yof

Raport bezpieczeństwa: pełny

Niezależna kontrola aplikacji internetowej

shop.example

https://shop.example/

Przykład: zmyślona strona

Przykład: zmyślona strona. To przykładowy raport dla shop.example, strony, która nie istnieje. Wszystkie ustalenia są zmyślone.

Rodzaj raportu	Raport bezpieczeństwa: pełny
Aplikacja	shop.example
Data wydania	2026-11-07
ID raportu	5a3b1e00-0000-4000-8000-000000000000
Wersja	1

Przygotował yof (yof.io)

Sprawdź ten raport
yof.io/verify/5a3b1e00-0000-4000-8000-000000000000
Odcisk (SHA-256): fb9ff4c2ff530d7b8b05f4622d7e389ab732e25f51676dc2cd21e1f34878fee5

Poufne: przygotowane dla właściciela shop.example

Metryka dokumentu

Historia wersji

v1 Wersja 1, wydana 2026-11-07

Zakres

shop.example: the outside of the app, the code of example-shop/storefront, attack results from the owner's CI.

Zlecający

Właściciel shop.example, przez aplikację yof (zamówienie 00000000).

Rozpowszechnianie

Poufne. Przygotowane dla właściciela shop.example, który może je udostępnić osobom, które go potrzebują, na przykład audytorom, ubezpieczycielom lub klientom. Każdy, kto ma kopię, może ją sprawdzić kodem QR.

Podpis

Algorytm	none (sample, not signed) (yof-report/v1 sample)
ID klucza	-
Odcisk (SHA-256)	fb9ff4c2ff530d7b8b05f4622d7e389ab732e25f51676dc2cd21e1f34878fee5
Wartość podpisu	Sample report: not signed and not in yof's register.

Jak sprawdzić

1. Zeskanuj kod QR z okładki albo otwórz adres wydrukowany pod nim. Strona sprawdza podpis yof w rejestrze.
2. Aby sprawdzić otrzymany plik, wybierz go na tej stronie: przeglądarka oblicza jego SHA-256 i porównuje go, plik nie jest wysyłany.
3. Przy podpisie Ed25519 każdy może też sprawdzić go offline kluczem publicznym yof, na podstawie ID raportu, wersji, daty wydania i skrótu treści.

   yof.io/verify/5a3b1e00-0000-4000-8000-000000000000

Czym jest ten raport, a czym nie jest

Ten raport to sprawdzenie shop.example w jednym momencie, według stanu na 2026-11-07. Obejmuje repozytorium, zależności i adresy na żywo wymienione w sekcji zakresu i nic więcej.

Powstał za pomocą automatycznych reguł i testów. Zależnie od zakresu yof przeczytał kod, sprawdził pakiety, przetestował działającą stronę i, za zgodą właściciela, przeprowadził testy ataków. Sekcja zakresu wymienia dokładnie, co zrobiono, a czego nie.

To materiał dowodowy wspierający Twoją pracę nad bezpieczeństwem i zgodnością. Nie jest certyfikacją, oceną zgodności ani poradą prawną i nie ocenia ładu organizacyjnego, polityk, procesów ani innych obowiązków, które nie są techniczne.

Wynik bez ustaleń nie oznacza, że aplikacja nie ma podatności. Automatyczne sprawdzenia nie znajdą każdej usterki. Słabości, o których nikt jeszcze nie wie (zero-day), błędy w logice biznesowej i wszystko poza zakresem mogą pozostać, a aplikacja może się zmienić po podanej dacie.

W granicach dozwolonych przez prawo odpowiedzialność yof za ten raport jest ograniczona do zapłaconej za niego ceny. Prawo konsumenckie w Twoim kraju może nie dopuszczać niektórych z tych ograniczeń; wtedy nie mają one do Ciebie zastosowania.

To tłumaczenie podpisanego raportu yof 5a3b1e00-0000-4000-8000-000000000000. Podpisany oryginał jest po angielsku; jego SHA-256 jest poniżej. Szczegóły techniczne (poprawki, kroki, odnośniki) są po angielsku.

Spis treści

Metryka dokumentu 2
Czym jest ten raport, a czym nie jest 2
Podsumowanie dla zarządu 5
Zakres 6
Ustalenia (5) 8
Załącznik A: Zaakceptowane ryzyka i powody 12
Załącznik B: Metodyka i narzędzia 13
Załącznik C: Słowniczek 14
Załącznik D: Podsumowanie wykazu składników (SBOM) 15
Załącznik E: Jak sprawdzić 16

Podsumowanie dla zarządu

Wynik ogólny

Wysokie

Wysokie: co najmniej jeden problem trzeba naprawić teraz.

Otwarte ustalenia według pilności

4	0	0	0	2
Napraw teraz	Napraw w tym	Warto poprawić	Do wiadomości	Zaakceptowane

Najważniejsze ryzyka, prostymi słowami

F-001 Pakiet instalowany przez Twoją aplikację ma znaną poważną lukę

Luka jest publiczna i ma opublikowaną poprawkę. Atakujący szukają aplikacji, które wciąż mają starą wersję.

Napraw teraz | package-lock.json:812 (example-shop/storefront at 0f1e2d3c4b5a)

F-002 Twój tajny klucz Stripe jest w kodzie strony

Każdy może go użyć, żeby przyjmować płatności, robić zwroty i czytać dane płatnicze Twoich klientów.

Napraw teraz | https://shop.example/assets/app.js

F-003 Każdy może czytać tabelę w Twojej bazie danych

Obcy może pobrać wiersze z tej tabeli, na przykład imiona, maile czy zamówienia klientów, bez logowania.

Napraw teraz | https://shop.example/

W zakresie

Z zewnątrz, Wewnątrz kodu, Testy ataków

Pokrycie: 45% części, które yof może zbadać.

Poza zakresem

- Strony za logowaniem nie były testowane: yof loguje się tylko podczas testów ataków uruchamianych przez właściciela.
- Przeczytano tylko pierwsze strony; reszta witryny nie została przeczytana.
- Asystent AI: nie testowano w tym raporcie.

Zakres

Pokrycie: 45% części, które yof może zbadać.
- Z zewnątrz: Przetestowane częściowo
- Wewnątrz kodu: Przetestowane w całości
- Strony za logowaniem: Nieprzetestowane
- Testy ataków: Przetestowane w całości
- Asystent AI: Nieprzetestowane
- Każda strona: Nieprzetestowane

Nieprzetestowane

- Strony za logowaniem nie były testowane: yof loguje się tylko podczas testów ataków uruchamianych przez właściciela.
- Przeczytano tylko pierwsze strony; reszta witryny nie została przeczytana.
- Asystent AI: nie testowano w tym raporcie.

Co sprawdzono, a czego nie

Tak wygląda zasięg tego raportu. Część, której nie sprawdzono, nigdy nie jest uznawana za w porządku.

Sprawdzono
- Z zewnątrz: Przetestowane częściowo
- Wewnątrz kodu: Przetestowane w całości
- Testy ataków: Przetestowane w całości

Nie sprawdzono i dlaczego
- Strony za logowaniem nie były testowane: yof loguje się tylko podczas testów ataków uruchamianych przez właściciela.
- Przeczytano tylko pierwsze strony; reszta witryny nie została przeczytana.
- Asystent AI: nie testowano w tym raporcie.

Nigdy nie obejmują tego automatyczne sprawdzenia
- Błędy w logice biznesowej, na przykład cena lub krok, który ktoś może pominąć.
- Słabości, o których nikt jeszcze nie wie (zero-day).
- Ludzie i procesy: phishing, hasła przekazywane ręcznie, kto ma dostęp.
- Wszystko poza zakresem wskazanym w tym raporcie i każda zmiana po jego dacie.

Jak pewny jest yof każdego ustalenia

Każde ustalenie ma jeden werdykt. Liczą się tylko ustalenia potwierdzone i te, które wymagają weryfikacji. Uwagi o wzmocnieniu i odrzucone ustalenia są wymienione osobno i nigdy nie są liczone.
- Potwierdzone: yof widział skutek albo ma dokładny dowód (na przykład klucz, który był udostępniany, albo zainstalowaną wersję wskazaną w biuletynie).
- Wymaga weryfikacji: Prawdopodobne, ale nie wykazane: yof widział wzorzec i nie mógł udowodnić skutku. Musi to potwierdzić człowiek albo głębszy test.
- Odrzucone: Obalone przez drugie, osobne sprawdzenie. Zachowane w dokumentacji, nie liczone.

Co oznacza każdy poziom
- Krytyczna: Tylko dla potwierdzonego ustalenia, które yof widział: dane lub kontrola należące do kogoś innego (dane innej osoby, cała tabela, klucz z pełnymi uprawnieniami udostępniany wszystkim) albo pakiet znany jako wykorzystywany lub złośliwy.
- Wysoka: Poważna szkoda jest prawdopodobna, jeśli ktoś to wykorzysta: potwierdzone, ograniczone ujawnienie albo prawdopodobny problem, którego yof nie mógł wykazać. Niepotwierdzone ustalenie może mieć najwyżej poziom Wysoka.
- Średnia: Ułatwia atak lub zwiększa szkody, pod warunkami takimi jak logowanie, kliknięcie użytkownika czy konkretne ustawienie.
- Niska: Sama w sobie mała szkoda albo dostępna tylko w narzędziach programistycznych i testowych.

- Info: Warto wiedzieć, samo w sobie bez szkody.

Ustalenia (5)

Napraw teraz | CVSS 9.3 | Otwarte

F-001 Pakiet instalowany przez Twoją aplikację ma znaną poważną lukę

Werdykt: Potwierdzone. Poziom: Wysoka. Dlaczego: Repozytorium zawiera dokładny dowód. yof nie widział tego w działającej aplikacji, więc najwyżej poziom Wysoka.

Luka jest publiczna i ma opublikowaną poprawkę. Atakujący szukają aplikacji, które wciąż mają starą wersję.
CVSS 4.0: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N
CWE: CWE-1395
Gdzie: package-lock.json:812 (example-shop/storefront at 0f1e2d3c4b5a)

Jak to zobaczyć:

1. Otwórz package-lock.json:812 (example-shop/storefront at 0f1e2d3c4b5a).
2. Oczekiwano zamiast tego: Zainstalowana wersja minimist to 1.2.6 lub nowsza.

Wpływ na firmę: Opublikowana słabość jednego z komponentów mogłaby zostać użyta przeciwko aplikacji.
Pakiet: minimist (npm)
Zainstalowana wersja: 1.2.5
Ostrzeżenia: GHSA-xvch-5gv4-984h https://github.com/advisories/GHSA-xvch-5gv4-984h, CVE-2021-44906 https://nvd.nist.gov/vuln/detail/CVE-2021-44906
Pierwsza poprawiona wersja: 1.2.6
Plik lockfile: package-lock.json:812
Wymieniony w: package.json
Poprawka: Zaktualizuj minimist z wersji 1.2.5 do 1.2.6 lub nowszej (ostrzeżenia: GHSA-xvch-5gv4-984h, CVE-2021-44906). Zaktualizuj package-lock.json za pomocą menedżera pakietów, a nie ręcznie, uruchom instalację i testy i sprawdź, czy aktualizacja wymaga zmian w kodzie.

Źródła:

GHSA-xvch-5gv4-984h (https://github.com/advisories/GHSA-xvch-5gv4-984h)
CVE-2021-44906 (https://nvd.nist.gov/vuln/detail/CVE-2021-44906)
CWE-1395: Dependency on Vulnerable Third-Party Component (https://cwe.mitre.org/data/definitions/1395.html)
OWASP A06:2021: Vulnerable and Outdated Components

Status: Otwarte. Pierwsze wykrycie: 2026-10-30

Napraw teraz | CVSS 9.3 | Otwarte

F-002 Twój tajny klucz Stripe jest w kodzie strony

Werdykt: Potwierdzone. Poziom: Krytyczna. Dlaczego: yof widział to w działającej aplikacji.

Każdy może go użyć, żeby przyjmować płatności, robić zwroty i czytać dane płatnicze Twoich klientów.
CVSS 4.0: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:N/VA:N/SC:H/SI:H/SA:H
CWE: CWE-798
Gdzie: https://shop.example/assets/app.js
Dowód (zamaskowany): sk_live_****9f2a

Jak to zobaczyć:

1. Otwórz https://shop.example/assets/app.js bez logowania.
2. Co zobaczył yof (zamaskowane): sk_live_****9f2a
3. Oczekiwano zamiast tego: W kodzie, który odwiedzający mogą pobrać, nie ma produkcyjnego tajnego klucza Stripe.

Wpływ na firmę: Ktoś mógłby zalogować się jako inna osoba albo zachować sesję, której nie powinien mieć.
Poprawka: Usuń tajny klucz Stripe (zaczyna się od sk_live_) z całego kodu przeglądarki i z każdej zmiennej środowiskowej, którą przeglądarka może odczytać. Twórz płatności wyłącznie na serwerze (w trasie API, akcji serwerowej lub Supabase Edge Function) i wysyłaj do przeglądarki tylko client secret danej płatności. Przeglądarka powinna widzieć wyłącznie klucz publikowalny (pk_live_). Nigdy nie wklejaj prawdziwego klucza do tego czatu, nigdy go nie wypisuj i nie wymyślaj klucza zastępczego. Daj mi znać, gdy kod nie będzie go już

zawierał, a wtedy utworzę nowy klucz w panelu dostawcy i usunę stary (stary był publiczny, więc trzeba go wymienić).

Źródła:

CWE-798: Use of Hard-coded Credentials (https://cwe.mitre.org/data/definitions/798.html)
OWASP A07:2021: Identification and Authentication Failures

Status: Otwarte. Pierwsze wykrycie: 2026-10-30

Napraw teraz | CVSS 8.7 | Otwarte

F-003 Każdy może czytać tabelę w Twojej bazie danych

Werdykt: Potwierdzone. Poziom: Wysoka. Dlaczego: yof widział to w działającej aplikacji.

Obcy może pobrać wiersze z tej tabeli, na przykład imiona, maile czy zamówienia klientów, bez logowania.

CVSS 4.0: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N

CWE: CWE-862

Gdzie: https://shop.example/

Dowód (zamaskowany): table orders: 3 rows readable without signing in

Jak to zobaczyć:

1. Otwórz https://shop.example/ bez logowania.
2. Co zobaczył yof (zamaskowane): table orders: 3 rows readable without signing in
3. Oczekiwano zamiast tego: Odwiedzający, którzy nie są zalogowani, nie mogą odczytać tej tabeli.

Wpływ na firmę: Ktoś mógłby zobaczyć lub zmienić dane, które należą do innej osoby.

Poprawka: Tabelę Supabase może odczytać każdy, kto ma publiczny klucz anon. Włącz Row Level Security dla tej tabeli (i każdej innej), a następnie dodaj polityki, aby zalogowana osoba mogła odczytywać tylko własne wiersze (na przykład using auth.uid() = user_id). Usuń każdą politykę, która pozwala roli anon wybierać wszystko. Napisz dla mnie SQL i wyjaśnij, na co pozwala każda polityka. Nigdy nie wklejaj do tego czatu prawdziwych danych klientów ani kluczy.

```
-- for each table the browser can reach
alter table public.your_table enable row level security;
create policy "own rows" on public.your_table
  for select using (auth.uid() = user_id);
```

Źródła:

CWE-862: Missing Authorization (https://cwe.mitre.org/data/definitions/862.html)
OWASP A01:2021: Broken Access Control

Status: Otwarte. Pierwsze wykrycie: 2026-10-30

Napraw teraz | CVSS 7.1 | Otwarte

F-004 Jeden użytkownik może czytać rekordy innego, zmieniając id

Werdykt: Potwierdzone. Poziom: Wysoka. Dlaczego: yof widział to w działającej aplikacji.

Konto testowe otworzyło rekord Twojego drugiego konta testowego, podając tylko jego id, więc prawdziwi użytkownicy mogą czytać nawzajem swoje dane.

CVSS 4.0: CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N

CWE: CWE-639

Gdzie: GET /api/orders/{id} (as user b)

Dowód (zamaskowany): 200 with another customer's order

Jak to zobaczyć:

1. Zaloguj się jako użytkownik testowy b.
2. Wyślij GET /api/orders/{id} z identyfikatorem należącym do innego użytkownika lub innej roli.
3. Co zobaczył yof (zamaskowane): 200 with another customer's order
4. Oczekiwano zamiast tego: Konto może otworzyć tylko własne rekordy; identyfikator innego konta daje odpowiedź 404 lub 403.

Wpływ na firmę: Ktoś mógłby zobaczyć lub zmienić dane, które należą do innej osoby.

Poprawka: W mojej aplikacji użytkownik B może odczytać rekord należący do użytkownika A, wpisując identyfikator rekordu A w adresie (lub widzi go na liście). Na serwerze każde zapytanie o rekord lub listę musi być filtrowane według zalogowanego użytkownika (lub jego przestrzeni roboczej), na przykład `where

id = :id and owner_id = :currentUser`, i odpowiadać 404, gdy rekord nie należy do niego. Jeśli baza danych obsługuje row-level security, włącz ją z polityką, która pozwala tylko właścicielowi. Nie polegaj na tym, że identyfikatory są trudne do odgadnięcia.

Źródła:

CWE-639: Authorization Bypass Through User-Controlled Key (https://cwe.mitre.org/data/definitions/639.html)
OWASP A01:2021: Broken Access Control
OWASP API1:2023: Broken Object Level Authorization

Status: Otwarte. Pierwsze wykrycie: 2026-10-30

Napraw w tym tygodniu | CVSS 5.1 | Naprawione

F-005 Twoja domena nie ma reguły dla fałszywych maili

Werdykt: Wymaga weryfikacji. Poziom: Średnia. Dlaczego: Sprawdzenie nie zachowało dowodu potrzebnego do potwierdzenia.

Dostawcy poczty nie wiedzą, co robić z wiadomościami udającymi Ciebie, więc wiele z nich dociera.
CVSS 4.0: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N
CWE: CWE-290
Gdzie: https://shop.example/

Jak to zobaczyć:

1. Sprawdź rekordy DNS TXT domeny (na przykład: dig TXT shop.example).
2. Oczekiwano zamiast tego: Domena publikuje politykę DMARC quarantine lub reject.

Wpływ na firmę: Jedno z ustawień zostawia lukę, którą łatwo znaleźć.
Poprawka: Domena nie ma rekordu DMARC. Dodaj rekord TXT, którego nazwa (host) to dokładnie _dmarc, a nie @ (rekord DMARC w katalogu głównym domeny jest ignorowany), zaczynając od monitorowania: v=DMARC1; p=none; rua=mailto:<an address you read> (adres, który czytasz). Po kilku tygodniach czystych raportów przejdź na p=quarantine, a potem na p=reject.

```
_dmarc.shop.example.  TXT  "v=DMARC1; p=quarantine; rua=mailto:dmarc@shop.example"
```

Źródła:

CWE-290: Authentication Bypass by Spoofing (https://cwe.mitre.org/data/definitions/290.html)
OWASP A05:2021: Security Misconfiguration

Status: Naprawione. Pierwsze wykrycie: 2026-10-30

Tylko otwarte ustalenia, według werdyktu: potwierdzone (4), wymaga weryfikacji (0). Osobno: uwagi o wzmocnieniu (1), odrzucone (0).

Uwagi o wzmocnieniu (nie liczone jako ustalenia)

Dobre praktyki, które dodałyby warstwę ochrony. yof nie wykazał ryzyka, więc to nie są ustalenia i nie wchodzą do żadnej liczby.

- H-001 Przeglądarki nie wiedzą, że zawsze mają używać https: https://shop.example/

Dodaj nagłówek Strict-Transport-Security do każdej odpowiedzi, na przykład: Strict-Transport-Security: max-age=31536000; includeSubDomains. Ustaw go w konfiguracji nagłówków hostingu lub frameworka.

```
// vercel.json (or the same header in your host's settings)
{
  "headers": [
    { "source": "/(.*)", "headers": [ { "key": "Strict-Transport-Security", "value": "max-a
ge=63072000; includeSubDomains" } ] }
  ]
}
```

- H-002 Żadna reguła nie ogranicza, jakie skrypty mogą działać: https://shop.example/

Dodaj nagłówek Content-Security-Policy, który określa, skąd mogą pochodzić skrypty, style, obrazy i połączenia. Zacznij w trybie Report-Only, sprawdź, czy aplikacja nadal działa, a potem go egzekwuj.

```
// vercel.json (or the same header in your host's settings)
{
  "headers": [
    { "source": "/(.*)", "headers": [ { "key": "Content-Security-Policy", "value": "default
-src 'self'; frame-ancestors 'self'; object-src 'none'; base-uri 'self'" } ] }
```

```
    ]
}
```

Załącznik A: Zaakceptowane ryzyka i powody

Właściciel przejrzał te ustalenia i zdecydował się zaakceptować ryzyko z podanego powodu. Nie są liczone jako otwarte i są wymienione tutaj, aby nic nie zostało ukryte.

H-002 Żadna reguła nie ogranicza, jakie skrypty mogą działać

Gdzie: https://shop.example/
Powód: The payment widget needs inline scripts until its next version; we review this in January.
Zaakceptował(a): owner@shop.example, 2026-11-05
Zaakceptowano w: aplikacji yof

Testowy tajny klucz Stripe jest w kodzie strony

Gdzie: tests/fixtures/payments.ts
Powód: A Stripe test key in a test fixture; it cannot take real payments.
Zaakceptował(a): Sam (developer), 2026-11-02
Zaakceptowano w: repozytorium (.yof/accepted.json)

Załącznik B: Metodyka i narzędzia

yof uruchomił własne reguły i testy: kontrolę działającej aplikacji z zewnątrz, odczyt połączonego repozytorium i jego pakietów, własne testy ataków właściciela oraz, za zgodą, stałe próby asystenta AI. Każda kontrola stosuje opublikowaną metodę podaną obok niej.

Żaden zewnętrzny skaner ani model AI nie oceniał ustaleń. Dowody są maskowane przed zapisaniem, a każde zapytanie yof jest na liście dowodów.

Przywołane metody i standardy

ASVS V14.1, ASVS V14.2.1, ASVS V14.3, ASVS V14.4, ASVS V2.10.4, ASVS V4.1, ASVS V5, ASVS V6.4.1, OWASP A06:2021, OWASP API Top 10 2023, RFC 6376 (DKIM), RFC 7208 (SPF), RFC 7489 (DMARC), WSTG-ATHN-04, WSTG-ATHZ-01, WSTG-ATHZ-02, WSTG-ATHZ-04, WSTG-CLNT-09, WSTG-CONF-03, WSTG-CONF-04, WSTG-CONF-05, WSTG-CONF-06, WSTG-CONF-07, WSTG-CONF-12, WSTG-INFO-05, WSTG-INPV, WSTG-SESS-01

Załącznik C: Słowniczek

Ustalenie
Coś, co yof zobaczył i co mogłoby wpuścić atakującego lub ujawnić dane, wraz z miejscem i sposobem sprawdzenia.

Pilność
Prosta kolejność prac yof: napraw teraz, w tym tygodniu, warto poprawić, dla informacji.

CVSS 4.0
Powszechny system oceny, jak poważna jest słabość, od 0 do 10, z wektorem wyjaśniającym ocenę.

CWE
Powszechna lista rodzajów słabości; każde ustalenie podaje swój rodzaj.

Pokrycie
Jaka część z sześciu obszarów, które yof może zbadać, została sprawdzona w tym raporcie.

Zaakceptowane ryzyko
Ustalenie, które właściciel postanowił zostawić, z powodem. Jest wymienione, ale nie liczy się jako otwarte.

SBOM
Wykaz składników oprogramowania: lista pakietów, z których zbudowana jest aplikacja.

Odcisk
Skrót SHA-256 treści raportu. Każda zmiana raportu go zmienia.

Załącznik D: Podsumowanie wykazu składników (SBOM)

Pakiety w pliku SBOM dołączonym do raportu: 640. Ze znanym ostrzeżeniem: 1.

- example-shop/storefront @ 0f1e2d3c4b5a
- npm: 640
- package-lock.json

SHA-256 5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a5a

Załącznik E: Jak sprawdzić

1. Zeskanuj kod QR z okładki albo otwórz adres wydrukowany pod nim. Strona sprawdza podpis yof w rejestrze.
2. Aby sprawdzić otrzymany plik, wybierz go na tej stronie: przeglądarka oblicza jego SHA-256 i porównuje go, plik nie jest wysyłany.
3. Przy podpisie Ed25519 każdy może też sprawdzić go offline kluczem publicznym yof, na podstawie ID raportu, wersji, daty wydania i skrótu treści.

- Zeskanuj kod QR lub otwórz https://yof.io/verify/5a3b1e00-0000-4000-8000-000000000000. Strona sprawdza podpis yof i pokazuje tylko to, czym właściciel raportu zdecydował się podzielić.

Sprawdź ten raport
yof.io/verify/5a3b1e00-0000-4000-8000-000000000000
Odcisk (SHA-256): fb9ff4c2ff530d7b8b05f4622d7e389ab732e25f51676dc2cd21e1f34878fee5

Ograniczenia

- Raport pokazuje dowody tego, co przetestowano we wskazanych dniach. Nie jest obietnicą co do niczego, czego nie przetestowano, ani późniejszych zmian.
- Dowody są zamaskowane: klucze, tokeny, hasła i adresy e-mail nigdy nie są drukowane w całości.