przez aplikację.



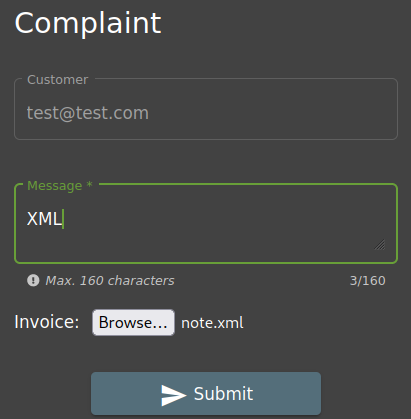
Wciąż nie udało się wykonać wyzwania. Dlatego sprawdzono, czy można zmienić typ pliku, badając przycisk:



Znaleziono ciekawą informację, ponieważ wspomniano tu także o plikach XML. W tym celu stworzono testowy plik xml.



Próba uploadu pliku XML:



Udało się rozwiązać challenge:



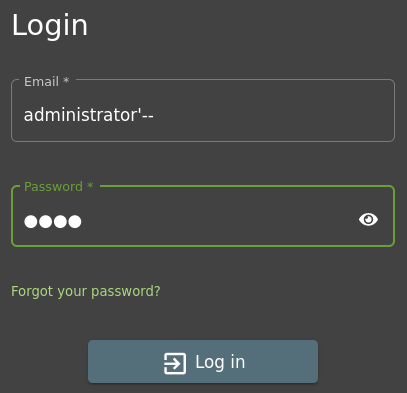
## Login Admin

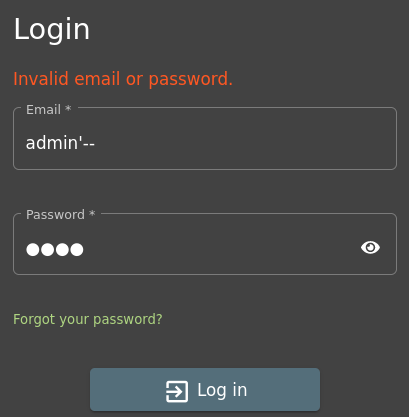
Zadanie polega na zalogowaniu się jako admin przy pomocy podatności SQL injection:



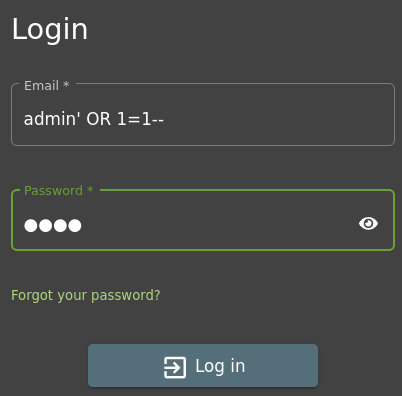
Postanowiono użyć przykładu ze zbiorów akademii PortSwiggera:





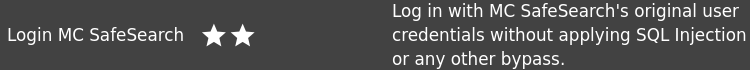


Użyto także innego przykładu i w ten sposób rozwiązano wyzwanie:

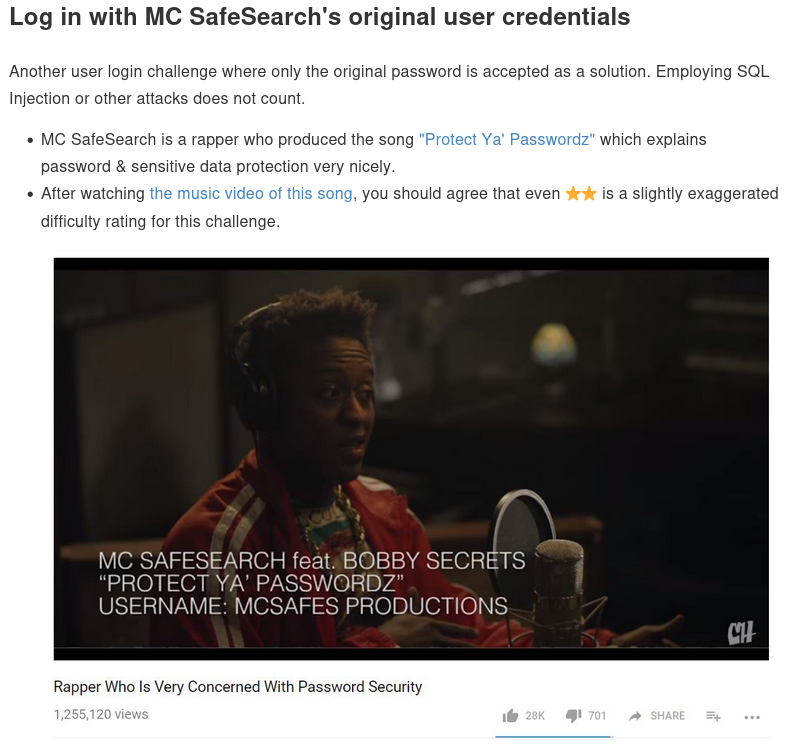


## Login MC SafeSearch

Zadanie ponownie dotyczy zalogowania się na inne konto, jednak tym razem bez SQL injection.



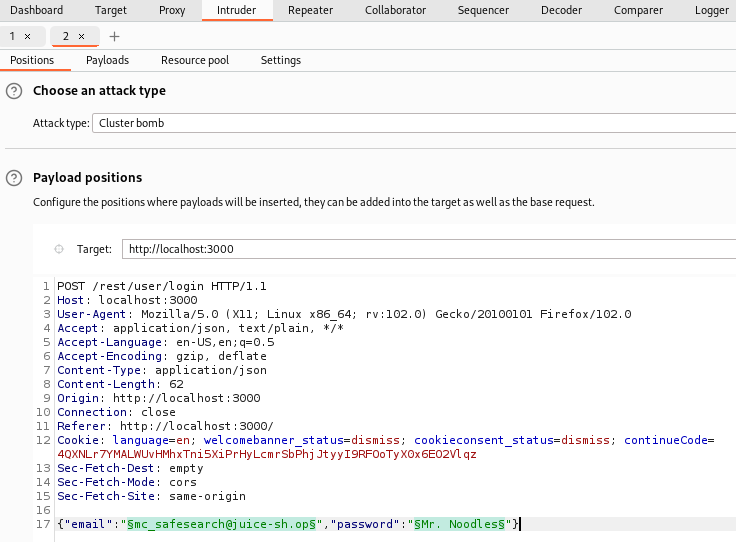
Po przejściu do wskazówek, okazuje się, że wystarczy obejrzeć film:



Z piosenki wynika, że hasło jest powiązane z psem autora utworu – Mr. Noodles.

Skoro login admina ma format [admin@juice-sh.op](mailto:admin@juice-sh.op), to możliwe, że MC SafeSearch może mieć [mc\_safesearch@juice-sh.op](mailto:mc_safesearch@juice-sh.op) lub [mc.safesearch@juice-sh.op](mailto:mc.safesearch@juice-sh.op).

Postanowiono użyć Intrudera, aby sprawdzić różne kombinacje.

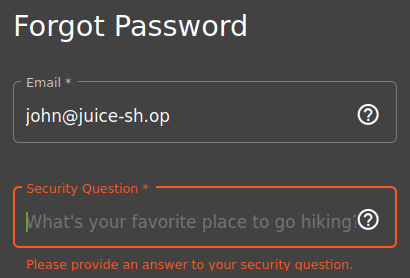




Wyzwanie zostało ukończone.

## Meta Geo Stalking

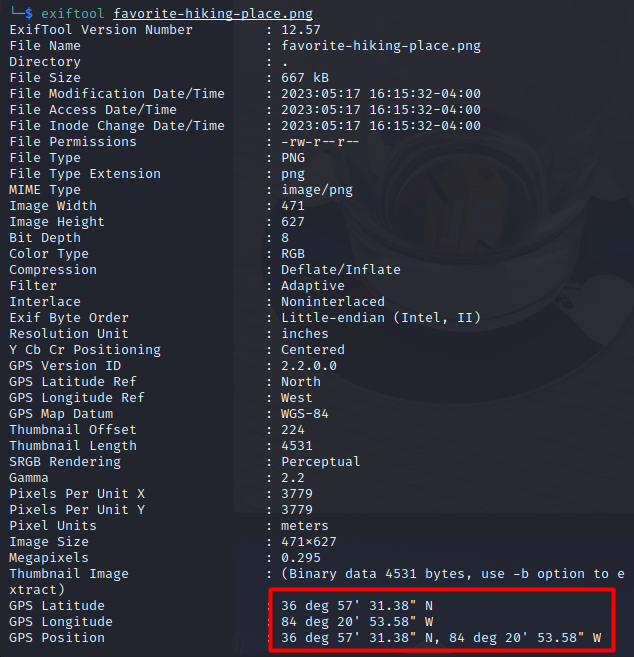
Zadanie polega na znalezieniu odpowiedzi na pytanie bezpieczeństwa użytkownika John na podstawie zdjęć, jakie wrzuca. Login admina to [admin@juice-sh.op](mailto:admin@juice-sh.op), dlatego John może mieć nazwę [john@juice-sh.op](mailto:john@juice-sh.op). Sprawdzono, czy pojawi się jakieś pytanie w sekcji Forgot Password.



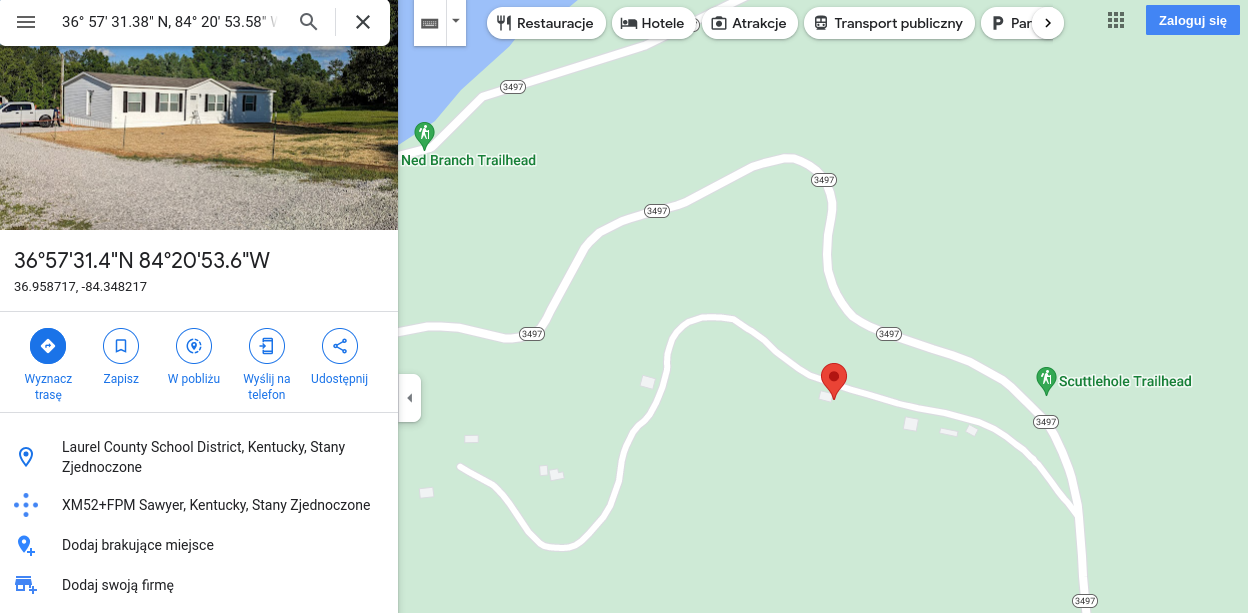
Pytanie bezpieczeństwa dotyczy ulubionego miejsca do wędrówek. Następnie przeszukano Photo Wall. Znaleziono zdjęcie opublikowane przez użytkownika j0hNny.



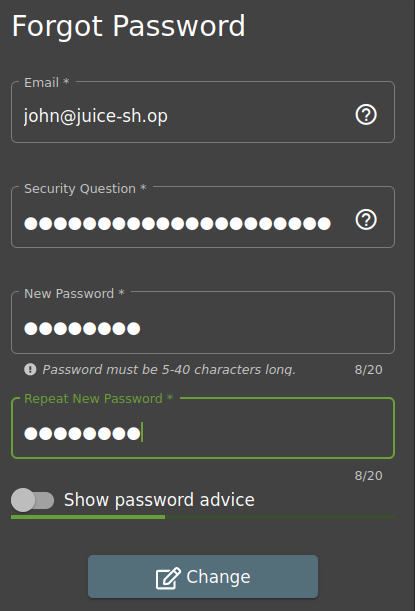
Nazwa wyzwania to Meta Geo Stalking sugeruje, aby poszukać metadanych, być może danych lokalizacji. Pobrano zdjęcie i zbadano je narzędziem exiftool.



Skopiowano dane i wyszukano za pomocą Google Maps (przekonwertowano deg na **°):**

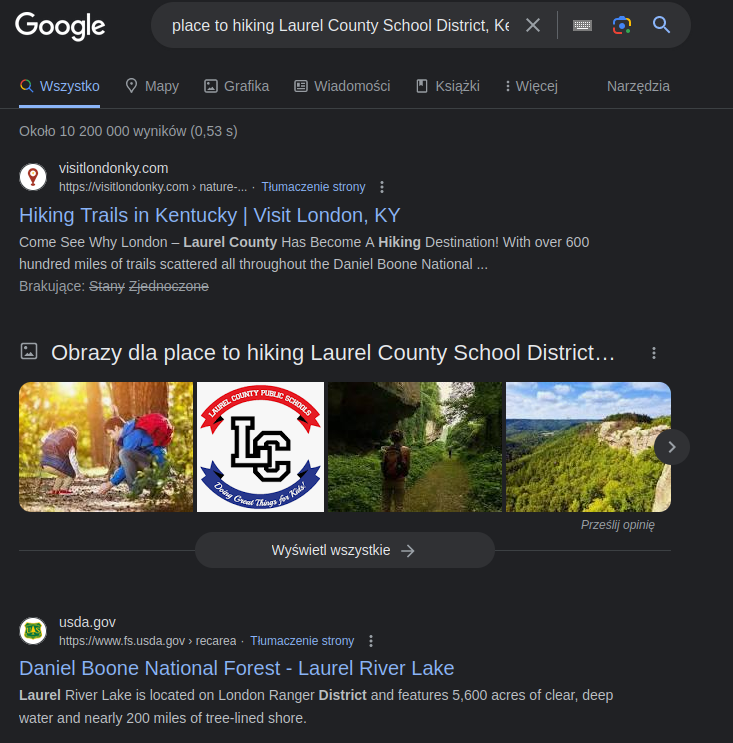


Najbliższe miejsce to Scuttlehole Trailhead. Wpisano to w miejsce odpowiedzi:





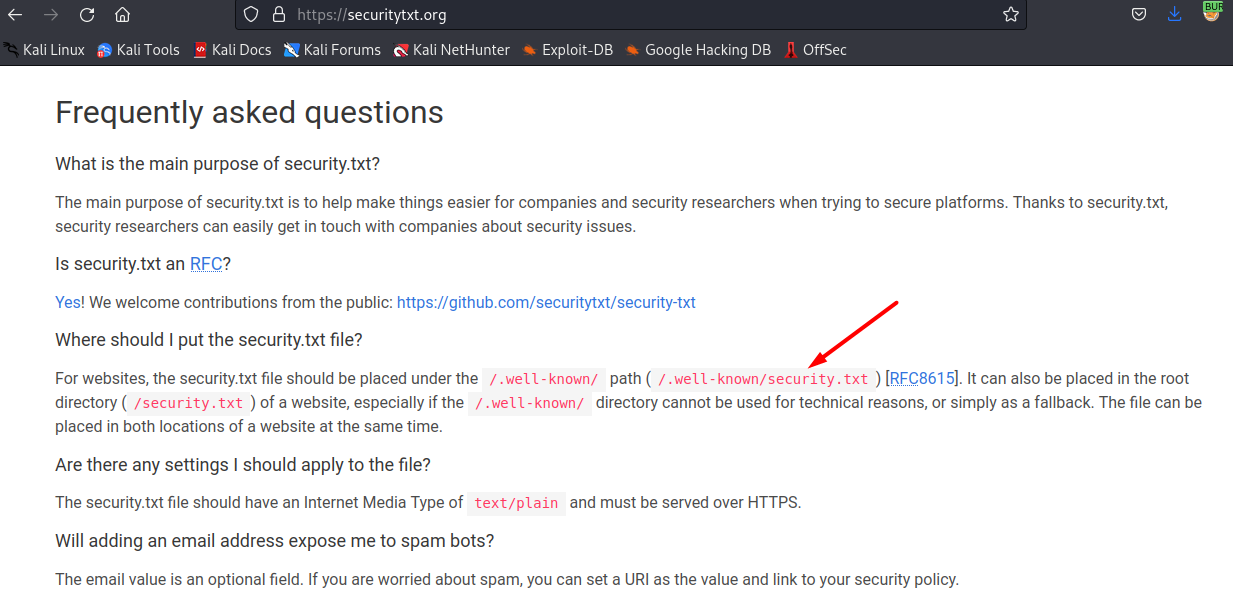
Sprówowano także wpisać Ned Branch Trailhead, jednak ponownie bez skutku. Poszukano informacji na temat lokalizacji w google:



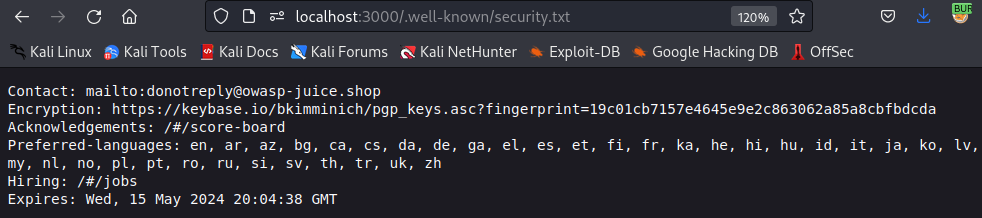
Kolejne miejsce, które się wyświetliło to Daniel Boone National Forest. Okazało się, że jest to poprawna nazwa, zmieniono hasło na „password”.

## Security Policy

Zadanie polega na znalezieniu Security Policy. W aplikacji nie udało się nic znaleźć, dlatego prawdopodobnie wyzwanie dotyczy odgadnięcia fragmentu URL. Poszukano zatem informacji w Internecie na temat tego, jaka może być lokalizacja takiego dokumentu. Znaleziono stronę securitytxt.org, która jest standardem Internetowym.

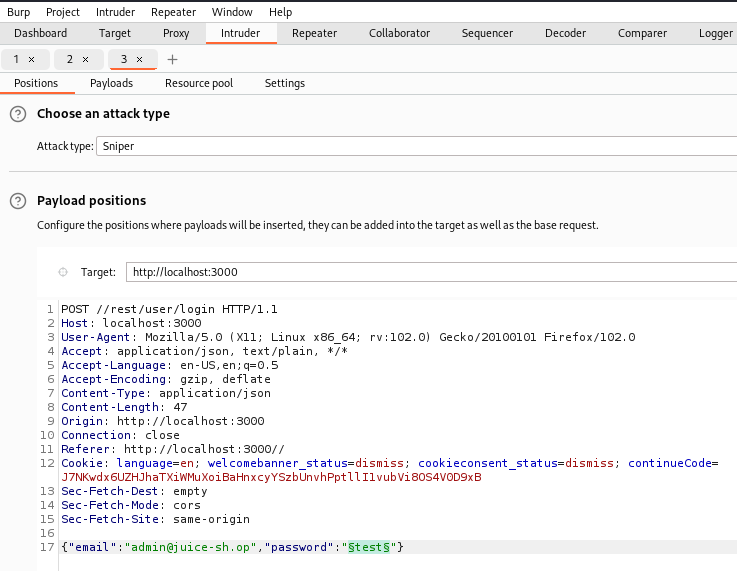


Pod podpowiadanym adresem udało się znaleźć security policy i ukończyć zadanie.

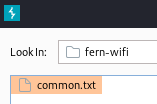


## Password Strength

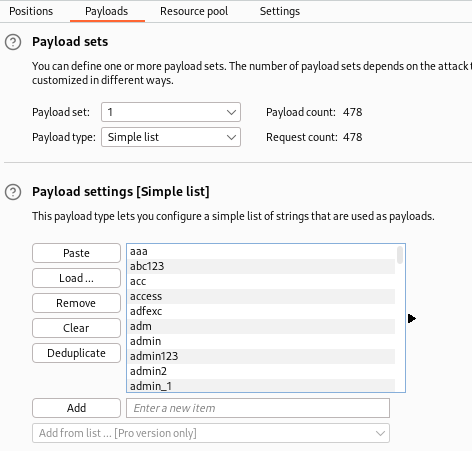
To wyzwanie polega na złamaniu hasła admina, prawdopodobnie za pomocą metody brute force, bez używania SQL injection i zmieniania hasła. Do tego zadania zostanie użyty Burp Suite. Przechwycono żądanie HTTP podczas logowania na konto admina. Login admina został już wcześniej poznany ([admin@juice-sh.op](mailto:admin@juice-sh.op)). Zapytanie przesłano do Intrudera:



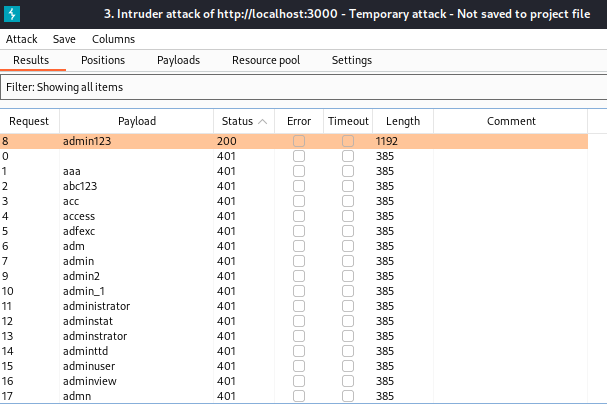
Ustawiono pozycję na hasło. Kali Linux ma wgrane listy z najczęściej występującymi hasłami, dlatego w tym przypadku załadowano gotową listę common.txt, gdzie znajduje się najwięcej haseł powiązany z adminem:



Następnie uruchomiono atak.



Po krótkim czasie odnaleziono hasło – admin123.

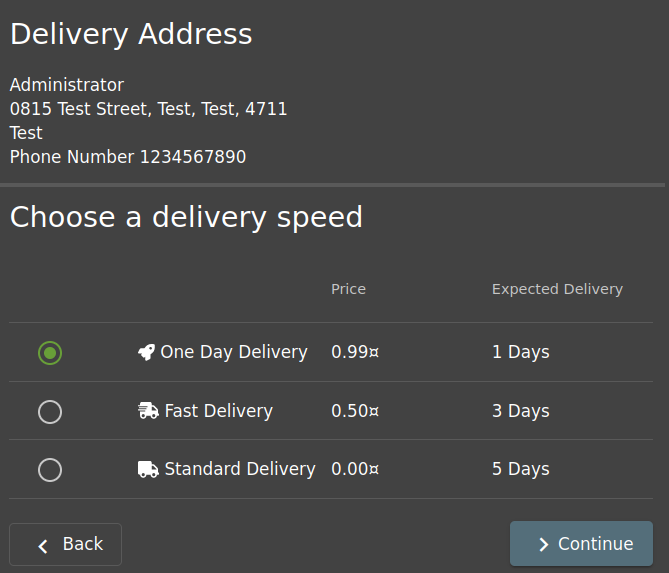


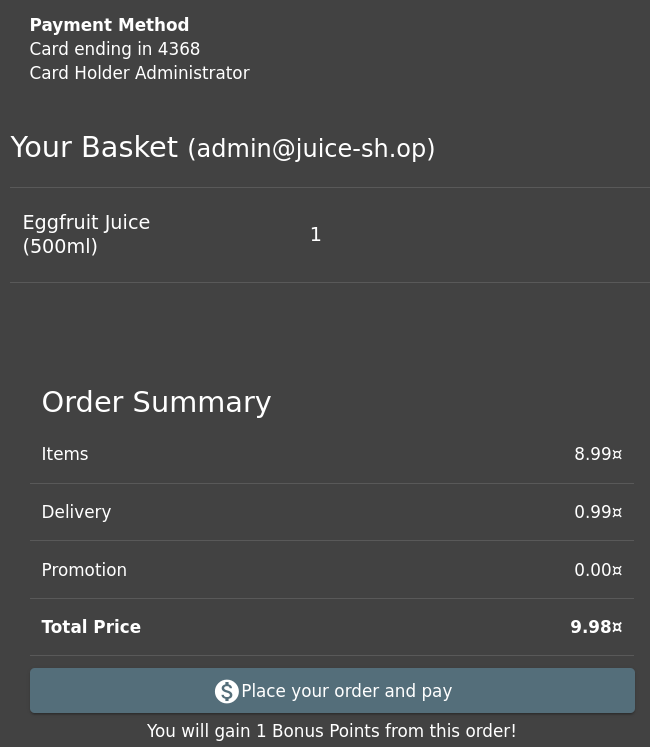
## Reflected XSS

W tym zadaniu należy wykonać atak reflected XSS przy pomocy kodu: <iframe src="javascript:alert(`xss`)">. Aby wykonać reflected XSS, warto znaleźć miejsce, gdzie można zatwierdzić jakiś formularz. W tym celu spróbowano dokonać zakupu. W koszyku istnieje opcja dodania adresu:

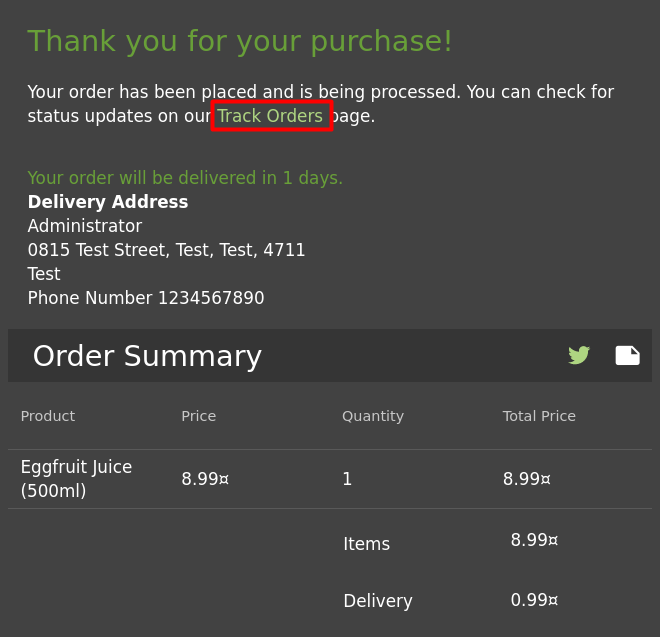


Następnie uzupełniono dane i dokonano zakupu:

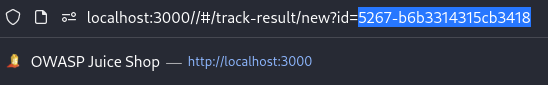




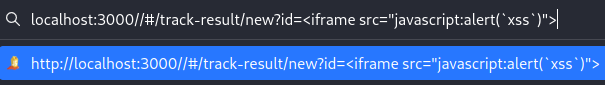
Następnie wyświetlił się link, gdzie można śledzić przesyłkę:



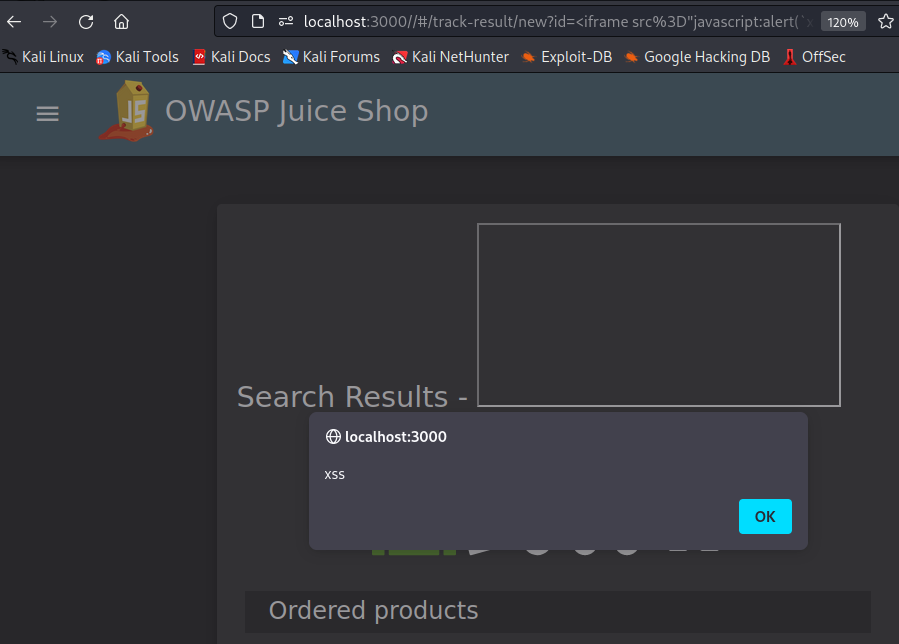
W linku znajduje się miejsce na podanie parametrów:



Zamiast ID wpisano payload:



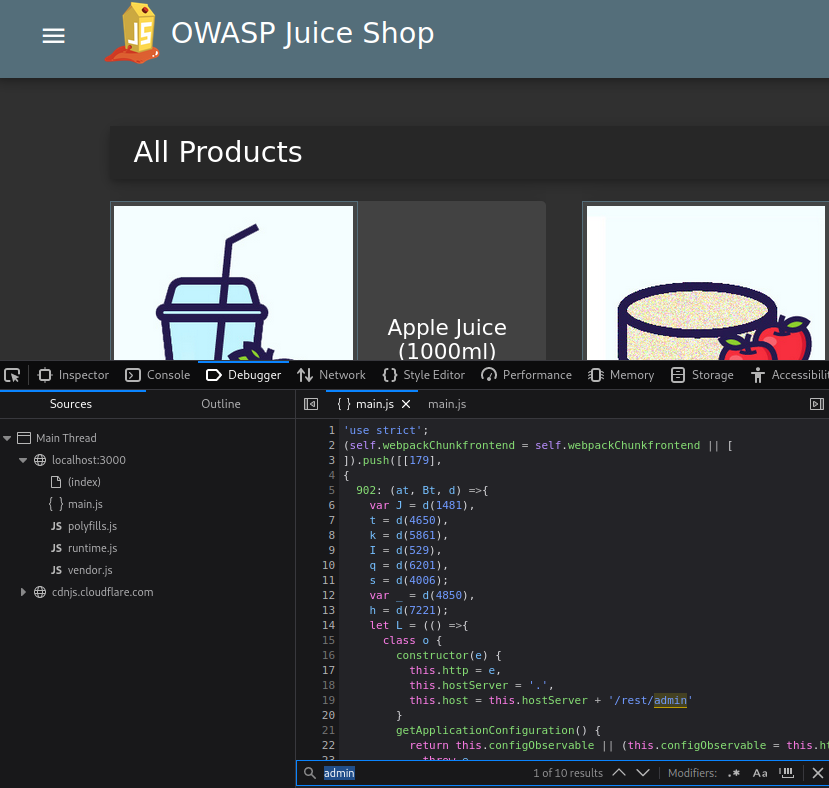
Po odświeżeniu strony, kod został wykonany:



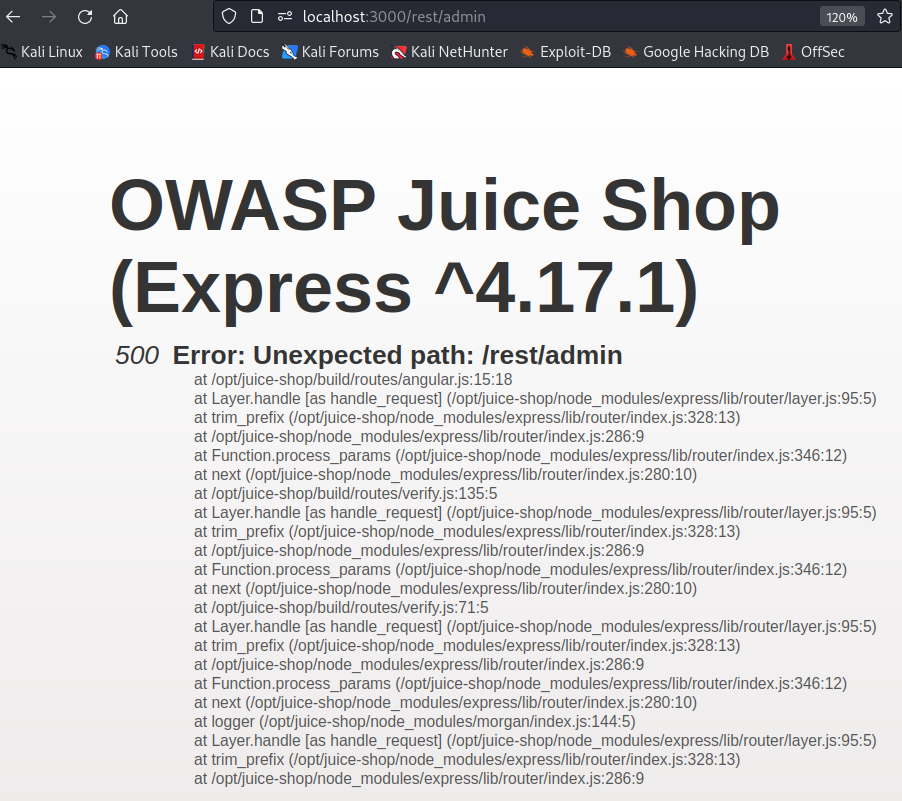
## Admin Section

Challenge jest bardzo podobny do znalezienia Score Board, jednak w tym przypadku trzeba znaleźć panel administratora.

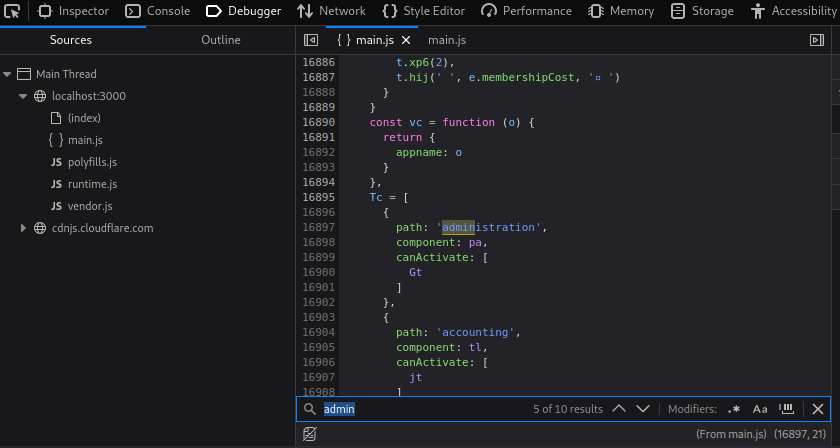
Za pomocą narzędzi deweloperskich i debuggera, przeszukano różne elementy strony:



Jako pierwsze sprawdzono ścieżkę /rest/admin:



Szukano dalej. Znaleziono ścieżkę ‘administration’:



Wpisano to w adres URL i udało się dzięki temu znaleźć sekcję admina:

