securitum

Raport z testów bezpieczeństwa

PRZEDMIOT PRAC

Test bezpieczeństwa infrastruktury (WAN)

DATA WYKONANIA PRAC

12.02.2024 – 22.02.2024

MIEJSCE WYKONANIA PRAC

Gdańsk, Polska

AUTOR

Maciej Szymczak

WERSJA DOKUMENTU

1.0

securitum

Podsumowanie wykonanych prac

Niniejszy raport jest podsumowaniem testów bezpieczeństwa przeprowadzonych przez firmę Securitum. Przedmiotem testów była infrastruktura zewnętrzna (WAN) firmy [redacted], dostępna pod niżej wymienionymi adresami IP:

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

Testy urządzeń sieciowych oraz serwerów zostały przeprowadzone w podejściu *blackbox*, tj. bez posiadania dodatkowych informacji na temat testowanej infrastruktury.

Najistotniejsze znalezione podatności to:

- Podatności typu Denial of Service w usługach WWW.

W trakcie prac szczególny nacisk położono na podatności mające lub mogące mieć negatywny wpływ na poufność, integralność oraz dostępność przetwarzanych danych.

Testy bezpieczeństwa przeprowadzono zgodnie z wewnętrznymi metodykami przeprowadzania testów bezpieczeństwa firmy Securitum.

W ramach prac wykorzystano podejście polegające na testach manualnych, które zostały wsparte szeregiem narzędzi automatycznych, m.in. Burp Suite Professional, Feroxbuster, Nessus Professional, nmap.

Podatności zostały szczegółowo opisane w dalszej części raportu.

Klasyfikacja podatności

Podatności zostały sklasyfikowane w pięciostopniowej skali odzwierciedlającej zarówno prawdopodobieństwo znalezienia podatności, jak i istotność skutków jej wykorzystania. Poniżej zawarto krótki opis każdego z poziomów istotności:

- CRITICAL (podatność krytyczna) – wykorzystanie podatności umożliwia przejęcie pełnej kontroli nad serwerem lub urządzeniem sieciowym albo pozwala uzyskać dostęp (w trybie zapisu i/lub odczytu) do danych o dużym poziomie poufności i istotności. Zazwyczaj podatność jest też łatwa do wykorzystania, tj. nie wymaga od napastnika posiadania dostępu do systemów, które są trudne do zdobycia, lub przeprowadzania ataków socjotechnicznych. Podatności oznaczone jako CRITICAL powinny być naprawione bezzwłocznie, jeśli występują na środowisku produkcyjnym.

- HIGH (podatność o wysokim poziomie istotności) – wykorzystanie podatności pozwala na uzyskanie dostępu do wrażliwych informacji (podobnie jak przy poziomie krytycznym), jednak może wcześniej wymagać spełnienia pewnych warunków (np. posiadania konta użytkownika w wewnętrznym systemie) w celu praktycznego wykorzystania. Alternatywnie: podatność może zostać w łatwy sposób wykorzystana, jednak ograniczone są jej skutki.

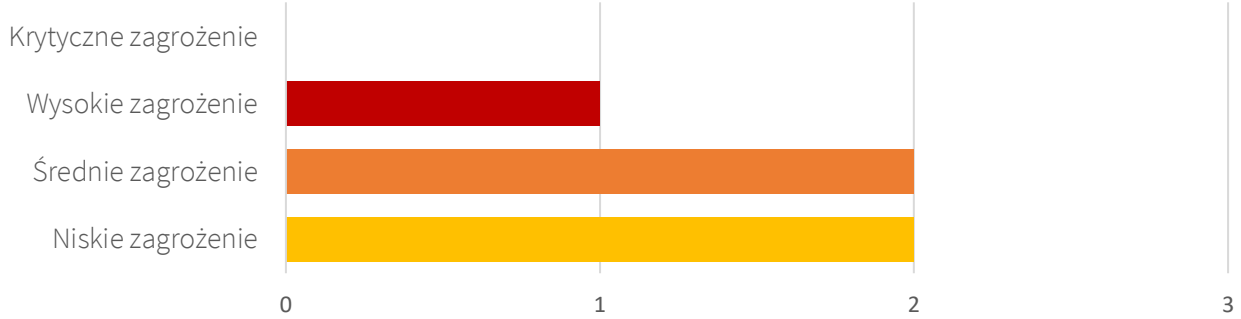
- MEDIUM (podatność o średnim poziomie istotności) – wykorzystanie podatności może zależeć od zewnętrznych czynników (np. wymaga przekonania użytkownika do kliknięcia w łącze) lub może wymagać trudnych do spełnienia warunków. Ponadto wykorzystanie podatności zazwyczaj umożliwia dostęp tylko do ograniczonej ilości danych lub do danych o mniejszym poziomie istotności.

- LOW (podatność o niskim poziomie istotności) – wykorzystanie podatności ma niewielki bezpośredni wpływ na bezpieczeństwo przedmiotu testów lub wymaga bardzo trudnych warunków do spełnienia (np. fizyczny dostęp do serwera).

- INFO (ogólne zalecenia lub informacja) – punkty oznaczone poziomem INFO nie są podatnościami bezpieczeństwa. Wskazują jednak dobre praktyki, których zastosowanie pozwala zwiększyć ogólny poziom bezpieczeństwa przedmiotu testów. Alternatywnie: zwracają uwagę na pewne rozwiązania (np. architektoniczne), pozwalające uszczelnić przedmiot testów.

Zestawienie statystyczne

Poniżej znajduje si ę zestawienie statystyczne znalezionych podatności:



Ponadto zgłoszono dwa punkty informacyjne.

Spis treści

Historia zmian

Data dokumentu	Wersja	Opis zmiany
22.02.2024	1.0	Utworzenie dokumentu.

Podatności w infrastrukturze

[HIGH] SECURITUM-24887-001: Serwer WWW podatny na atak *Denial-of-Service*

OPIS

Podczas audytu ustalono, że zainstalowana wersja serwera WWW Apache to 2.4.57. Wersja ta posiada znane podatności, m.in.:

- CVE-2023-45802,
- CVE-2023-43622,
- CVE-2023-31122,

które zostały szerzej opisane na stronie Apache:

- https://httpd.apache.org/security/vulnerabilities_24.html

Podatności te, choć pozornie niegroźne, mogą wpłynąć na niedostępność usług. W kolejnej sekcji pokazano przykładowe wykorzystanie takiego ataku, który nie tylko jest skuteczny, ale również trudny do wykrycia – w logach serwera nie odkładają się żadne jednoznaczne informacje sugerujące, że jest to atak.

WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Brak, usługa dostępna jest z Internetu bez żadnych ograniczeń.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Po instalacji serwera Apache jest on standardowo podatny na ataki typu DoS (ang. *Denial-of-Service*). Polityka twórców oprogramowania Apache HTTP Server jest taka, że dostarczony serwer jest standardowej konfiguracji nie zawiera żadnych dodatkowych zabezpieczeń przed atakami, potrafi serwować treści, ale to na administratorze spoczywa obowiązek dalszej poprawnej konfiguracji usługi, w tym m.in. zabezpieczenie przed atakami.

W poniższym przykładzie wykorzystano nie tylko brak poprawnej konfiguracji serwera, która szerzej znana jest jako podatność Slowloris (CVE-2007-6750), ale również połączono ją z podatnością CVE-2023-43622.

Atak dotyczy modułu odpowiedzialnego za obsługę protokołu HTTP/2 w Apache. Badacze odkryli, że przesłanie dużej ilości zapytań z niepoprawnie ustawioną początkową wielkością ramki HTTP/2, powoduje, że serwer WWW zachowuje się podobnie jak w przypadku ataku Slowloris – tj. połączenia te nie są poprawnie zamykane, co z kolei prowadzi do szybkiego wyczerpania puli dostępnych gniazd, a tym samym niedostępność usługi WWW.

Z uwagi na brak publicznie dostępnego eksploita dot. podatności CVE-2023-43622, poniżej przedstawiono tylko fragment narzędzia oraz przebieg ataku. Testowanym serwerem był kontener z uruchomionym oficjalnym obrazem serwera Apache HTTP Server w wersji 2.4.57 (`httpd:2.4.57-alpine`), dostępny pod adresem https://127.0.0.1:8443/.

Aby wysłać zmodyfikowaną komunikację HTTP/2 można posłużyć się narzędziem Scapy. Fragment zdefiniowanej komunikacji HTTP/2 przedstawiono poniżej.

```
H2Frame()/H2SettingsFrame()/H2Setting(id=H2Setting.SETTINGS_INITIAL_WINDOW_SIZE, value=0)
```

Wysłanie dużej ilości zapytań, jednocześnie podtrzymując nawiązane połączenia, powoduje szybką niedostępność serwera, co zostało przedstawione na poniższym przykładzie:

```
$ python3 CVE-2023-43622_poc.py
.................................................. 50 connections…
..................................................100 connections…
..................................................150 connections…
..................................................200 connections…
..................................................250 connections…
..................................................300 connections…
..................................................350 connections…
..................................................400 connections…
..................................................450 connections…
..................................................500 connections…
..................................................550 connections…
..................................................600 connections…
..................................................650 connections…
..................................................700 connections…
..................................................750 connections…
..................................................800 connections…
..................................................850 connections…
..................................................900 connections…
............^C
Killed. 912 connections were used.
```

Po nawiązaniu 912 połączeń serwer przestał akceptować nowe, co zostało potwierdzone równoległym wywołaniem narzędzia `curl` (parametr `-k` został podany celem zaakceptowania samopodpisanego certyfikatu (ang. *self-signed*)):

```
# curl -k https://127.0.0.1:8443/
curl: (28) SSL connection timeout
```

Z uwagi na produkcyjny charakter testowanej infrastruktury, wszelkie testy oraz powyższa symulacja zostały przeprowadzone w kontrolowanym środowisku.

LOKALIZACJA

Podatne serwery:

- [redacted] (tcp/80, tcp/443)
- [redacted] (tcp/80, tcp/443)
- [redacted] (tcp/80, tcp/443)
- [redacted] (tcp/80, tcp/443)
- [redacted] (tcp/80, tcp/443)

REKOMENDACJA

Zaleca się aktualizację oprogramowania do najnowszej, stabilnej wersji. Dodatkowo należy zaimplementować ochronę przed innymi atakami Denial-of-Service, które zostały szczegółowo opisane w oficjalnej dokumentacji:

- https://httpd.apache.org/docs/trunk/misc/security_tips.html#dos

Warto rozważyć również migrację z serwera Apache na np. nginx lub Caddy, które to w podstawowej konfiguracji są znacznie lepiej przystosowane do obsługi nadmiarowego ruchu sieciowego.

[MEDIUM] SECURITUM-24887-002: Problemy z certyfikatami TLS

OPIS

Podczas audytu zidentyfikowano wiele usług wykorzystujących protokółTLS. O ile koncepcja szyfrowania ruchu jest słuszna, o tyle wykorzystywanie w tym celu niezaufanych lub nieważnych certyfikatów pozbawia ją sensu.

Klienci korzystający z usług zmuszeni są zaakceptować tzw. „wyjątki" w przeglądarkach (lub innych klientach), co oznacza, że nie są w stanie zweryfikować poprawności certyfikatu, więc akceptują bez dodatkowej weryfikacji to, co wyświetli się na ekranie. Brak możliwości zweryfikowania poprawności certyfikatu prowadzić może do potencjalnego naruszenia bezpieczeństwa. W przypadku skutecznego ataku MitM (ang. *Man-in-the-Middle*) atakujący może celowo wykorzystać taką politykę firmy (tj. pracy na niezaufanych certyfikatach), aby nakłonić pracowników do „ponownego" zaakceptowania wyjątku, a tym samym uzyskać dostęp do poufnych danych.

Więcej informacji:

- https://pl.wikipedia.org/wiki/Atak_man_in_the_middle

WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Konieczne jest przeprowadzenie skutecznego ataku Man-in-the-Middle.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Do zweryfikowania szczegółów dot. obsługi TLS oraz poprawności wykorzystanych certyfikatów użyto narzędzi `testssl.sh` oraz `openssl`. Wnioski z pracy narzędzi przedstawiono poniżej:

[redacted] (tcp/443) – wygaśnięty (ang. *expired*) certyfikat:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted] 2021 GMT
```

[redacted] (tcp/443) – samopodpisany (ang. *self-signed*) certyfikat:

```
|-Subject : O=Acme Co/CN=Kubernetes Ingress Controller Fake Certificate
|-Issuer : O=Acme Co/CN=Kubernetes Ingress Controller Fake Certificate
```

[redacted] (tcp/443) – samopodpisany certyfikat:

```
|-Subject : O=Acme Co/CN=Kubernetes Ingress Controller Fake Certificate
|-Issuer : O=Acme Co/CN=Kubernetes Ingress Controller Fake Certificate
```

[redacted] (tcp/21) – wygaśnięty certyfikat:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted] 2021 GMT
```

[redacted] (tcp/443) – wygaśnięty certyfikat:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted] 2021 GMT
```

[redacted] (tcp/443) – wygaśnięty certyfikat CA:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
```

```
|-Not After : [redacted] 2021 GMT
```

[redacted] (tcp/443) – wygaśnięty certyfikat CA:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted]2021 GMT
```

[redacted] (tcp/443) – niekompletny zestaw certyfikatów (ang. *wrong chain order*):

```
|-Subject : CN=docker.[redacted].eu
|-Issuer : C=US/O=Let's Encrypt/CN=R3
```

[redacted] (tcp/443) - niekompletny łańcuch certyfikatów:

```
|-Subject : CN=servicedesk.[redacted].eu
|-Issuer : C=US/O=Let's Encrypt/CN=R3
```

[redacted] (tcp/443) – wygaśnięte certyfikaty:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted] 2021 GMT
```

oraz

```
|-Subject : CN=*.kube.[redacted].pl
|-Not After : [redacted] 2024 GMT
```

Warto również wspomnieć o niepoprawnej konfiguracji – tj. serwer wspiera przestarzały i niebezpieczny protokół TLS v1.0.

[redacted] (tcp/8895) – samopodpisany certyfikat:

```
|-Subject : C=PL/ST=[redacted]/O=[redacted]/CN=[redacted]
|-Issuer : C=PL/ST=[redacted]/O=[redacted]/CN=[redacted]
```

[redacted] (tcp/443) – wygaśnięty certyfikat:

```
|-Subject : O=Digital Signature Trust Co./CN=DST Root CA X3
|-Not After : [redacted] 2021 GMT
```

LOKALIZACJA

[redacted] (tcp/443)
[redacted] (tcp/443)
[redacted] (tcp/443, tcp/8895)
[redacted] (tcp/443)
[redacted] (tcp/443)
[redacted] (tcp/443)
[redacted] (tcp/443)
[redacted] (tcp/443)
[redacted] (tcp/21, tcp/443)
[redacted] (tcp/443)

REKOMENDACJA

Zaleca się wykorzystywanie wyłącznie zaufanych certyfikatów TLS. Ponadto usługi (w tym wypadku – serwery WWW oraz FTP) powinny być skonfigurowane w taki sposób, aby wspierać protokół TLS v1.3, z ewentualnym wsparciem również dla TLS v1.2. Należy również zwrócić uwagę na dodatkowe parametry, zależnie od serwera WWW.

Dobrym źródłem przykładowych konfiguracji jest narzędzie przygotowane przez Mozillę:

- https://ssl-config.mozilla.org/

[MEDIUM] SECURITUM-24887-003: Serwer FTP podatny na atak *Denial-of-Service*

OPIS

W testowanym środowisku zidentyfikowano serwer FTP, oprogramowanie vsftpd w wersji 3.0.3, która posiada znaną podatność, wynikającą m.in. z niepoprawnej konfiguracji. Podobnie jak przy podatności opisanej w punkcie SECURITUM-24887-001, usługa ta również może zostać zaatakowana w analogiczny sposób, powodując niedostępność usługi.

Autor oprogramowania poprawił w najnowszym wydaniu również kilka problemów związanych z obsługą TLS, o czym szczegółowo przeczytać można na oficjalnej stronie vsftpd:

- https://security.appspot.com/vsftpd.html

WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Brak, usługa dostępna jest z Internetu.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Podatność została zidentyfikowana na podstawie wersji oprogramowania, którą odczytać można łącząc się bezpośrednio do usługi, np. z wykorzystaniem polecenia `nc`:

```
$ nc [redacted] 21
220 (vsFTPd 3.0.3)
```

LOKALIZACJA

[redacted] (tcp/21)

REKOMENDACJA

Zaleca się aktualizację do najnowszej, stabilnej wersji vsftpd. Niezbędna jest również poprawna konfiguracja (m.in. limit połączeń per IP, itp.), co zostało szczegółowo opisane w dokumentacji:

- https://security.appspot.com/vsftpd/vsftpd_conf.html

Opcje, na które należy zwrócić uwagę to `connect_timeout`, `data_connection_timeout` oraz `accept_timeout`.

[LOW] SECURITUM-24887-004: Niewspierane oprogramowanie Jetty (część Sonatype Nexus Repository)

OPIS

Podczas audytu zidentyfikowano nieaktualne oprogramowanie Jetty w wersji 9.4.51.v20230217. Na podstawie charakterystycznej odpowiedzi HTTP ustalono, że serwer Jetty jest częścią oprogramowania Sonatype Nexus Repository. Nieaktualny komponent wskazuje równocześnie na nieaktualne oprogramowanie główne – co najmniej od listopada 2023 roku.

Serwer Jetty w wersji 9.4.x stracił wsparcie w czerwcu 2022 roku:

- https://github.com/jetty/jetty.project/issues/7958

Powyższe oznacza, że od tego czasu komponent ten nie otrzymuje aktualizacji ani łatek bezpieczeństwa. Brak regularnych aktualizacji prowadzi do ekspozycji na znane ataki. Warto nadmienić, że wielu producentów (szczególnie zamkniętego oprogramowania (ang. *closed source*)) często nie pisze wprost o załataniu podatności w danym wydaniu, dlatego szczególnie ważne jest utrzymywanie możliwie najnowszej, stabilnej wersji.

WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Brak, usługa dostępna jest z Internetu.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Na podstawie poniższej odpowiedzi HTTP ustalono wersję oprogramowania:

```
$ curl -ik https://[redacted]
HTTP/1.1 400 Bad Request
Date: Thu, 22 Feb 2024 11:03:44 GMT
Server: Jetty(9.4.51.v20230217)

[…]

<title>Error 400 Not a Docker request</title>
</head>
<body><h2>HTTP ERROR 400 Not a Docker request</h2>
<table>
<tr><th>URI:</th><td>/</td></tr>
<tr><th>STATUS:</th><td>400</td></tr>
<tr><th>MESSAGE:</th><td>Not a Docker request</td></tr>
<tr><th>SERVLET:</th><td>-</td></tr>
</table>
<hr/><a href="https://eclipse.org/jetty">Powered by Jetty:// 9.4.51.v20230217</a><hr/>
[…]
```

Jednocześnie charakterystyczna odpowiedź `HTTP ERROR 400 Not a Docker request` pozwoliła na ustalenie, że prawdopodobnie jest to odpowiedź z systemu Sonatype Nexus Repository w wersji niższej niż 3.62.0 (wydanej w listopadzie 2023 roku), co potwierdza notatka towarzysząca temu wydaniu:

3.62.0	November 7, 2023	• New Cleanup Preview Experience for Pro Customers Using PostgreSQL
		• New Combined Helm Chart for AWS, Azure, or On-Premises High Availability Deployments
		• Azure HA Performance Data
		• Support Zip Improvements
		  ◦ Generate zip for all nodes
		  ◦ Download link persists
		• Expanded Audit Logging
		  ◦ Include records for "Clear Cache" and "Change (server) order" LDAP events.
		  ◦ Added logging for when you create, update, or delete a routing rule.
		• Upgraded Jetty from version 9.4.51.v20230217 to version 9.4.53.v20231009

LOKALIZACJA

[redacted] (tcp/443)

REKOMENDACJA

Zaleca się aktualizację oprogramowania do najnowszej, stabilnej oraz wspieranej przez dostawce wersji.

[LOW] SECURITUM-24887-005: Nieaktualne oprogramowanie oraz nadmiarowe nagłówki HTTP

OPIS

Audyt wykazał, że część usług (serwerów WWW) ujawnia swoją wersję oprogramowania w nagłówkach HTTP. Sam fakt prezentowania wersji oprogramowania nie jest bezpośrednim zagrożeniem dla infrastruktury, dopóki wersja ta jest aktualna. W testowanym przypadku, część usług nie jest w najnowszych wersjach oraz posiada znane podatności.

WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Zależnie od podatności, natomiast warto mieć na uwadze, że wszystkie wskazane usługi dostępne są z Internetu.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Niżej wymienione usługi zdradzają rodzaj oraz wersję stosowanego oprogramowania.

[redacted] (tcp/80, tcp/443) – nieaktualna wersja serwera Apache:

```
Apache/2.4.56 (Debian)
```

[redacted] (tcp/80, tcp/443) – nieaktualna wersja serwera Apache:

```
Apache/2.4.57 (Debian)
```

[redacted] (tcp/80) – nieaktualna wersja serwera Apache:

```
Apache/2.4.57 (Debian)
```

[redacted] (tcp/443) – omówiony w punkcie SECURITUM-24887-004:

```
Jetty(9.4.51.v20230217)
```

[redacted] (tcp/443) – niewspierana wersja serwera nginx:

```
nginx/1.22.1
```

[redacted] (tcp/80):

```
Microsoft-IIS/10.0
```

[redacted] (tcp/443):

```
Microsoft-HTTPAPI/2.0
```

[redacted] (tcp/80, tcp/443):

```
Microsoft-HTTPAPI/2.0
```

[redacted] (tcp/80):

```
Apache/2.4.57 (Debian)
```

[redacted] (tcp/80, tcp/443) niewspierana wersja serwera nginx:

```
nginx/1.18.0
```

[redacted] (tcp/80/www) - nieaktualna wersja serwera Apache:

```
Apache/2.4.56 (Debian)
```

[redacted] (tcp/80, tcp/443) – niewspierana wersja serwera nginx:

```
nginx/1.22.1
```

LOKALIZACJA

Adresy IP oraz numery portów zostały wylistowane w sekcji Szczegóły Techniczne.

REKOMENDACJA

Zaleca się przeprowadzanie regularnych aktualizacji systemowych, wraz z zainstalowanymi usługami. Aktualizacje powinny być realizowane zgodnie z wypracowaną polityką (harmonogramem) aktualizacji.

Punkty informacyjne

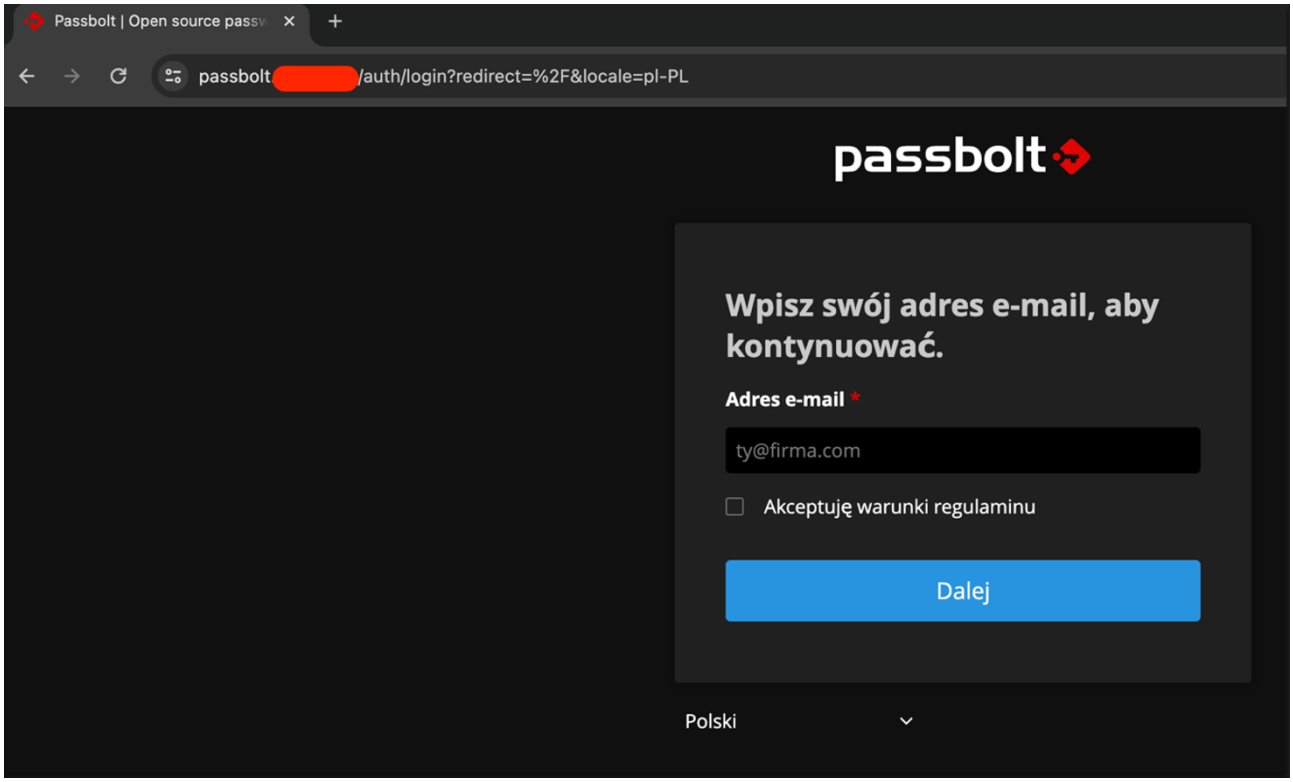
[INFO] SECURITUM-24887-006: Usługa współdzielenia poświadczeń dostępna z Internetu

OPIS

Produkt Passbolt jest systemem współdzielenia poświadczeń. Jest to krytyczne, z punktu widzenia przetwarzanych danych, miejsce dla organizacji. Oprogramowanie tego typu nie powinno być eksponowane bezpośrednio do Internetu, jeżeli nie jest to absolutnie konieczne.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Poniżej przedstawiono zrzut ekranu panelu logowania, który zabezpiecza dostęp do menadżera haseł:



LOKALIZACJA

[redacted] (tcp/443, tcp/80)

REKOMENDACJA

Zaleca się ukrycie panelu logowania za dodatkowym zabezpieczeniem – np. mTLS (ang. *mutual TLS*) lub ew. VPN.

[INFO] SECURITUM-24887-007: Standardowe strony powitalne serwerów WWW

OPIS

Niepoprawnie skonfigurowane usługi wyświetlają standardowe strony powitalne, co pozwala potencjalnemu atakującemu lepiej poznać środowisko swojej ofiary.

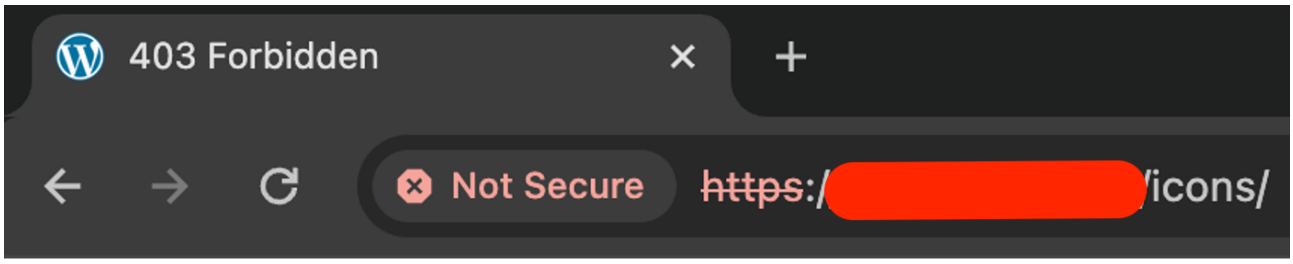
WARUNKI NIEZBĘDNE DO WYKORZYSTANIA PODATNOŚCI

Brak, usługi są dostępne z Internetu.

SZCZEGÓŁY TECHNICZNE (PROOF OF CONCEPT)

Na poniższych zrzutach ekranu przedstawiono zidentyfikowane strony powitalne serwerów WWW.

[redacted] (tcp/443) - standardowy *vhost* serwera Apache został zidentyfikowany za pomocą domyślnego katalogu `/icons` – błąd HTTP 403 Forbidden oznacza, że katalog ten jest zablokowany, ale równocześnie potwierdza istnienie takiego *vhosta*:

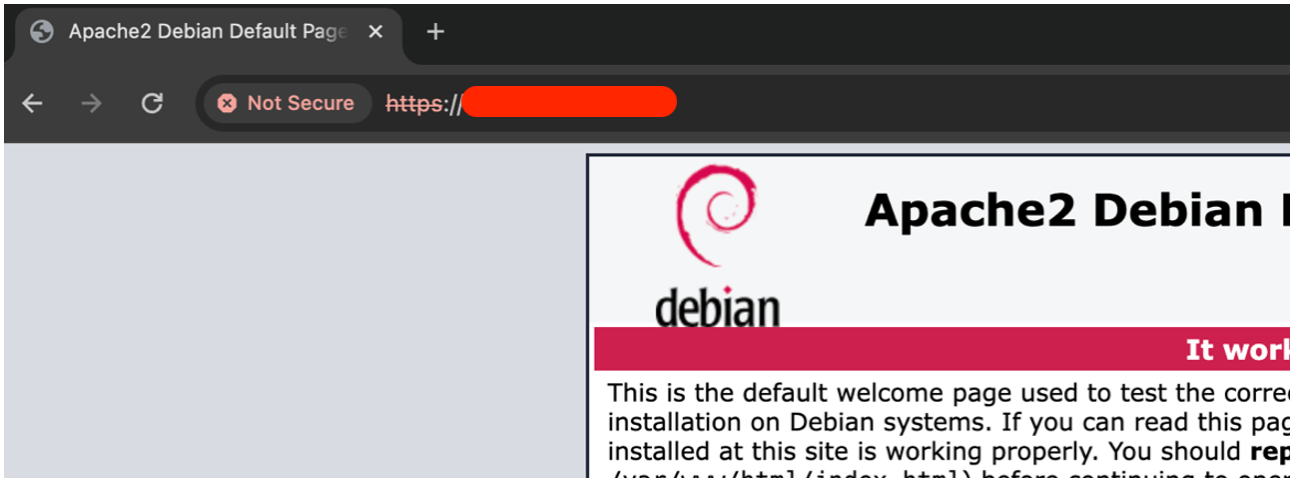


Forbidden

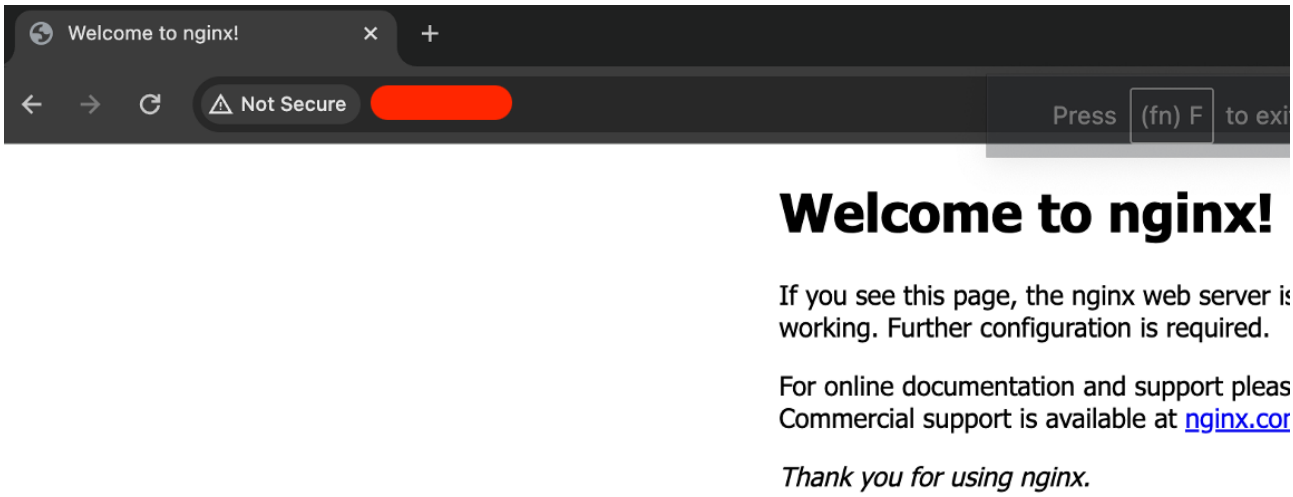
You don't have permission to access this resource.

Apache/2.4.56 (Debian) Server at [redacted] *Port 443*

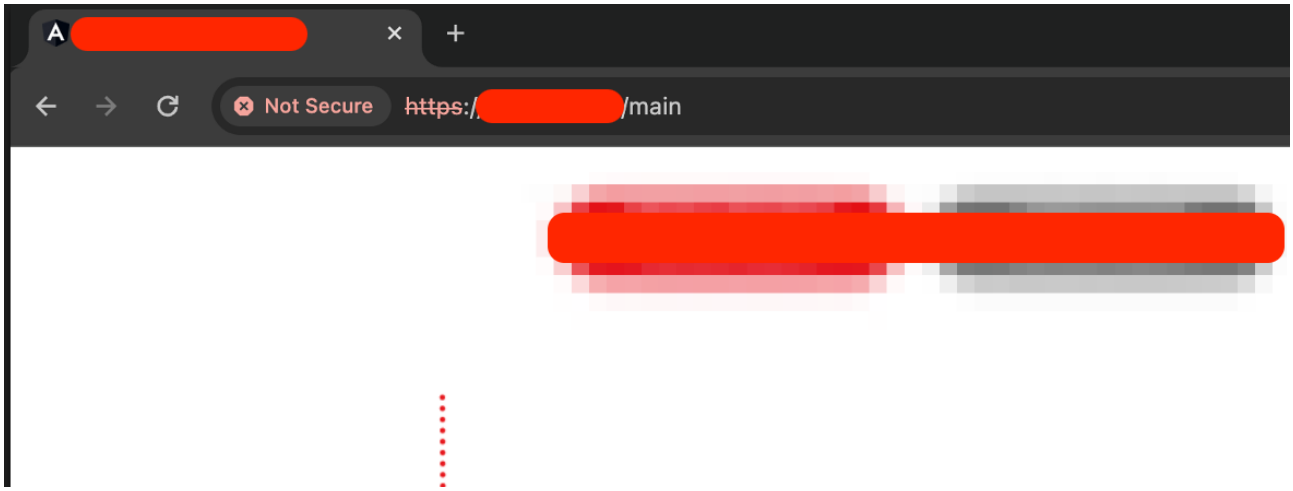
[redacted] (tcp/443) - łącząc się bezpośrednio pod adres IP (bez używania nazwy domenowej), serwer wyświetla standardową stronę Apache:



[redacted] (tcp/80) - łącząc się bezpośrednio pod adres IP (bez używania nazwy domenowej), serwer wyświetla standardową stronę nginx:



[redacted] (tcp/443) - łącząc się bezpośrednio pod adres IP (bez używania nazwy domenowej), serwer wyświetla aplikację WWW:



LOKALIZACJA

- [redacted] (tcp/443)
- [redacted] (tcp/443)
- [redacted] (tcp/80)
- [redacted] (tcp/443)

REKOMENDACJA

Zaleca się konfigurację serwerów HTTP w taki sposób, aby obsługiwane były wyłącznie domeny (*vhosty*), pod którymi dana usługa powinna być dostępna. Zapytania bezpośrednie (np. po IP lub innej domenie) powinny być odrzucane (np. za pomocą kodu błędu HTTP 444 – w nginx spowoduje to natychmiastowe zamknięcie połączenia).