inlockly

MODEL BEZPIECZEŃSTWA I KRYPTOGRAFIA

Raport Bezpieczeństwa Inlockly™

Jak Inlockly chroni Twoje dane: model zerowej wiedzy, szyfrowanie na urządzeniu, procesy konta krok po kroku, odzyskiwanie dostępu i granice ochrony.

Wersja 2.1 · 5 października 2026

© 2026 Inlockly™ · Wszelkie prawa zastrzeżone

Spis treści

01 Najważniejsze w skrócie 3
02 Wstęp 5
03 Fundamentalne zasady ochrony danych 7
04 Co widzi serwer, a czego nie 9
05 Technologie kryptograficzne 12
06 Infrastruktura i ochrona transmisji 17
07 Hasło Główne 18
08 Tworzenie konta 20
09 Logowanie 26
10 Uwierzytelnianie dwuskładnikowe (2FA) 30
11 Skarbce z danymi 32
12 Zmiana Hasła Głównego 36
13 Zmiana adresu e-mail 40
14 Klucz odzyskiwania 44
15 Reset konta bez klucza odzyskiwania 49
16 Przywrócenie dostępu do grup po resecie 53
17 Porównanie ścieżek odzyskiwania dostępu 56
18 Udostępnianie danych 57
19 Funkcje pomocnicze a prywatność 59
20 Model zagrożeń i granice ochrony 61
21 Zalecenia 63
22 Słownik pojęć 64
23 Standardy i źródła 66
24 Historia zmian 67

01

Najważniejsze w skrócie

Inlockly™ to menedżer haseł zbudowany w modelu **zerowej wiedzy** (ang. *zero-knowledge*). Hasła, notatki, kody 2FA i inne dane w skarbcu są szyfrowane na urządzeniu użytkownika, zanim trafią na serwer. Serwer przechowuje i synchronizuje wyłącznie szyfrogramy - nie zna Hasła Głównego ani żadnego klucza, którym można je odczytać.

Ten dokument opisuje, jak to działa: jakich algorytmów używamy, jak powstają i są chronione klucze, co dzieje się przy rejestracji, logowaniu, odblokowaniu skarbca, zmianie hasła i adresu e-mail, jak działa odzyskiwanie dostępu i udostępnianie danych w grupach - oraz czego żaden menedżer haseł nie jest w stanie ochronić.

Właściwość	Jak to robimy
Szyfrowanie danych	AES-256-GCM na urządzeniu użytkownika, nowy losowy wektor przy każdym zapisie
Klucz z hasła	Argon2id (64 MiB, 4 przebiegi, 4 wątki) - każda próba odgadnięcia hasła jest kosztowna
Uwierzytelnianie	Serwer dostaje tylko jednokierunkowy skrót, który dodatkowo haszuje funkcją BCRYPT
Udostępnianie	Klucze skarbców grup przekazywane szyfrowaniem RSA-OAEP (2048 bitów, SHA-256)
Odzyskiwanie dostępu	Klucz odzyskiwania znany tylko użytkownikowi; bez niego - reset konta z 7-dniową karencją
Drugi składnik	Kod e-mail, aplikacja uwierzytelniająca, klucz sprzętowy YubiKey i kody zapasowe
Transmisja	Każde połączenie z serwerem szyfrowane protokołem TLS

Właściwość	Jak to robimy
Lokalizacja danych	Serwery w Unii Europejskiej

Nowe w wersji 2.1

Opisujemy **klucz odzyskiwania** (odzyskanie konta po zapomnieniu Hasła Głównego bez utraty danych), **reset konta bez klucza** z przywróceniem dostępu do grup przez administratora, rozszerzone uwierzytelnianie dwuskładnikowe, a także nowe rozdziały: „Co widzi serwer", model zagrożeń, zalecenia dla administratorów i słownik pojęć. Pełna lista zmian - w rozdziale „Historia zmian".

02

Wstęp

Eksperci do spraw cyberbezpieczeństwa rekomendują używanie unikalnych, losowo wygenerowanych haseł dla każdego konta, które tworzysz w sieci. Długie i złożone hasła są bezpieczne, ale niezwykle trudne do zapamiętania. Z tego powodu wiele osób korzysta ze słabych i łatwych do zapamiętania haseł. Badanie brytyjskiego Narodowego Centrum Cyberbezpieczeństwa (ang. *National Cyber Security Centre*, NCSC) wykazało, że miliony osób logują się do serwisów online hasłami zawierającymi przewidywalne frazy, takie jak imię zwierzęcia, nazwa ulubionego zespołu, a także „hasło" i „123456" (szczegóły w rozdziale „Standardy i źródła").

Tworzenie takich haseł sprawia, że konta są narażone na ataki: cyberprzestępcy łamią słabe hasła w ciągu kilku sekund, korzystając z zautomatyzowanych narzędzi. Hasło o długości 11 znaków składające się wyłącznie z cyfr można złamać niemal natychmiast. Jeśli dodamy do niego małe i wielkie litery oraz znaki specjalne, czas potrzebny na jego złamanie wydłuża się do dziesiątek lat.

Unikalność hasła także pozostawia wiele do życzenia. Użytkownicy często wykorzystują jedno hasło do wielu kont. Złamanie takiego hasła daje przestępcy dostęp do dużej ilości poufnych danych, których wyciek zagraża nie tylko Tobie, ale także Twojej firmie, rodzinie lub zespołowi. W miejscach pracy dochodzi do tego udostępnianie haseł w niezabezpieczony sposób - w wiadomościach, arkuszach czy na kartkach.

Inlockly™ pomaga rozwiązać te problemy. Dostarcza narzędzia do tworzenia, przechowywania i udostępniania haseł oraz poufnych informacji użytkowników i grup, zachowując przy tym najwyższy poziom bezpieczeństwa. Dane synchronizują się automatycznie i bezpiecznie między wszystkimi urządzeniami, które autoryzował użytkownik.

2.1 Zakres i odbiorcy dokumentu

Raport jest przeznaczony dla użytkowników, administratorów IT, zespołów bezpieczeństwa i audytorów, którzy chcą zrozumieć, jak Inlockly™ chroni dane. Opisuje stan aplikacji w dniu wydania tej wersji dokumentu. Szczegółowe procesy przedstawiamy na diagramach - każdy z nich ma część opisową, krok po kroku.

W dokumencie używamy nazw własnych kluczy pisanych wielką literą (np. Klucz Główny, Klucz Symetryczny). Ich definicje znajdziesz w rozdziale „Słownik pojęć”.

03

Fundamentalne zasady ochrony danych

Aplikacja Inlockly™ korzysta z kluczowych środków bezpieczeństwa w celu ochrony wszelkich danych użytkownika. W skład nich wchodzą między innymi:

3.1 Szyfrowanie od końca do końca

Szyfrowanie od końca do końca (ang. *end-to-end*) blokuje dostęp do Twoich haseł, danych i prywatnych informacji dzięki kompleksowemu wdrożeniu szyfrowania AES-256-GCM, wyprowadzaniu kluczy funkcją Argon2id oraz haszowaniu z wykorzystaniem tzw. soli. Dane są czytelne wyłącznie na Twoich urządzeniach.

3.2 Model wyłącznie lokalnego szyfrowania

Wszystkie klucze kryptograficzne są generowane i zarządzane wyłącznie przez aplikację na urządzeniu użytkownika. To samo dotyczy wszelkich operacji szyfrowania i odszyfrowania - odbywają się lokalnie, na autoryzowanym urządzeniu. Serwer nie wykonuje żadnej operacji kryptograficznej na danych skarbca.

3.3 Szyfrowanie z zerową wiedzą

Członkowie zespołu Inlockly™ nie widzą Twoich haseł ani prywatnych danych. Klucz, który otwiera skarbiec, powstaje z Twojego Hasła Głównego i adresu e-mail - na Twoim urządzeniu.

Hasło Główne nigdy nie jest przechowywane ani przesyłane do serwerów Inlockly™. Tak samo jak klucz odzyskiwania, który dostajesz przy rejestracji.

3.4 Prywatność w fazie projektowania

Inlockly™ przechowuje Twoje loginy i prywatne dane w formie zaszyfrowanego skarbca, synchronizowanego pomiędzy wszystkimi Twoimi urządzeniami. Zanim dane opuszczą urządzenie, są w pełni zaszyfrowane, dlatego tylko Ty masz do nich dostęp. Zespół programistów i inni pracownicy Inlockly™ nie mają możliwości odczytania Twoich danych -

nawet gdybyśmy chcieli to zrobić. Funkcje, które mogłyby zdradzić zawartość skarbca, działają lokalnie: na przykład sprawdzanie haseł w znanych wyciekach odbywa się na urządzeniu, bez wysyłania haseł ani ich skrótów do zewnętrznych usług.

3.5 Bezpieczne udostępnianie danych

Inlockly™ umożliwia tworzenie grup (rodzina, zespół, firma), w których użytkownicy bezpiecznie udostępniają sobie dane. Kombinacja szyfrowania symetrycznego i asymetrycznego chroni dane podczas udostępniania: każdy członek dostaje własną, zaszyfrowaną kopię klucza skarbca grupy, którą odczyta tylko on. Serwer przechowuje te kopie, ale nie jest w stanie ich otworzyć.

3.6 Audyty bezpieczeństwa

Poddajemy aplikację Inlockly™ regularnym przeglądom i testom bezpieczeństwa. Przeprowadzamy rutynowe kontrole naszych usług i całej infrastruktury. Aplikacja jest zgodna z przepisami o ochronie danych osobowych, w tym z RODO (GDPR).

3.7 Szybka reakcja na incydenty

Zespół Inlockly™ reaguje na każdy zgłoszony incydent. Badamy, analizujemy i eliminujemy wynikające ze zgłoszenia błędy bądź potencjalne luki w zabezpieczeniach. O każdym znaczącym incydencie informujemy użytkowników na naszej stronie internetowej oraz w informacjach o zmianach w nowej wersji aplikacji.

3.8 Program bug bounty

Inlockly™ prowadzi program bug bounty, w ramach którego badacze bezpieczeństwa mogą zgłaszać odnalezione podatności. Za znalezienie i opisanie podatności przewidujemy nagrody. Trzymamy się jasnych reguł i prosimy o zachowanie zasady odpowiedzialnego ujawniania (ang. *responsible disclosure*) - zgłoszenie błędu zespołowi Inlockly™ i **nieujawnianie go publicznie** do czasu potwierdzenia, że błąd został usunięty.

04

Co widzi serwer, a czego nie

Model zerowej wiedzy nie oznacza, że serwer nie wie nic. Aby synchronizować dane, rozliczać plany i egzekwować uprawnienia w grupach, usługa musi znać część informacji administracyjnych. Poniżej uczciwie rozdzielamy jedne od drugich.

4.1 Dane zaszyfrowane na urządzeniu

Te dane trafiają na serwer wyłącznie w postaci szyfrogramów. Serwer nie ma klucza, którym mógłby je odczytać:

Dane	Czym są zaszyfrowane
Treść pozycji skarbca: nazwa, login, hasło, adresy, notatki, pola własne, klucze 2FA	Klucz Symetryczny (skarbiec prywatny) albo klucz grupy
Metadane pozycji: ulubione, kosz, klucze aktywnych linków udostępniania	jak wyżej
Historia zmian pozycji (poprzednie wersje)	jak wyżej
Klucz Symetryczny konta	Klucz Główny (z Hasła Głównego) i - osobno - klucz z klucza odzyskiwania
Klucz Prywatny RSA	Klucz Symetryczny
Kopie kluczy skarbców grup	Klucz Publiczny członka (RSA-OAEP)
Ustawienia autouzupełniania dla witryn	Klucz Symetryczny
Treść udostępniona linkiem	Jednorazowy klucz linku, który nie trafia na serwer

4.2 Dane, które serwer zna

Dane	Po co
Imię, nazwisko i adres e-mail	Konto, logowanie, powiadomienia o bezpieczeństwie
Skrót BCRYPT z Hasha Hasła Głównego	Uwierzytelnienie bez znajomości hasła
Plan, subskrypcja, rozliczenia	Uprawnienia planu (płatności obsługuje operator płatności)
Nazwy grup, działów i folderów, członkostwa i uprawnienia	Kontrola dostępu w grupach
Struktura skarbca: identyfikatory pozycji, folder, autor i czas zmiany, numer wersji	Synchronizacja i historia zmian
Urządzenia i sesje: system, aplikacja, adres IP, przybliżona lokalizacja, czas	Limity urządzeń, alerty o nowym logowaniu, dziennik aktywności
Dziennik aktywności konta i grupy	Przejrzystość dla użytkownika i administratorów
Ustawienia metod 2FA	Weryfikacja drugiego składnika (patrz rozdział o 2FA)

Dlaczego nazwy folderów i grup nie są szyfrowane?

Serwer egzekwuje uprawnienia w grupach: decyduje, kto widzi który folder, i wysyła powiadomienia o zmianach. Nie szyfrujemy więc struktury organizacyjnej, tylko treść. Nie umieszczaj poufnych informacji w nazwach folderów, działów ani grup - trzymaj je w pozycjach skarbca.

© 2026 Inlockly™ · inlockly.com

4.3 Czego serwer nigdy nie otrzymuje

- Hasła Głównego - ani przy rejestracji, ani przy logowaniu, ani przy zmianie hasła.
- Klucza Głównego, Klucza Symetrycznego, Klucza Prywatnego ani kluczy grup w jawnej postaci.
- Klucza odzyskiwania - serwer zna tylko wyprowadzony z niego dowód, zapisany pod pieczęcią HMAC.
- Klucza linku udostępniania - jest w części adresu po znaku `#`, której przeglądarka nie wysyła.
- Haseł sprawdzanych w poszukiwaniu wycieków - porównanie odbywa się na urządzeniu.

05

Technologie kryptograficzne

Inlockly™ korzysta wyłącznie ze sprawdzonych, ustandaryzowanych algorytmów, dostępnych w natywnych bibliotekach kryptograficznych platform (WebCrypto). Funkcję Argon2id, której WebCrypto nie oferuje, realizuje biblioteka referencyjna skompilowana do WebAssembly.

5.1 Parametry w jednym miejscu

Element	Algorytm i parametry
Klucz Główny	Argon2id: 64 MiB pamięci, 4 przebiegi, 4 wątki, wynik 256 bitów; sól: pierwsze 16 bajtów SHA-256 znormalizowanego adresu e-mail
Hash Hasła Głównego	PBKDF2-SHA256, 1 powtórzenie, ładunek: Klucz Główny, sól: Hasło Główne; wynik 256 bitów
Skrót hasła na serwerze	BCRYPT, koszt 12
Klucz Symetryczny	256 bitów z CSPRNG
Szyfrowanie danych i kluczy	AES-256-GCM, nonce 96 bitów z CSPRNG (nowy przy każdym szyfrowaniu), znacznik 128 bitów
Format szyfrogramu	`ILK1.` + nonce + szyfrogram ze znacznikiem (base64url)
Para kluczy	RSA-OAEP, 2048 bitów, SHA-256 (OAEP i MGF1)
Klucz grupy	256 bitów z CSPRNG
Klucz odzyskiwania	128 bitów z CSPRNG, zapis: 28 znaków Crockford base32 z 12-bitową sumą kontrolną

Element	Algorytm i parametry
Klucze z klucza odzyskiwania	HKDF-SHA256, pusta sól, kontekst „inlockly recovery v1 wrap" i „... auth"
Odblokowanie lokalne	HMAC-SHA256 (klucz: Hash Hasła Głównego, wiadomość: adres e-mail)
Kopia zapasowa skarbca (.inlockly)	Argon2id z losową solą i parametrami zapisanymi w pliku, AES-256-GCM
Transmisja	TLS

5.2 Hierarchia kluczy

Klucze tworzą łańcuch - każdy kolejny jest chroniony poprzednim. Dzięki temu zmiana Hasła Głównego lub adresu e-mail wymaga ponownego zaszyfrowania jednej krótkiej wartości, a nie całego skarbca.

1. **Hasło Główne** (tylko w pamięci użytkownika) → funkcją Argon2id powstaje **Klucz Główny**.
2. **Klucz Główny** szyfruje **Klucz Symetryczny** konta (Chroniony Klucz Symetryczny).
3. **Klucz Symetryczny** szyfruje pozycje skarbca prywatnego oraz **Klucz Prywatny** RSA.
4. **Klucz Prywatny** odszyfrowuje kopie **kluczy grup**, a te szyfrują pozycje skarbców grup.
5. Niezależnie od punktu 2 **klucz odzyskiwania** szyfruje drugą kopię tego samego Klucza Symetrycznego.

5.3 Standard szyfrowania AES

Zaawansowany Standard Szyfrowania (ang. *Advanced Encryption Standard*, AES) to nowoczesny, blokowy szyfr symetryczny - jeden z najpopularniejszych na świecie. Opublikowali go Vincent Rijmen i Joan Daemen, a w 2001 roku został przyjęty jako standard federalny USA (FIPS PUB 197). Opisuje go również norma ISO/IEC 18033-3.

W Inlockly™ korzystamy z wariantu AES-256, gdzie 256 oznacza długość klucza w bitach. Szyfrowanie kluczem 256-bitowym składa się z 14 rund przekształceń każdego 128-bitowego bloku danych. Szyfr jest symetryczny: do zaszyfrowania i odszyfrowania danych używamy tego samego klucza.

Tryb GCM

AES-256 działa w trybie GCM (ang. *Galois/Counter Mode*), zdefiniowanym w standardzie NIST SP 800-38D. GCM szyfruje z uwierzytelnieniem (ang. *authenticated encryption*): jedna operacja zapewnia zarówno poufność danych, jak i ich integralność. Do każdego szyfrogramu dołączany jest 128-bitowy znacznik uwierzytelniający. Każda, nawet jednobitowa zmiana zaszyfrowanych danych sprawia, że odszyfrowanie kończy się błędem, a zmodyfikowane dane nie są akceptowane.

Każda operacja szyfrowania korzysta z nowego, losowego 96-bitowego wektora inicjalizującego (ang. *nonce*) z generatora CSPRNG - także przy ponownym zapisie tej samej pozycji. Identyczne dane za każdym razem dają inny szyfrogram, więc z zaszyfrowanej postaci nie da się wywnioskować, czy dwie pozycje mają tę samą treść.

5.4 Algorytm RSA

Algorytm Rivesta-Shamira-Adlemana (RSA) to jeden z najpopularniejszych algorytmów kryptografii asymetrycznej. Tworzy dwa powiązane klucze: publiczny, którym każdy może zaszyfrować wiadomość, oraz prywatny, którym tylko jego posiadacz może ją odczytać. Bezpieczeństwo RSA opiera się na trudności faktoryzacji dużych liczb złożonych.

W Inlockly™ korzystamy z RSA z kluczem 2048 bitów i schematem dopełnienia OAEP (ang. *Optimal Asymmetric Encryption Padding*) z funkcją skrótu SHA-256, zdefiniowanym w standardzie PKCS#1 v2.2 (RFC 8017). OAEP wprowadza do każdego szyfrowania element losowy i chroni przed znanymi atakami na starsze dopełnienie PKCS#1 v1.5.

RSA nie szyfruje w Inlockly™ samych danych. Służy wyłącznie do bezpiecznego przekazania członkom grupy losowego, 256-bitowego klucza skarbca grupy, którym dane szyfruje AES. Takie połączenie, nazywane szyfrowaniem hybrydowym, łączy zalety kryptografii asymetrycznej (nie trzeba wymieniać tajnego klucza) z wydajnością i siłą szyfru symetrycznego.

5.5 Funkcja Argon2id

Argon2 to funkcja wyprowadzania klucza z hasła, która w 2015 roku wygrała międzynarodowy konkurs Password Hashing Competition. Została ustandaryzowana w dokumencie RFC 9106. Wariant Argon2id łączy odporność na ataki kanałami bocznymi z odpornością na ataki z użyciem kart graficznych i układów specjalizowanych (ASIC). Jest wariantem zalecanym przez RFC 9106 i OWASP.

W odróżnieniu od funkcji takich jak PBKDF2, których koszt zależy wyłącznie od liczby powtórzeń, Argon2id wymaga do każdego obliczenia dużej ilości pamięci operacyjnej. Atakujący, który próbuje odgadnąć hasło, nie może tanio prowadzić tysięcy prób równolegle: każda próba kosztuje tyle samo pamięci co pojedyncze logowanie użytkownika.

Parametry (64 MiB pamięci, 4 przebiegi, 4 wątki) odpowiadają zalecanej konfiguracji RFC 9106 dla urządzeń o ograniczonej pamięci, powiększonej o dodatkowy przebieg. Są stałą aplikacji i nie są przechowywane na serwerze.

Dlaczego 64 MiB, a nie więcej?

Hasło Główne jest potrzebne także w rozszerzeniach systemowych - na przykład w autouzupełnianiu na iPhonie, które ma twardy limit pamięci procesu. Większy koszt pamięci oznaczałby, że system zamknie rozszerzenie w trakcie odblokowania. Wybrane parametry to norma rynkowa lub lepiej: o jeden przebieg więcej niż zalecana konfiguracja.

5.6 PBKDF2, HKDF i HMAC

PBKDF2-SHA256 służy wyłącznie do wyprowadzenia Hasha Hasła Głównego z gotowego Klucza Głównego, w jednym powtórzeniu (patrz rozdział „Tworzenie konta"). **HKDF-SHA256** (RFC 5869) wyprowadza dwa niezależne klucze z losowego klucza odzyskiwania. **HMAC-SHA256** pozwala sprawdzić Hasło Główne przy lokalnym odblokowaniu skarbca, bez udziału serwera.

5.7 Algorytm BCRYPT

BCRYPT to algorytm haszujący hasła, z wbudowaną solą i mechanizmem rozciągania klucza. Serwer Inlockly™ używa go, aby ponownie wymieszać otrzymany Hash Hasła Głównego (koszt 12) - nawet wyciek bazy danych nie ujawnia wartości, którą można by przesłać przy logowaniu.

5.8 Generowanie liczb losowych - CSPRNG

Kryptograficznie Bezpieczny Generator Liczb Pseudolosowych (ang. *Cryptographically Secure Pseudo-Random Number Generator*, CSPRNG) dostarcza nieprzewidywalnych wartości: kluczy, wektorów inicjalizujących i klucza odzyskiwania. Inlockly™ korzysta z generatora systemu operacyjnego udostępnianego przez platformę.

5.9 Protokół TLS

TLS (ang. *Transport Layer Security*), następca protokołu SSL, zabezpiecza transmisję między aplikacją a serwerem (RFC 8446). W Inlockly™ jest drugą, niezależną warstwą ochrony: dane przesyłane przez TLS są już zaszyfrowane na urządzeniu.

06

Infrastruktura i ochrona transmisji

Usługa Inlockly™ działa na serwerach zlokalizowanych w Unii Europejskiej. Aplikacje łączą się z serwerem wyłącznie przez szyfrowane połączenia TLS.

Bezpieczeństwo danych w skarbcu nie zależy od infrastruktury: serwer przechowuje szyfrogramy, do których nie ma kluczy. Przejęcie serwera, kopii zapasowej albo bazy danych nie daje dostępu do treści skarbców. Infrastruktura odpowiada natomiast za dostępność usługi, integralność danych administracyjnych i ochronę przed nadużyciami. Dlatego serwer stosuje między innymi:

- **limity prób** dla logowania (6 na minutę z jednego adresu IP i 20 na minutę dla jednego adresu e-mail), odzyskiwania i resetu konta - każda ścieżka ma własny, niezależny licznik;
- **ochronę przed botami** (Cloudflare Turnstile) przy rejestracji, logowaniu i rozpoczęciu odzyskiwania;
- **pieczęć HMAC** dla wszystkiego, co decyduje o dostępie (tokeny sesji, tokeny zaufanych urządzeń, ustawienia 2FA, kody zapasowe) - wpis do bazy bez znajomości sekretu serwera nie pozwala obejść hasła ani drugiego składnika;
- **odpowiedzi odporne na sprawdzanie kont** - rejestracja, logowanie, odzyskiwanie i reset zwracają tę samą odpowiedź niezależnie od tego, czy adres ma konto w Inlockly™.

Szczegóły infrastruktury

Szczegółowy opis architektury infrastruktury (dostępność, kopie zapasowe, regiony przetwarzania danych) opublikujemy w kolejnej wersji raportu.

07

Hasło Główne

Przy zakładaniu konta w Inlockly™ użytkownik ustala swoje **Hasło Główne**. Służy ono do uwierzytelniania, a przede wszystkim to z niego - na urządzeniu - wyprowadzane są klucze, które otwierają skarbiec prywatny i skarbce grup, do których należy użytkownik.

Skarbiec to miejsce, w którym przechowywane są loginy, hasła, kody 2FA i inne poufne dane. Dlatego Hasło Główne powinno być silne, a konto dodatkowo zabezpieczone uwierzytelnianiem dwuskładnikowym (2FA).

7.1 Silne Hasło Główne

Podczas tworzenia konta aplikacja pokazuje wskaźnik siły hasła i nie przyjmie hasła ocenionego jako słabe. Hasło Główne powinno być długie i niepowtarzalne. Dobrym wyborem jest fraza złożona z kilku przypadkowych słów albo dłuższy ciąg różnych znaków: liter małych i wielkich, cyfr i znaków specjalnych.

Siła Hasła Głównego bezpośrednio wpływa na bezpieczeństwo danych. Funkcja Argon2id sprawia, że każda próba odgadnięcia hasła jest kosztowna, ale nie zastąpi dobrego hasła: krótkie lub przewidywalne hasło da się odgadnąć mimo wysokiego kosztu pojedynczej próby.

> ⚠ Uwaga
>
> **Nigdy nie używaj Hasła Głównego w innej witrynie lub aplikacji.** Wyciek danych w innym serwisie mógłby wtedy zagrozić bezpieczeństwu Twojego skarbca.
>
> **Nigdy nie udostępniaj Hasła Głównego ani klucza odzyskiwania** - także osobom podającym się za pracowników Inlockly™. Nikt z naszego zespołu nie potrzebuje ich znać i nigdy Cię o nie nie poprosi. Każdą taką prośbę - mailem, telefonem czy na czacie - traktuj jako próbę wyłudzenia i zgłoś ją nam.

7.2 Ochrona Hasła Głównego

Zaszyfrowane dane skarbca są bezużyteczne bez klucza, który powstaje z Hasła Głównego i adresu e-mail użytkownika. Hasło Główne jest usuwane z pamięci aplikacji zaraz po wyprowadzeniu kluczy i **nigdy nie jest wysyłane** na serwer Inlockly™. Serwer otrzymuje wyłącznie jednokierunkowy skrót (Hash Hasła Głównego), z którego nie da się odtworzyć ani hasła, ani kluczy.

Dzięki temu nikt z zespołu Inlockly™ nie uzyska dostępu do Twojego Hasła Głównego i nie będzie w stanie odczytać zaszyfrowanych danych Twojego skarbca ani skarbców grup, do których należysz.

7.3 Gdy zapomnisz Hasła Głównego

Ponieważ Hasło Główne nie trafia na serwer, nie możemy go przypomnieć ani zresetować tak jak w zwykłych serwisach. Od wersji 2.1 masz jednak dwie drogi wyjścia - obie zgodne z modelem zerowej wiedzy:

- **Klucz odzyskiwania** - otrzymujesz go przy rejestracji. Pozwala ustawić nowe Hasło Główne i zachować wszystkie dane: skarbiec prywatny, klucze i dostęp do grup. Opis w rozdziale „Klucz odzyskiwania”.
- **Reset konta bez klucza** - gdy nie masz ani hasła, ani klucza. Zachowuje konto, plan i członkostwa w grupach, ale skarbiec prywatny zostaje usunięty, bo nikt nie jest w stanie go odszyfrować. Dostęp do danych grup przywraca administrator grupy. Opis w rozdziałach „Reset konta” i „Przywrócenie dostępu do grup”.

08

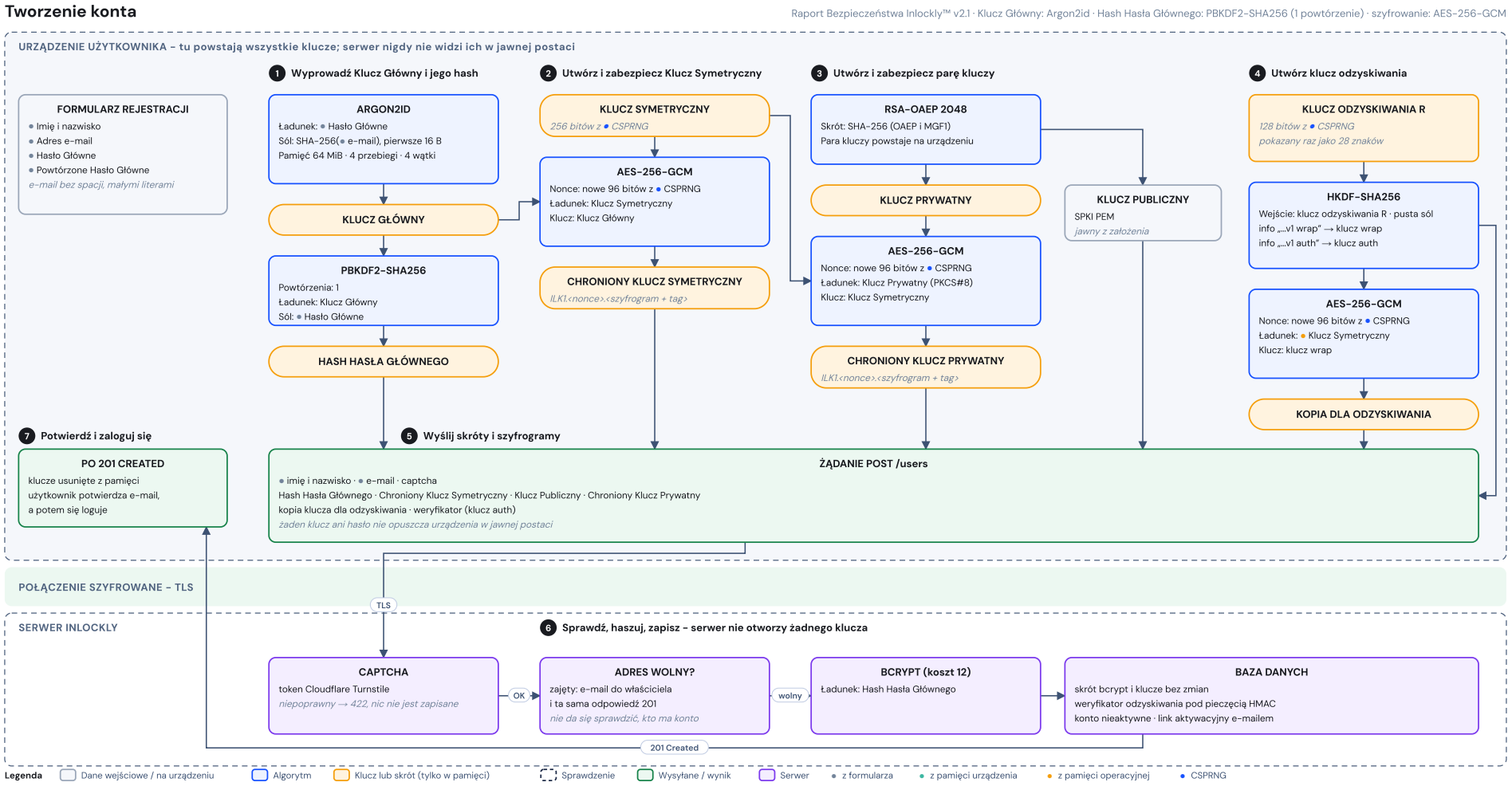
Tworzenie konta

Rejestracja to moment, w którym powstają wszystkie klucze konta. Dzieje się to w całości na urządzeniu użytkownika - serwer otrzymuje wyłącznie skróty i szyfrogramy.

W skrócie:

- z Hasła Głównego i adresu e-mail powstaje **Klucz Główny** (Argon2id) i **Hash Hasła Głównego** (PBKDF2);
- losowy **Klucz Symetryczny** szyfruje skarbiec prywatny, a sam jest szyfrowany Kluczem Głównym;
- **para kluczy RSA** pozwala bezpiecznie otrzymywać klucze skarbców grup;
- **klucz odzyskiwania** chroni drugą kopię Klucza Symetrycznego na wypadek zapomnienia hasła;
- serwer haszuje Hash Hasła Głównego funkcją BCRYPT i zapisuje klucze bez zmian - nie może ich odczytać.

8.1 Diagram procesu tworzenia konta



Rysunek 1. Tworzenie konta: wszystkie klucze, łącznie z kluczem odzyskiwania, powstają na urządzeniu użytkownika.

8.2 Część opisowa procesu tworzenia konta

Formularz rejestracyjny

W pierwszej kolejności użytkownik wypełnia formularz rejestracyjny, w którym podaje:

- imię i nazwisko,
- adres e-mail,
- Hasło Główne,
- powtórzone Hasło Główne.

Użytkownik akceptuje regulamin usługi i może wyrazić zgodę na otrzymywanie informacji handlowych. Formularz jest chroniony zabezpieczeniem przed botami (Cloudflare Turnstile). Hasło Główne podaje się dwukrotnie, aby wyeliminować literówkę - aplikacja porównuje oba pola na urządzeniu. Wskaźnik siły informuje, jak odporne jest hasło na odgadnięcie, a aplikacja nie przyjmie hasła ocenionego jako słabe.

Adres e-mail jest normalizowany: aplikacja usuwa z niego zbędne spacje i zamienia litery na małe. Ten sam adres zawsze daje więc te same klucze, niezależnie od sposobu wpisania. Wszystkie opisane poniżej operacje kryptograficzne wykonywane są na urządzeniu użytkownika, zanim jakiekolwiek dane zostaną wysłane na serwer.

Wyprowadzenie Klucza Głównego

Klucz Główny (ang. *Master Key*) wyprowadzany jest lokalnie z Hasła Głównego funkcją Argon2id. W jednej operacji funkcja miesza hasło z solą i rozciąga je tak, aby każda próba odgadnięcia hasła była kosztowna. Solą jest skrót SHA-256 znormalizowanego adresu e-mail (pierwsze 16 bajtów). Pozostałe parametry: 64 MiB pamięci, 4 przebiegi i 4 wątki. W wyniku powstaje 256-bitowy Klucz Główny, który służy do zaszyfrowania Klucza Symetrycznego.

Klucz Główny nigdy nie jest przechowywany ani przesyłany do serwerów Inlockly™.

Generowanie Klucza Symetrycznego

Klucz Symetryczny (ang. *Symmetric Key*) to losowa, 256-bitowa wartość z generatora CSPRNG. Szyfruje dane w skarbcu prywatnym oraz Klucz Prywatny. Aby można go było bezpiecznie przechować, aplikacja szyfruje go algorytmem AES-256-GCM z parametrami:

- wektor inicjalizujący (nonce): nowa, losowa wartość 96-bitowa z generatora CSPRNG,

- ładunek: Klucz Symetryczny,
- klucz: Klucz Główny.

W wyniku powstaje Chroniony Klucz Symetryczny (ang. *Protected Symmetric Key*), który trafia na serwer. Odczytanie go wymaga Klucza Głównego, a ten może wyprowadzić wyłącznie osoba znająca Hasło Główne.

Dlaczego dane nie są szyfrowane bezpośrednio Kluczem Głównym?

Klucz Główny wynika z Hasła Głównego i adresu e-mail, więc zmienia się przy każdej zmianie któregoś z nich. Gdyby dane były zaszyfrowane bezpośrednio Kluczem Głównym, każda zmiana hasła lub adresu wymagałaby odszyfrowania i ponownego zaszyfrowania całego skarbca - operacji długiej i narażonej na przerwanie. Dlatego dane szyfruje losowy Klucz Symetryczny, który nie zmienia się przez cały okres istnienia konta. Przy zmianie hasła lub adresu ponownie szyfrowana jest tylko ta jedna krótka wartość.

Dlaczego Chroniony Klucz Symetryczny trafia na serwer?

Klucz Symetryczny jest potrzebny na każdym urządzeniu użytkownika. Jego zaszyfrowana postać jest przechowywana na serwerze i przekazywana po zalogowaniu na nowym urządzeniu. Odszyfrowanie odbywa się wyłącznie na urządzeniu i wymaga Klucza Głównego - sama znajomość szyfrogramu nie pozwala odczytać danych.

Generowanie pary kluczy RSA

Aplikacja generuje na urządzeniu Parę Kluczy RSA (2048 bitów, OAEP z SHA-256). Służy ona do bezpiecznego przekazywania kluczy skarbców grup, dzięki którym użytkownicy udostępniają sobie dane.

Klucz Publiczny jest jawny z założenia. Gdy Użytkownik A dodaje Użytkownika B do grupy, szyfruje klucz skarbca grupy Kluczem Publicznym B i przekazuje mu tylko zaszyfrowaną kopię. Odszyfrować ją może wyłącznie Klucz Prywatny B. Serwer przechowuje tę kopię, ale nie jest w stanie jej odczytać.

Klucz Prywatny pozwala odszyfrować kopie kluczy grup, a przez to dane we wszystkich skarbcach udostępnionych użytkownikowi. Dlatego jest chroniony na równi z Kluczem Symetrycznym: nigdy nie opuszcza urządzenia w jawnej postaci. Przed wysłaniem na serwer aplikacja szyfruje go algorytmem AES-256-GCM z nowym wektorem inicjalizującym, a kluczem jest **Klucz Symetryczny**. Powstaje Chroniony Klucz Prywatny. Ponieważ chroni go Klucz Symetryczny, a nie Klucz Główny, zmiana hasła lub adresu e-mail nie wymaga ponownego szyfrowania Klucza Prywatnego.

Hash Hasła Głównego

Hash Hasła Głównego (ang. *Master Password Hash*) służy do uwierzytelnienia - serwer nie potrzebuje do tego Hasła Głównego i nigdy go nie otrzymuje. Wyprowadzamy go funkcją PBKDF2-SHA256 z parametrami:

- powtórzenia: 1,
- ładunek: Klucz Główny,
- sól: Hasło Główne.

W wyniku powstaje 256-bitowy Hash Hasła Głównego. Jedno powtórzenie jest świadomym wyborem: przed odgadnięciem hasła chroni już Argon2id, a Klucz Główny, z którego powstaje hash, jest 256-bitową wartością niemożliwą do odgadnięcia. Hash ma być jedynie jednokierunkowy - drugie wywołanie Argon2id wydłużyłoby logowanie bez zysku dla bezpieczeństwa.

Klucz odzyskiwania

Przed utworzeniem konta aplikacja generuje **klucz odzyskiwania** - 128 losowych bitów z generatora CSPRNG. Użytkownik widzi go jako 28 znaków w siedmiu grupach po cztery i musi go zachować poza Inlockly™ (kopiując, pobierając lub drukując). Konto nie powstanie, dopóki użytkownik nie potwierdzi, że zapisał klucz.

Z klucza odzyskiwania funkcją HKDF-SHA256 powstają dwa niezależne klucze: **klucz wrap** szyfruje (AES-256-GCM) drugą kopię Klucza Symetrycznego, a **klucz auth** staje się weryfikatorem, który pozwoli serwerowi sprawdzić, że użytkownik zna klucz odzyskiwania, bez poznania samego klucza. Szczegóły - w rozdziale „Klucz odzyskiwania".

Przesłanie danych i rola serwera

Połączeniem szyfrowanym TLS aplikacja przesyła na serwer imię, nazwisko, adres e-mail i token zabezpieczenia przed botami oraz materiał kryptograficzny: Hash Hasła Głównego, Chroniony Klucz Symetryczny, Klucz Publiczny, Chroniony Klucz Prywatny, kopię Klucza

Symetrycznego dla odzyskiwania i weryfikator klucza odzyskiwania. Żaden klucz ani hasło nie opuszcza urządzenia w jawnej postaci - poza Kluczem Publicznym, jawnym z założenia.

Serwer najpierw sprawdza zabezpieczenie przed botami, a następnie, czy adres e-mail nie jest już zajęty. Jeśli jest, serwer nie tworzy konta, wysyła właścicielowi adresu wiadomość o próbie rejestracji i zwraca tę samą odpowiedź co przy udanej rejestracji - nie da się więc ustalić, czy dany adres ma konto w Inlockly™. Jeśli adres jest wolny, serwer haszuje Hash Hasła Głównego funkcją BCRYPT (koszt 12), weryfikator zapisuje pod pieczęcią HMAC, a klucze - bez żadnych modyfikacji, bo nie jest w stanie ich odczytać. Konto pozostaje nieaktywne do czasu potwierdzenia adresu e-mail linkiem aktywacyjnym.

Po otrzymaniu odpowiedzi aplikacja usuwa wszystkie klucze z pamięci operacyjnej i niczego nie zapisuje na urządzeniu. Użytkownik potwierdza adres e-mail, a następnie loguje się w sposób opisany w rozdziale „Logowanie”.

09

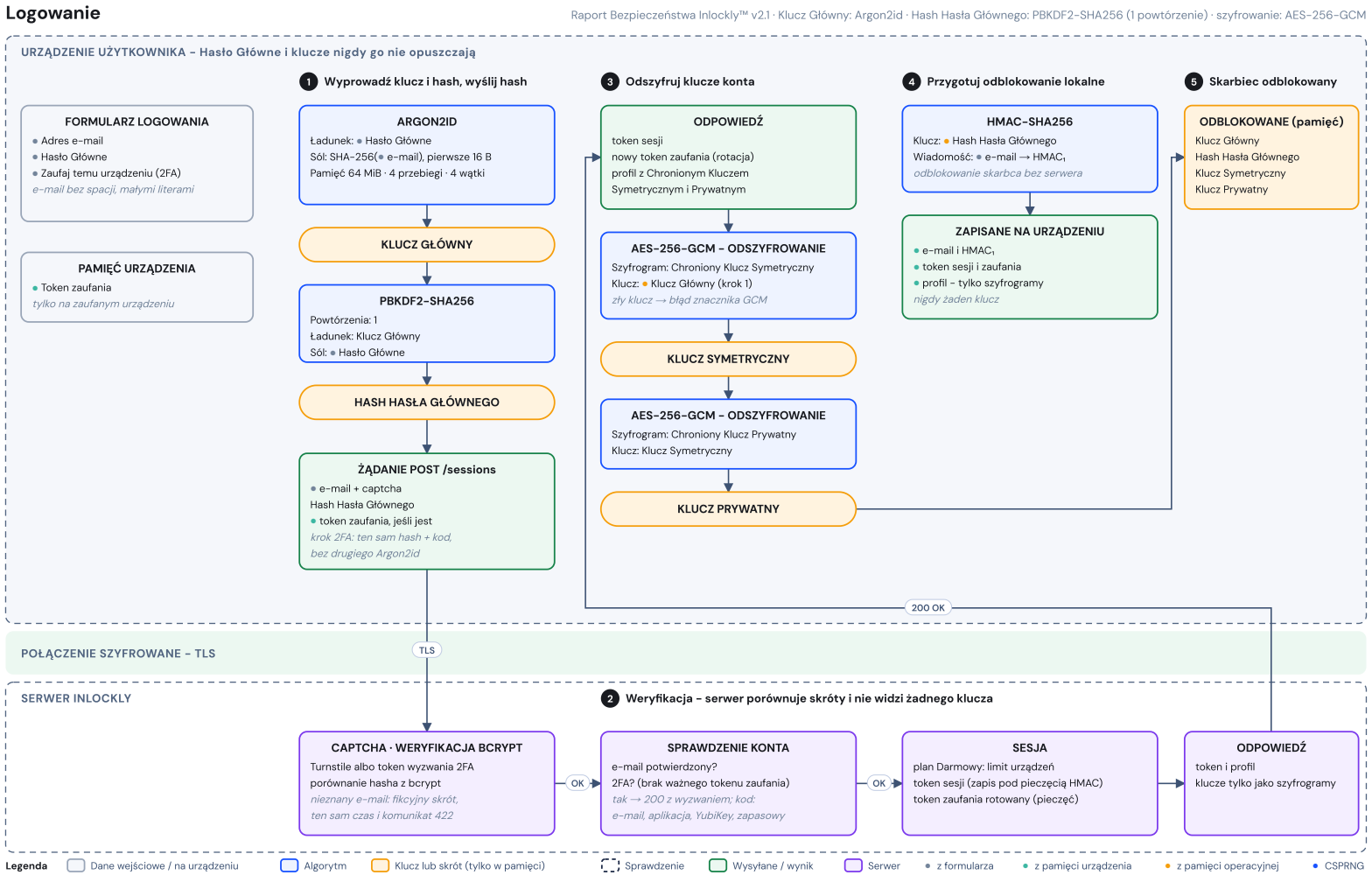
Logowanie

Logowanie odbywa się w dwóch krokach: serwer sprawdza, czy użytkownik zna Hasło Główne (porównując skróty), a urządzenie - mając Klucz Główny - samo odszyfrowuje klucze konta. Samo przejście weryfikacji na serwerze nie wystarcza, żeby odczytać dane.

W skrócie:

- na serwer trafia tylko Hash Hasła Głównego - nigdy hasło ani Klucz Główny;
- konto z włączonym 2FA wymaga drugiego składnika, chyba że urządzenie jest zaufane;
- serwer odsyła klucze konta wyłącznie w postaci zaszyfrowanej;
- urządzenie zapisuje kod HMAC, który pozwala później odblokować skarbiec bez połączenia z serwerem.

9.1 Diagram procesu logowania



Rysunek 2. Logowanie: serwer porównuje skróty i odsyła klucze wyłącznie w postaci zaszyfrowanej.

9.2 Część opisowa procesu logowania

Użytkownik wypełnia formularz logowania, podając adres e-mail i Hasło Główne. Formularz jest chroniony zabezpieczeniem przed botami (Cloudflare Turnstile), a adres e-mail jest normalizowany tak samo jak przy rejestracji.

1. **Wyprowadzenie kluczy i wysłanie Hasha Hasła Głównego.** Aplikacja wyprowadza z Hasła Głównego Klucz Główny funkcją Argon2id (sól: skrót SHA-256 adresu e-mail, parametry jak przy rejestracji). Klucz Główny pozostaje wyłącznie w pamięci operacyjnej. Z Klucza Głównego i Hasła Głównego powstaje Hash Hasła Głównego (PBKDF2-SHA256, jedno powtórzenie). Połączeniem TLS aplikacja wysyła adres e-mail, Hash Hasła Głównego i token zabezpieczenia przed botami. Na serwer nie trafia ani Hasło Główne, ani Klucz Główny.

2. **Weryfikacja na serwerze.** Serwer sprawdza zabezpieczenie przed botami i porównuje otrzymany Hash Hasła Głównego z zapisanym skrótem BCRYPT. Jeżeli adres nie istnieje lub hash jest niezgodny, zwraca ten sam komunikat o błędnych danych. Dla nieznanego adresu wykonuje porównanie z fikcyjnym skrótem, dzięki czemu czas odpowiedzi jest taki sam - nie da się ustalić, czy adres ma konto. Liczbę prób ograniczają limity: 6 na minutę z jednego adresu IP i 20 na minutę dla jednego adresu e-mail.

   Następnie serwer sprawdza, czy adres konta został potwierdzony i czy konto ma włączone uwierzytelnianie dwuskładnikowe, a urządzenie nie jest oznaczone jako zaufane. Jeśli tak, zamiast sesji zwraca wezwanie do podania drugiego składnika (rozdział „Uwierzytelnianie dwuskładnikowe"). Aplikacja odsyła wtedy ten sam Hash Hasła Głównego razem z kodem - bez ponownego obliczania Argon2id.

   Na koniec serwer tworzy sesję urządzenia, z uwzględnieniem limitu urządzeń planu. Wydaje token sesji, a na zaufanym urządzeniu również token zaufania; oba przechowuje w bazie wyłącznie pod pieczęcią HMAC. Udane logowanie anuluje oczekujący reset konta (rozdział „Reset konta"). W odpowiedzi przesyła tokeny i dane konta, w tym Chroniony Klucz Symetryczny i Chroniony Klucz Prywatny - klucze wyłącznie w postaci zaszyfrowanej.

3. **Odszyfrowanie kluczy konta.** Aplikacja odszyfrowuje Chroniony Klucz Symetryczny algorytmem AES-256-GCM z użyciem Klucza Głównego z kroku 1, a następnie Kluczem Symetrycznym odszyfrowuje Chroniony Klucz Prywatny. Znacznik uwierzytelniający GCM gwarantuje, że niezgodny klucz lub zmieniony szyfrogram kończy się błędem, a nie błędnie odczytaną wartością. Samo uwierzytelnienie na serwerze nie wystarcza do odczytania danych - potrzebna jest znajomość Hasła Głównego.

4. **Przygotowanie odblokowania lokalnego.** Aplikacja oblicza kod HMAC-SHA256, w którym kluczem jest Hash Hasła Głównego, a wiadomością adres e-mail. Kod, adres e-mail, tokeny i dane konta (wyłącznie zaszyfrowane) zapisuje w pamięci urządzenia. Pozwala to później odblokować skarbiec bez serwera (rozdział „Skarbce z danymi"). W pamięci urządzenia nie jest zapisywany żaden klucz.

5. **Odblokowany skarbiec.** Klucz Główny, Hash Hasła Głównego, Klucz Symetryczny i Klucz Prywatny pozostają wyłącznie w pamięci operacyjnej. Aplikacja pobiera zaszyfrowane pozycje i odszyfrowuje je na urządzeniu: pozycje skarbca prywatnego Kluczem Symetrycznym, a pozycje skarbców grup kluczami grup, odszyfrowanymi wcześniej Kluczem Prywatnym.

Powiadomienia o logowaniu

Logowanie na nowym urządzeniu wywołuje wiadomość e-mail ze szczegółami próby: systemem, adresem IP, krajem i czasem. Jeśli to nie Ty - zmień Hasło Główne i sprawdź listę urządzeń w ustawieniach konta.

10

Uwierzytelnianie dwuskładnikowe (2FA)

Uwierzytelnianie dwuskładnikowe (ang. *two-factor authentication*, 2FA) weryfikuje użytkownika za pomocą dwóch różnych form identyfikacji: czegoś, co wie (Hasło Główne), i czegoś, co ma (skrzynka e-mail, telefon, klucz sprzętowy). Znajomość adresu i hasła nie wystarczy wtedy do zalogowania się na Twoje konto.

10.1 Metody drugiego składnika

Metoda	Jak działa	Dostępność
Kod e-mail	Jednorazowy, 6-cyfrowy kod wysyłany na adres konta, ważny 15 minut	Każdy plan
Aplikacja uwierzytelniająca (TOTP)	6-cyfrowy kod zmieniający się co 30 sekund (RFC 6238), np. z Google Authenticator	Od planu Osobistego
Klucz sprzętowy YubiKey (OTP)	Jednorazowe hasło generowane przez fizyczny klucz po dotknięciu	Od planu Osobistego
Kody zapasowe	10 jednorazowych kodów na wypadek utraty telefonu lub klucza	Od planu Osobistego

Do konta można dodać kilka metod naraz - przy logowaniu użytkownik wybiera wygodną. Metody skonfigurowane w planie płatnym pozostają aktywne także po przejściu na plan Darmowy: obniżenie planu nigdy nie osłabia zabezpieczeń konta. Na planie Darmowym nie da się jedynie dodać nowej metody ani wygenerować nowych kodów.

10.2 Zabezpieczenia drugiego składnika

- **Limit prób:** 5 błędnych kodów z aplikacji lub kodów zapasowych blokuje drugi składnik na 15 minut. Licznik jest wspólny, aby kody zapasowe nie były drugą, niezależną ścieżką zgadywania. Do kroku 2FA dochodzi się dopiero po poprawnym haśle, dlatego blokada

wywołuje e-mail z zaleceniem zmiany Hasła Głównego.

- **Ochrona przed powtórzeniem:** ten sam kod z aplikacji nie zostanie przyjęty dwa razy.
- **Kody zapasowe** mają format `xxxx-xxxx`, są jednorazowe i zapisane pod pieczęcią HMAC. Użycie kodu wywołuje e-mail z liczbą pozostałych; gdy zostają najwyżej trzy, aplikacja zaleca wygenerowanie nowego kompletu. Nowy komplet unieważnia poprzedni w całości.
- **Zaufane urządzenie:** po zaznaczeniu „Zaufaj temu urządzeniu" serwer wydaje token zaufania, który zwalnia z kroku 2FA na tym urządzeniu. Token jest rotowany przy każdym logowaniu, a odzyskanie konta i reset konta unieważniają wszystkie zaufane urządzenia.

Czy sekret aplikacji uwierzytelniającej jest objęty zerową wiedzą?

Nie - i to świadoma różnica. Aby sprawdzić kod przy logowaniu, serwer musi znać sekret TOTP konta. Jest on przechowywany zaszyfrowany kluczem aplikacji serwerowej, poza bazą danych. Dotyczy to wyłącznie drugiego składnika logowania do Inlockly™. **Kody 2FA, które zapisujesz w pozycjach skarbca** dla innych serwisów, są częścią zaszyfrowanej treści pozycji i liczone wyłącznie na Twoim urządzeniu.

10.3 Utrata drugiego składnika

⚠ Uwaga

Przechowuj kody zapasowe poza Inlockly™, tak aby były dostępne także wtedy, gdy nie możesz się zalogować. W przypadku kluczy sprzętowych zalecamy dodanie do konta co najmniej dwóch: jednego, który nosisz przy sobie, i drugiego - zapasowego.

Jeśli stracisz dostęp do aplikacji uwierzytelniającej lub klucza, zaloguj się kodem zapasowym, a następnie skonfiguruj metodę ponownie. Zespół wsparcia Inlockly™ nie wyłącza 2FA ręcznie - takie obejście byłoby furtką dla każdego, kto podszyje się pod właściciela konta. Ścieżki odzyskiwania dostępu (klucz odzyskiwania i reset konta) również wymagają drugiego składnika, jeśli konto go ma.

11

Skarbce z danymi

Skarbcem nazywamy zbiór danych zaszyfrowanych tym samym kluczem. Każdy użytkownik po założeniu konta staje się właścicielem **skarbca prywatnego**. **Skarbce grup** powstają w obrębie grup użytkowników (rodzina, zespół, firma), między którymi dane mają być udostępniane.

11.1 Szyfrowanie danych w skarbcach

Pozycje **skarbca prywatnego** są szyfrowane Kluczem Symetrycznym użytkownika (AES-256-GCM). Klucz ten powstaje przy zakładaniu konta i jest chroniony Kluczem Głównym wyprowadzonym z Hasła Głównego.

Skarbiec grupy ma własny, losowy 256-bitowy klucz, wygenerowany przy tworzeniu grupy. Każdy członek otrzymuje własną kopię tego klucza, zaszyfrowaną jego Kluczem Publicznym (RSA-OAEP). Tylko ten członek ma Klucz Prywatny, który odszyfruje przypisaną mu kopię, a dzięki niej - dane skarbca grupy.

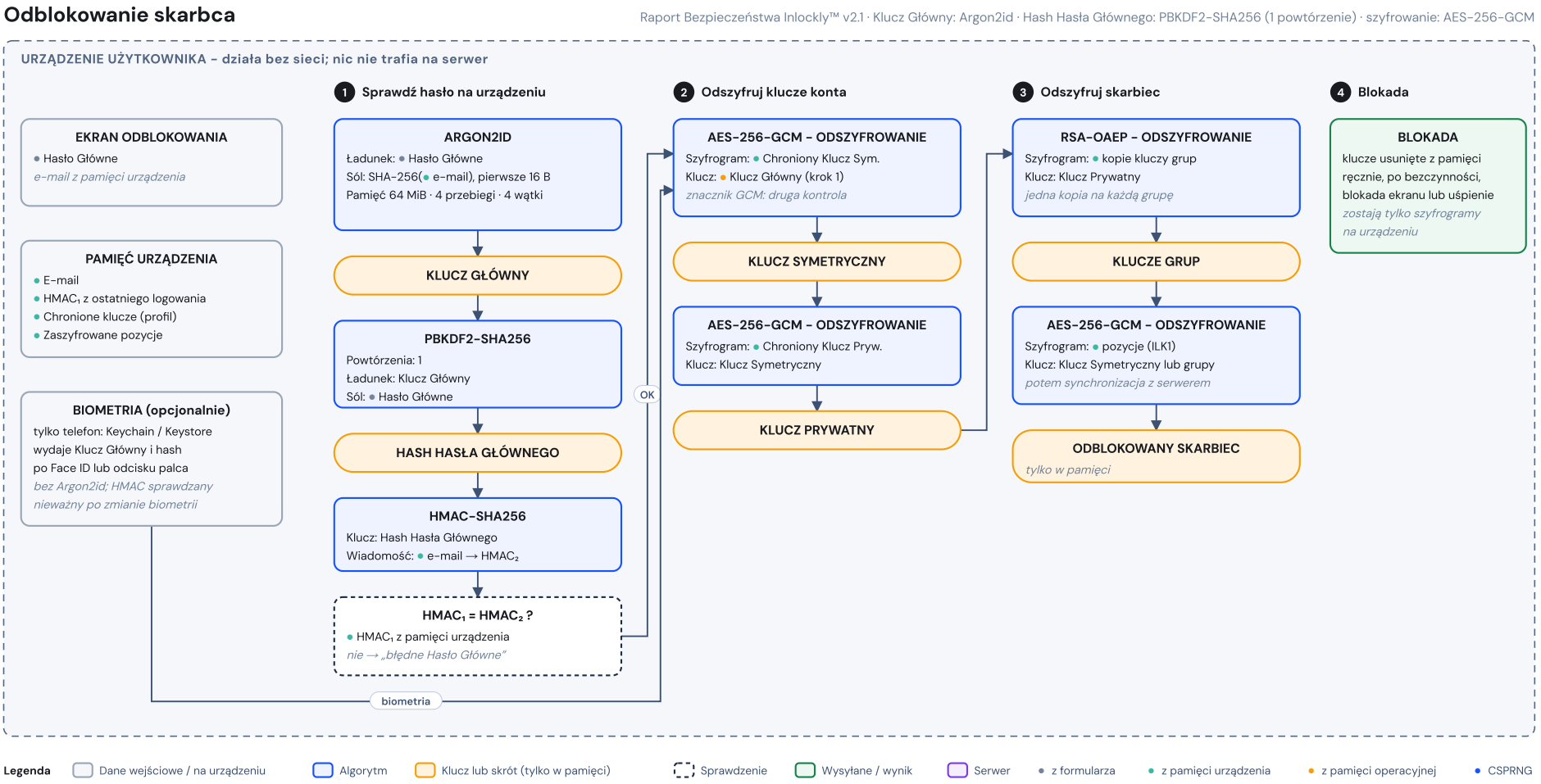
Szyfrowana jest cała treść pozycji (nazwa, login, hasło, adresy, notatki, pola własne, klucze 2FA) oraz jej metadane i historia zmian. Każdy zapis używa nowego wektora inicjalizującego.

11.2 Blokowanie skarbców

Skarbiec blokuje się ręcznie, po czasie braku aktywności (domyślnie 5 minut), przy blokadzie ekranu lub uśpieniu systemu. Blokada nie wylogowuje z konta, ale usuwa z pamięci wszystkie klucze kryptograficzne. Aby ponownie skorzystać z aplikacji, wystarczy podać Hasło Główne (albo - na telefonie - użyć biometrii).

Czas bezczynności można zmienić w preferencjach. Zamiast blokady można też wybrać wylogowanie - wtedy ponowne użycie aplikacji wymaga pełnego logowania, a także drugiego składnika, jeśli jest włączony.

11.3 Diagram procesu odblokowania skarbca



Rysunek 3. Odblokowanie skarbca odbywa się wyłącznie na urządzeniu – działa także bez połączenia z internetem.

11.4 Część opisowa procesu odblokowania skarbca

Zablokowana aplikacja wyświetla formularz, w którym użytkownik wpisuje Hasło Główne. Adres e-mail nie jest wymagany - użytkownik nie został wylogowany, więc adres jest w pamięci urządzenia (na ekranie częściowo zamaskowany). Odblokowanie odbywa się w całości na urządzeniu, bez udziału serwera.

1. **Sprawdzenie hasła na urządzeniu.** Z Hasła Głównego wyprowadzany jest Klucz Główny (Argon2id, sól z adresu e-mail zapisanego na urządzeniu), a z niego Hash Hasła Głównego. Aplikacja oblicza kod HMAC-SHA256 i porównuje go z kodem zapisanym przy ostatnim logowaniu. Jeżeli kody się różnią, informuje o błędnym Haśle Głównym i nie próbuje odszyfrować kluczy. Zapisany kod nie ułatwia odgadnięcia hasła, bo każda próba wymaga pełnego obliczenia Argon2id.
2. **Odszyfrowanie kluczy konta.** Aplikacja odszyfrowuje Chroniony Klucz Symetryczny Kluczem Głównym, a następnie Chroniony Klucz Prywatny Kluczem Symetrycznym. Znacznik uwierzytelniający GCM stanowi drugą, niezależną kontrolę: niewłaściwy klucz lub zmodyfikowany szyfrogram kończy odszyfrowanie błędem.
3. **Odszyfrowanie skarbców.** Kluczem Prywatnym aplikacja odszyfrowuje (RSA-OAEP) kopie kluczy grup, a następnie pozycje zapisane lokalnie: skarbca prywatnego - Kluczem Symetrycznym, skarbców grup - odpowiednim kluczem grupy. Gdy jest połączenie z internetem, aplikacja synchronizuje skarbce z serwerem, przesyłając i pobierając wyłącznie dane zaszyfrowane.
4. **Blokada.** Skarbiec pozostaje odblokowany do ręcznej blokady, upływu czasu bezczynności, blokady ekranu lub uśpienia systemu. Blokada usuwa wszystkie klucze z pamięci.

11.5 Odblokowanie biometryczne

Na telefonie użytkownik może włączyć odblokowanie za pomocą Face ID, Touch ID lub odcisku palca (od planu Osobistego). Po włączeniu Klucz Główny i Hash Hasła Głównego są zapisywane w systemowym, sprzętowo chronionym magazynie kluczy (Keychain w iOS, Keystore w Androidzie) z zabezpieczeniami:

- system wydaje je wyłącznie po pozytywnej weryfikacji biometrycznej, przy każdym użyciu;
- nie trafiają do kopii zapasowych i nie da się ich przenieść na inne urządzenie;

- zostają trwale unieważnione, gdy w systemie zmieni się zestaw zapisanych odcisków palców lub twarzy.

Odblokowanie biometryczne pomija obliczenie Argon2id, ale nadal sprawdza kod HMAC. Po zmianie Hasła Głównego lub adresu e-mail na innym urządzeniu zapisany sekret staje się nieaktualny - aplikacja wykrywa to po niezgodności kodu, wyłącza biometrię i prosi o Hasło Główne. Sekret jest usuwany przy wylogowaniu, wygaśnięciu sesji oraz zmianie hasła lub adresu. To jedyny przypadek, w którym Klucz Główny opuszcza pamięć operacyjną aplikacji - wyłącznie na wyraźne życzenie użytkownika.

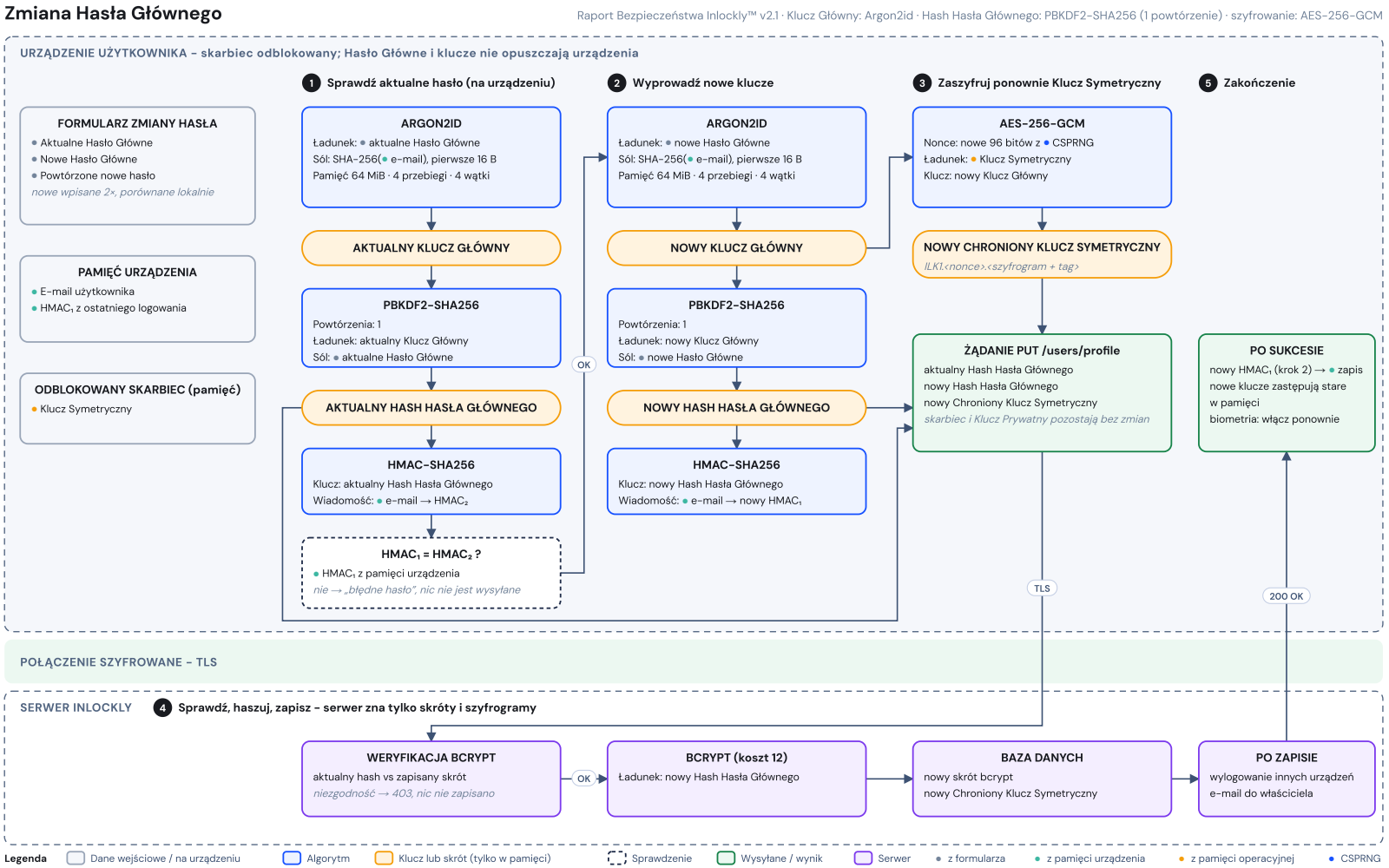
12

Zmiana Hasła Głównego

Zmiana Hasła Głównego jest szybka i bezpieczna niezależnie od wielkości skarbca: ponownie szyfrowany jest wyłącznie Klucz Symetryczny – jedna krótka wartość. Dane w skarbcach, Klucz Prywatny i klucz odzyskiwania pozostają bez zmian.

Aktualne hasło jest sprawdzane dwukrotnie: najpierw na urządzeniu (bez kontaktu z serwerem), a potem na serwerze. Po zmianie pozostałe urządzenia są wylogowywane, a właściciel konta dostaje powiadomienie e-mail.

12.1 Diagram procesu zmiany Hasła Głównego



Rysunek 4. Zmiana Hasła Głównego: ponownie szyfrowany jest tylko Klucz Symetryczny – dane skarbca pozostają bez zmian.

12.2 Część opisowa procesu zmiany Hasła Głównego

W ustawieniach konta użytkownik wypełnia formularz zmiany hasła, podając aktualne Hasło Główne oraz dwukrotnie nowe Hasło Główne. Nowe hasło podaje się dwa razy, aby wyeliminować literówkę - aplikacja porównuje oba pola na urządzeniu. Aktualne hasło potwierdza, że operację wykonuje właściciel konta: najpierw na urządzeniu, a następnie na serwerze. Zmiana odbywa się przy odblokowanym skarbcu, więc Klucz Symetryczny jest już w pamięci i nie jest wyprowadzany ponownie.

1. **Weryfikacja aktualnego hasła na urządzeniu.** Z aktualnego Hasła Głównego powstaje Aktualny Klucz Główny (Argon2id, sól z adresu e-mail zapisanego na urządzeniu), a z niego Aktualny Hash Hasła Głównego. Aplikacja oblicza kod HMAC-SHA256 i porównuje go z kodem zapisanym przy ostatnim logowaniu. Jeżeli kody się różnią, informuje o błędnym haśle, a na serwer nie trafia żadne żądanie.

2. **Wyprowadzenie nowych kluczy.** W ten sam sposób z nowego Hasła Głównego powstaje Nowy Klucz Główny (sól bez zmian, bo adres e-mail się nie zmienia), Nowy Hash Hasła Głównego i nowy kod HMAC.

3. **Ponowne zaszyfrowanie Klucza Symetrycznego.** Klucz Symetryczny jest szyfrowany algorytmem AES-256-GCM Nowym Kluczem Głównym, z nowym wektorem inicjalizującym. Powstaje Nowy Chroniony Klucz Symetryczny - jedyny element, który wymaga ponownego zaszyfrowania. Klucz Prywatny i dane w skarbcach są zaszyfrowane Kluczem Symetrycznym, a ten się nie zmienia.

4. **Weryfikacja i zapis na serwerze.** Połączeniem TLS aplikacja przesyła Aktualny Hash Hasła Głównego, Nowy Hash Hasła Głównego i Nowy Chroniony Klucz Symetryczny. Serwer ponownie autoryzuje operację, porównując Aktualny Hash z zapisanym skrótem BCRYPT; przy niezgodności odrzuca żądanie i niczego nie zapisuje. Po pozytywnej weryfikacji haszuje Nowy Hash funkcją BCRYPT (koszt 12) i zapisuje go razem z Nowym Chronionym Kluczem Symetrycznym. Następnie wylogowuje pozostałe urządzenia i wysyła właścicielowi powiadomienie e-mail. Przez cały proces serwer operuje wyłącznie na skrótach i szyfrogramach.

5. **Zakończenie.** Aplikacja zapisuje na urządzeniu nowy kod HMAC i zastępuje klucze w pamięci nowymi. Jeżeli użytkownik korzystał z odblokowania biometrycznego, zostaje ono wyłączone (zapisany sekret opierał się na poprzednim Kluczu Głównym) - można je ponownie włączyć w ustawieniach.

Klucz odzyskiwania pozostaje ważny

Zmiana Hasła Głównego nie wpływa na klucz odzyskiwania: chroni on ten sam Klucz Symetryczny, który się nie zmienia. Jeśli podejrzewasz, że ktoś poznał także klucz odzyskiwania, wygeneruj nowy w ustawieniach konta - poprzedni przestanie działać.

Czy zmiana hasła cofa skutki wycieku?

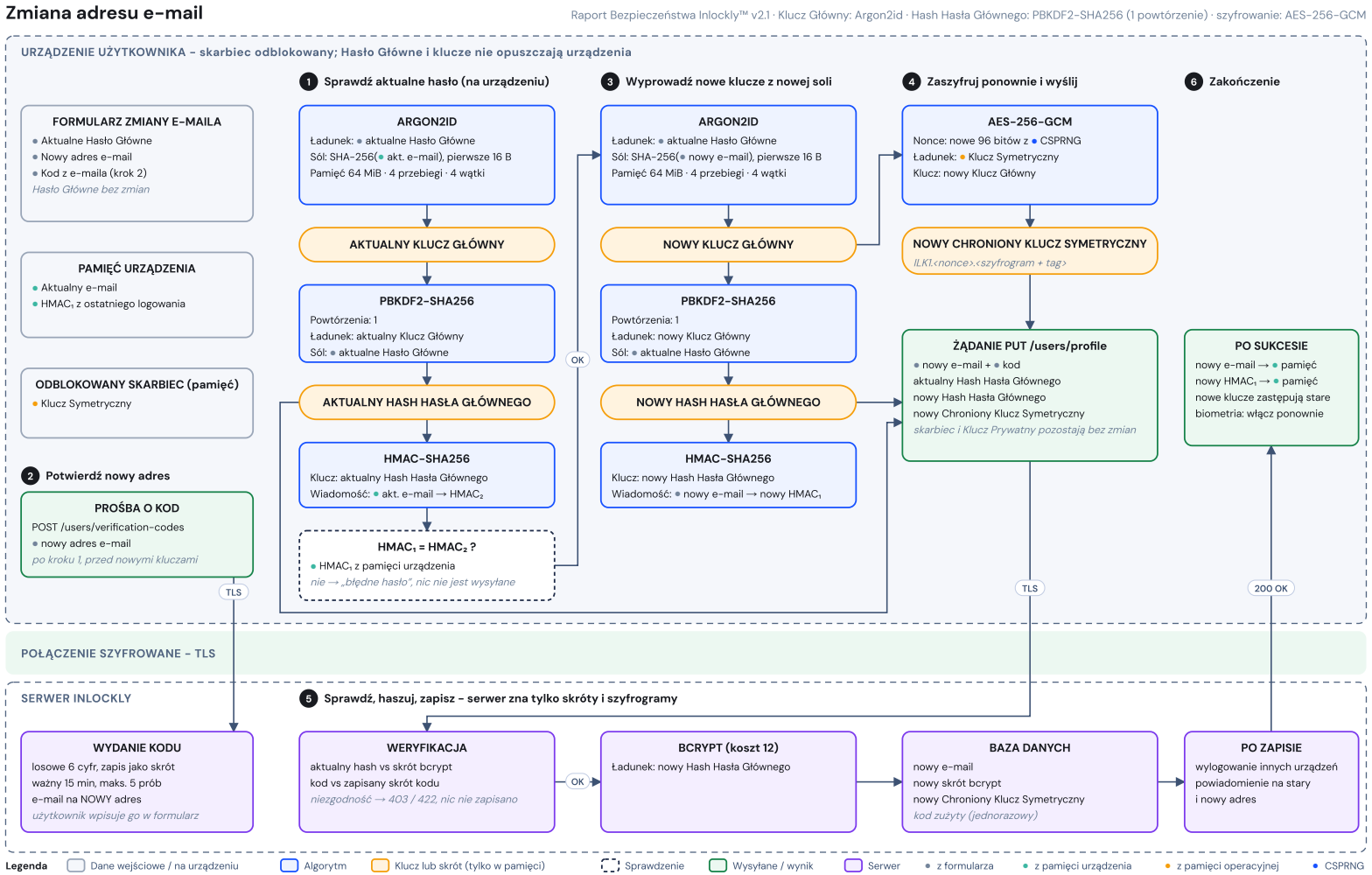
Zmiana hasła odcina dostęp komuś, kto zna tylko stare hasło. Nie jest jednak wymianą klucza, którym zaszyfrowane są dane. Jeśli ktoś miał wcześniej dostęp do odblokowanego skarbca, mógł odczytać jego zawartość - wtedy, oprócz zmiany Hasła Głównego, zmień także hasła przechowywane w skarbcu.

13

Zmiana adresu e-mail

Adres e-mail jest solą funkcji Argon2id, dlatego jego zmiana oznacza nowy Klucz Główny i nowy Hash Hasła Głównego - choć Hasło Główne pozostaje to samo. Nowy adres trzeba potwierdzić kodem, a powiadomienie o zmianie trafia zarówno na stary, jak i na nowy adres.

13.1 Diagram procesu zmiany adresu e-mail



Rysunek 5. Zmiana adresu e-mail: nowy adres to nowa sól funkcji Argon2id, więc powstają nowe klucze.

13.2 Część opisowa procesu zmiany adresu e-mail

W ustawieniach konta użytkownik podaje aktualne Hasło Główne, nowy adres e-mail i kod weryfikacyjny wysłany na nowy adres. Aktualne hasło potwierdza, że operację wykonuje właściciel konta. Zmiana odbywa się przy odblokowanym skarbcu, więc Klucz Symetryczny jest już w pamięci.

Zmiana adresu e-mail jest jednocześnie zmianą kluczy: solą funkcji Argon2id jest skrót adresu e-mail, więc to samo Hasło Główne połączone z nowym adresem daje inny Klucz Główny.

1. **Weryfikacja aktualnego hasła na urządzeniu.** Z Hasła Głównego powstaje Aktualny Klucz Główny (sól z aktualnego adresu), a z niego Aktualny Hash Hasła Głównego. Aplikacja porównuje kod HMAC-SHA256 z kodem zapisanym przy ostatnim logowaniu. Jeżeli kody się różnią, na serwer nie trafia żadne żądanie.

2. **Potwierdzenie nowego adresu.** Aplikacja normalizuje nowy adres i prosi serwer o kod weryfikacyjny. Serwer generuje jednorazowy, 6-cyfrowy kod związany z adresem docelowym, przechowuje wyłącznie jego zabezpieczoną postać i wysyła kod na nowy adres. Kod jest ważny 15 minut i unieważnia się po 5 błędnych próbach. Wpisując go, użytkownik potwierdza dostęp do nowej skrzynki.

3. **Wyprowadzenie nowych kluczy z nowej soli.** Z tego samego Hasła Głównego powstaje Nowy Klucz Główny - tym razem solą jest skrót nowego adresu - a z niego Nowy Hash Hasła Głównego i nowy kod HMAC.

4. **Ponowne zaszyfrowanie Klucza Symetrycznego i wysłanie danych.** Klucz Symetryczny jest szyfrowany (AES-256-GCM) Nowym Kluczem Głównym z nowym wektorem inicjalizującym. Połączeniem TLS aplikacja przesyła w jednym żądaniu: nowy adres, kod weryfikacyjny, Aktualny i Nowy Hash Hasła Głównego oraz Nowy Chroniony Klucz Symetryczny.

5. **Weryfikacja i zapis na serwerze.** Serwer porównuje Aktualny Hash ze skrótem BCRYPT i sprawdza kod. Jeżeli którykolwiek warunek nie jest spełniony, odrzuca żądanie. Po pozytywnej weryfikacji zapisuje nowy adres, nowy skrót hasła i Nowy Chroniony Klucz Symetryczny, zużywa kod, wylogowuje pozostałe urządzenia i wysyła powiadomienie zarówno na stary, jak i na nowy adres - właściciel dowie się o zmianie, nawet jeśli nie wykonał jej osobiście.

6. **Zakończenie.** Aplikacja zapisuje nowy adres i nowy kod HMAC, zastępuje klucze w pamięci nowymi i wyłącza odblokowanie biometryczne. Przy kolejnym logowaniu użytkownik podaje nowy adres i dotychczasowe Hasło Główne.

Klucz odzyskiwania pozostaje ważny

Klucz odzyskiwania chroni Klucz Symetryczny, który się nie zmienia - po zmianie adresu e-mail działa nadal.

14

Klucz odzyskiwania

Klucz odzyskiwania pozwala odzyskać konto po zapomnieniu Hasła Głównego - **bez utraty danych**. Zachowuje skarbiec prywatny, Klucz Prywatny i dostęp do wszystkich grup. Każdy użytkownik dostaje go przy rejestracji, w każdym planie.

Klucz nie przywraca starego hasła. Otwiera drugą, niezależną kopię Klucza Symetrycznego, dzięki czemu użytkownik może ustawić nowe Hasło Główne. Serwer nie zna klucza odzyskiwania - model zerowej wiedzy pozostaje nienaruszony.

14.1 Postać klucza

- **128 losowych bitów** z generatora CSPRNG, wygenerowanych na urządzeniu - nie wybiera ich użytkownik.
- Zapis: **28 znaków Crockford base32**, w siedmiu grupach po cztery, z **12-bitową sumą kontrolną**, która wyłapuje literówki przy przepisywaniu (suma kontrolna nie zwiększa siły klucza).
- Przy wpisywaniu wielkość liter nie ma znaczenia, a mylące znaki są ujednolicane (O jako 0, I i L jako 1).

Dlaczego HKDF, a nie Argon2id?

Argon2id spowalnia odgadywanie haseł wymyślonych przez ludzi. Klucz odzyskiwania to 128 losowych bitów - nie da się go odgadnąć, więc wystarcza funkcja HKDF-SHA256 (RFC 5869), która wyprowadza z niego klucze o rozdzielonym przeznaczeniu.

14.2 Dwa klucze z jednego sekretu

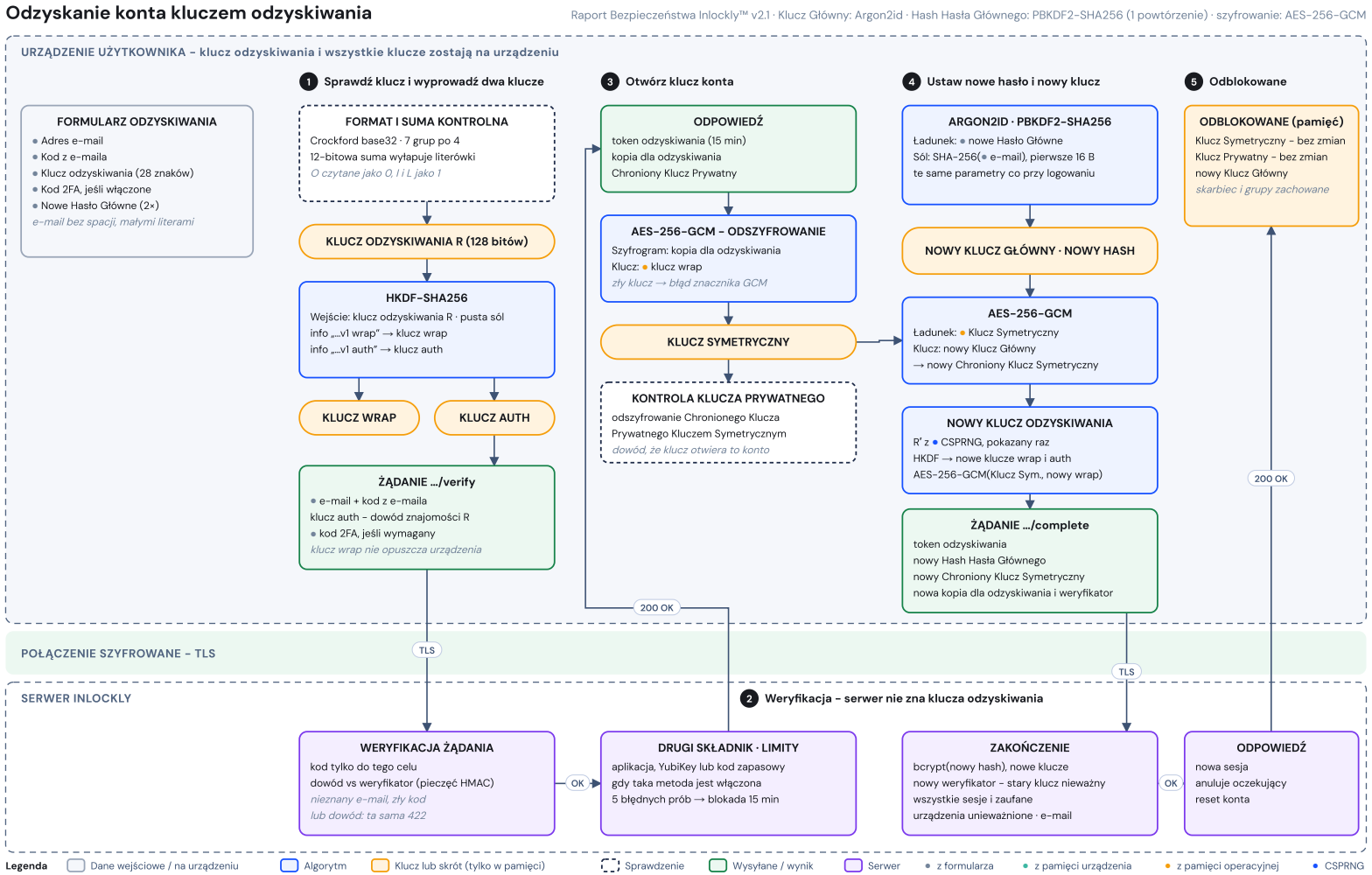
Z klucza odzyskiwania R funkcja HKDF-SHA256 (pusta sól) wyprowadza dwie 256-bitowe wartości o różnym kontekście:

- **klucz wrap** (kontekst „inlockly recovery v1 wrap") - szyfruje AES-256-GCM kopię Klucza Symetrycznego, która trafia na serwer;

- **klucz auth** (kontekst „inlockly recovery v1 auth") – dowód znajomości klucza odzyskiwania. Serwer przechowuje go jako weryfikator pod pieczęcią HMAC i nigdy nie zwraca go w żadnej odpowiedzi.

Rozdzielenie kontekstów sprawia, że dowód wysyłany na serwer nie pozwala odszyfrować kopii klucza konta.

14.3 Diagram procesu odzyskania konta



Rysunek 6. Odzyskanie konta kluczem odzyskiwania: skarbiec, klucze i grupy zostają zachowane.

14.4 Część opisowa procesu odzyskania konta

Na ekranie logowania użytkownik wybiera „Nie pamiętasz hasła?" i podaje adres e-mail (formularz chroniony zabezpieczeniem przed botami). Serwer zawsze odpowiada tak samo, a kod weryfikacyjny wysyła tylko wtedy, gdy konto istnieje i ma klucz odzyskiwania - nie da się w ten sposób sprawdzić, kto ma konto.

1. **Sprawdzenie klucza i wyprowadzenie dwóch kluczy.** Aplikacja sprawdza format i sumę kontrolną klucza, a następnie wyprowadza klucz wrap i klucz auth. Połączeniem TLS wysyła adres, kod z e-maila (przeznaczony wyłącznie do odzyskiwania), klucz auth jako dowód oraz - jeśli konto ma aplikację uwierzytelniającą lub klucz YubiKey - kod drugiego składnika (lub kod zapasowy). Klucz wrap nie opuszcza urządzenia.

2. **Weryfikacja.** Serwer sprawdza kod i porównuje dowód z weryfikatorem. Nieznany adres, zły kod i zły dowód dają tę samą odpowiedź. Pięć błędnych prób blokuje odzyskiwanie dla danego adresu na 15 minut. Po pozytywnej weryfikacji serwer zużywa kod i zwraca krótkotrwały token odzyskiwania (ważny 15 minut), kopię Klucza Symetrycznego dla odzyskiwania i Chroniony Klucz Prywatny - klucze wyłącznie zaszyfrowane.

3. **Otwarcie klucza konta.** Aplikacja odszyfrowuje kopię Klucza Symetrycznego kluczem wrap (AES-256-GCM) i sprawdza, że Klucz Symetryczny rzeczywiście otwiera Klucz Prywatny tego konta.

4. **Nowe hasło i nowy klucz odzyskiwania.** Z nowego Hasła Głównego powstają Nowy Klucz Główny i Nowy Hash Hasła Głównego - tak jak przy zmianie hasła. Klucz Symetryczny zostaje zaszyfrowany Nowym Kluczem Głównym. Aplikacja generuje też **nowy klucz odzyskiwania** (pokazany raz) wraz z nową kopią i nowym weryfikatorem. Wszystko trafia na serwer razem z tokenem odzyskiwania.

5. **Zapis i nowa sesja.** Serwer w jednej transakcji zapisuje nowy skrót hasła i nowe szyfrogramy, zastępuje weryfikator (stary klucz odzyskiwania przestaje działać), unieważnia wszystkie sesje i zaufane urządzenia, anuluje ewentualny oczekujący reset konta i wysyła e-mail o odzyskaniu konta ze szczegółami próby. Aplikacja dostaje nową sesję i od razu odblokowuje skarbiec.

> ⚠ **Przechowuj klucz odzyskiwania poza Inlockly™**
>
> Klucz zapisany wyłącznie w skarbcu nie pomoże, gdy zapomnisz hasła - skarbiec będzie wtedy zamknięty. Wydrukuj go albo zapisz na nośniku przechowywanym w bezpiecznym miejscu. Nigdy nie przesyłaj go pracownikom Inlockly™, administratorowi ani w zgłoszeniu do pomocy technicznej.

14.5 Wymiana klucza i jego granice

- Nowy klucz można wygenerować w ustawieniach konta (wymaga Hasła Głównego) - poprzedni przestaje działać w całości, a właściciel dostaje e-mail bezpieczeństwa.
- Każde użycie klucza kończy się wydaniem nowego - stary działa raz.
- Klucz odzyskiwania **nie zastępuje drugiego składnika ani dostępu do skrzynki e-mail** - odzyskanie wymaga obu, jeśli konto ma włączone 2FA. Dlatego kody zapasowe przechowuj tak, by nie zależały od zablokowanego konta.
- Wymiana klucza usuwa jego ścieżkę w usłudze, ale nie cofa wcześniejszego ujawnienia: jeśli ktoś miał jednocześnie klucz odzyskiwania i skopiowane wcześniej szyfrogramy, mógł je odczytać poza usługą.

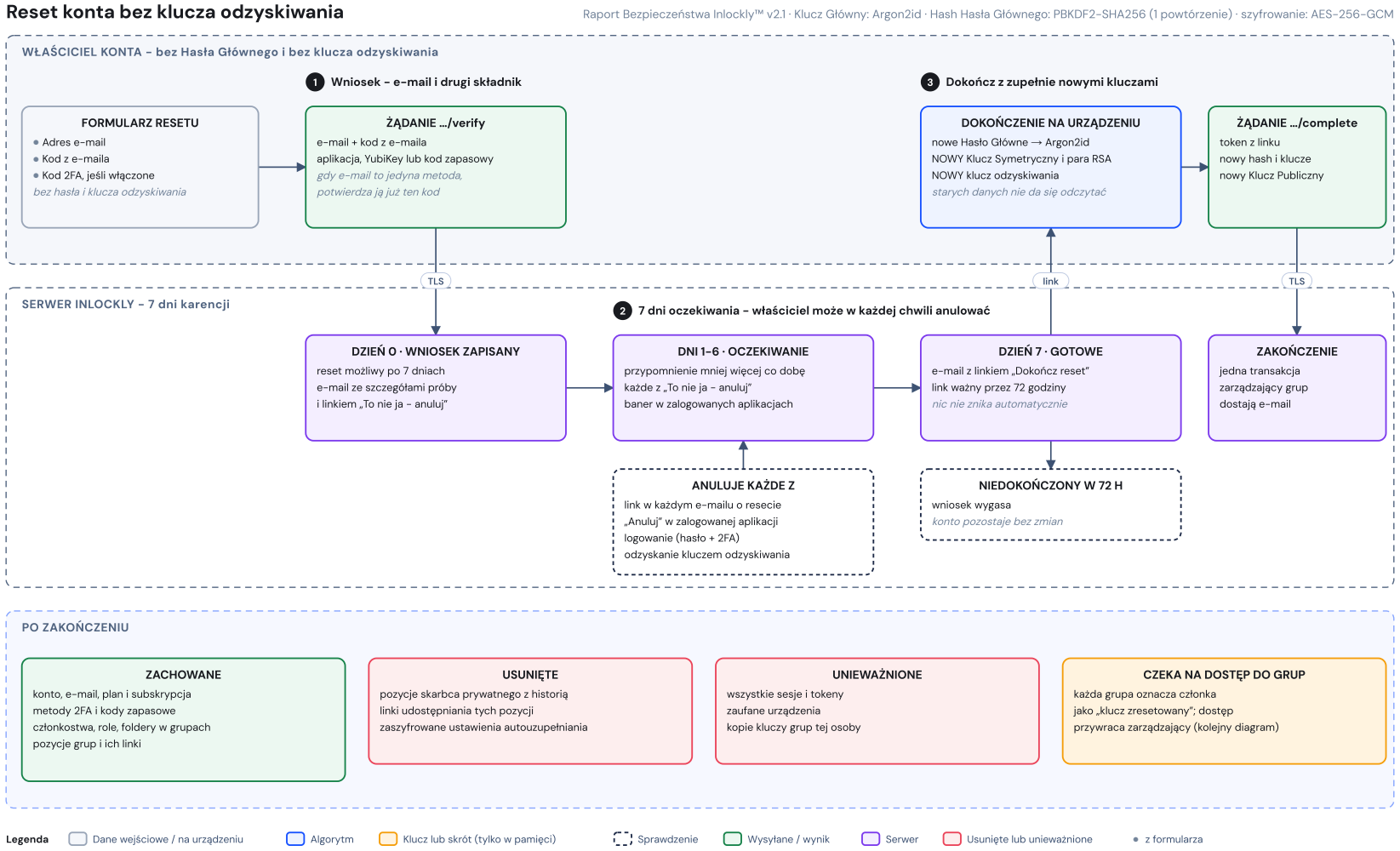
15

Reset konta bez klucza odzyskiwania

Reset konta to ścieżka dla osoby, która nie zna Hasła Głównego **i** nie ma klucza odzyskiwania. Nie jest odzyskaniem danych: skarbca prywatnego nie da się odszyfrować, bo nikt - także Inlockly™ - nie ma do niego klucza. Reset pozwala natomiast zachować konto, adres e-mail, plan i subskrypcję oraz członkostwa w grupach, tworząc zupełnie nowy zestaw kluczy. Dostęp do danych grup przywraca potem administrator grupy (rozdział „Przywrócenie dostępu do grup").

Reset jest operacją nieodwracalną dla skarbca prywatnego, dlatego chroni go **siedmiodniowa karencja** z codziennymi powiadomieniami. Skrzynka e-mail i drugi składnik mogą przepaść razem - na przykład wraz ze skradzionym telefonem - a karencja daje właścicielowi czas na reakcję.

15.1 Diagram procesu resetu konta



Rysunek 7. Reset konta: wniosek, 7 dni karencji z możliwością anulowania i dokończenie z nowymi kluczami.

15.2 Część opisowa procesu resetu konta

1. **Wniosek.** Użytkownik podaje adres e-mail (formularz chroniony zabezpieczeniem przed botami) i kod wysłany na ten adres. Jeśli konto ma aplikację uwierzytelniającą lub klucz YubiKey, podaje także kod drugiego składnika albo kod zapasowy; gdy jedyną metodą 2FA jest e-mail, potwierdza ją już kod z tego kroku. Nieznany adres, zły kod i zły drugi składnik dają tę samą odpowiedź, a błędne próby blokują ścieżkę tak samo jak przy odzyskiwaniu. Zespół wsparcia nie może złożyć wniosku za użytkownika.

2. **Karencja - 7 dni.** Serwer zapisuje wniosek i od razu wysyła e-mail ze szczegółami próby i linkiem „To nie ja - anuluj". Przez kolejne dni wysyła przypomnienie mniej więcej raz na dobę, a zalogowane aplikacje pokazują baner z przyciskiem anulowania. Wniosek anuluje każde z poniższych:
   - link w dowolnym e-mailu o resecie,
   - „Anuluj" w zalogowanej aplikacji,
   - udane logowanie (hasło i drugi składnik) - kto zna hasło, nie potrzebuje resetu,
   - odzyskanie konta kluczem odzyskiwania.

3. **Dokończenie na urządzeniu.** Po 7 dniach użytkownik dostaje e-mail z linkiem „Dokończ reset", ważnym 72 godziny. **Nic nie dzieje się automatycznie** - jeśli link nie zostanie użyty, wniosek wygasa, a konto pozostaje nietknięte. Po kliknięciu aplikacja tworzy od podstaw wszystkie klucze: nowe Hasło Główne i Klucz Główny, nowy Klucz Symetryczny, nową parę kluczy RSA i nowy klucz odzyskiwania (pokazany raz). Wszystko powstaje na urządzeniu, tak jak przy rejestracji.

4. **Zakończenie na serwerze.** W jednej transakcji serwer zapisuje nowy skrót hasła i nowe klucze, usuwa dane zaszyfrowane starymi kluczami, unieważnia sesje i oznacza członkostwa w grupach jako oczekujące na przywrócenie dostępu. Wysyła e-mail o zakończeniu resetu, a zarządzający każdą grupą dostają wiadomość z prośbą o potwierdzenie tożsamości użytkownika przed przywróceniem dostępu.

15.3 Skutki resetu

Obszar	Wynik
Konto, adres e-mail, plan, subskrypcja	Zachowane

Obszar	Wynik
Metody 2FA i kody zapasowe	Zachowane - reset nie jest sposobem na obejście 2FA
Członkostwa w grupach, role, działy, foldery, własność grupy	Zachowane
Pozycje grup i linki udostępniające je	Nienaruszone (zaszyfrowane kluczami grup)
Pozycje skarbca prywatnego z historią zmian	Usunięte
Linki udostępniania pozycji prywatnych, ustawienia autouzupełniania	Usunięte
Sesje, tokeny i zaufane urządzenia	Unieważnione
Kopie kluczy grup tej osoby	Usunięte - czekają na przywrócenie przez zarządzającego

Synchronizacja wymusza na wszystkich urządzeniach pełne pobranie stanu, bo kopie lokalne są zaszyfrowane starymi kluczami. Usunięcie danych z bazy usługi nie oznacza natychmiastowego usunięcia z kopii zapasowych infrastruktury - te podlegają odrębnej retencji, a i tak zawierają wyłącznie szyfrogramy.

Kopia zapasowa skarbca

Jeśli wcześniej wykonałeś zaszyfrowaną kopię zapasową skarbca (plik `.inlockly` z osobnym hasłem), możesz ją zaimportować po resecie. To niezależna ścieżka - reset jej nie tworzy ani nie usuwa.

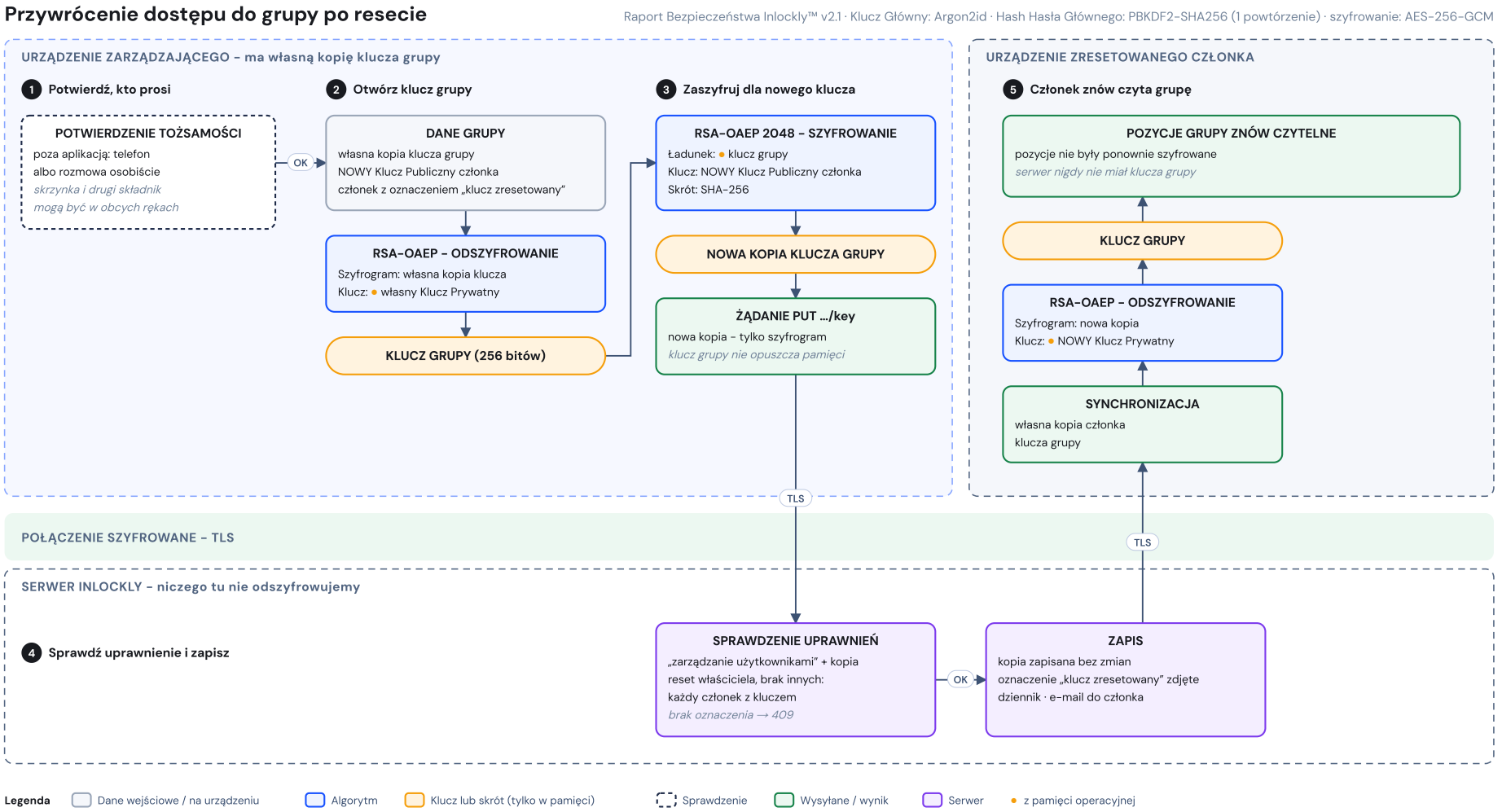
16

Przywrócenie dostępu do grup po resecie

Po resecie użytkownik nadal należy do swoich grup, z tymi samymi uprawnieniami, działami i folderami. Jego nowy Klucz Prywatny nie otwiera jednak kopii klucza grupy zaszyfrowanej dla poprzedniej pary kluczy - ta kopia została usunięta. Dostęp przywraca inna osoba, która ma własną, działającą kopię klucza grupy.

Serwer nie bierze w tym udziału kryptograficznie: niczego nie odszyfrowuje i nie generuje klucza zastępczego. Administrator nie dostaje też dostępu do skarbca prywatnego użytkownika - ten przepadł w resecie i nikt nie może go odtworzyć.

16.1 Diagram procesu przywrócenia dostępu



Rysunek 8. Przywrócenie dostępu do grupy: zarządzający szyfruje klucz grupy nowym Kluczem Publicznym członka.

16.2 Część opisowa procesu przywrócenia dostępu

1. **Potwierdzenie tożsamości.** Zanim zarządzający przywróci dostęp, powinien potwierdzić tożsamość osoby poza aplikacją - telefonicznie albo w rozmowie osobistej, kanałem znanym organizacji wcześniej. Osoba, która przejęła skrzynkę e-mail i drugi składnik, może dla usługi wyglądać jak prawowity właściciel konta. Karencja i powiadomienia zmniejszają to ryzyko, ale go nie eliminują.

2. **Otwarcie klucza grupy.** Aplikacja zarządzającego odszyfrowuje (RSA-OAEP) własną kopię klucza grupy swoim Kluczem Prywatnym.

3. **Zaszyfrowanie dla nowego klucza.** Klucz grupy jest szyfrowany (RSA-OAEP 2048, SHA-256) **nowym** Kluczem Publicznym członka. Powstała kopia trafia na serwer - klucz grupy nie opuszcza pamięci.

4. **Sprawdzenie uprawnienia i zapis.** Serwer sprawdza, czy osoba przywracająca ma uprawnienie zarządzania użytkownikami i własną kopię klucza, a członek rzeczywiście czeka na przywrócenie (to nie jest sposób na podmianę kluczy). Zapisuje kopię bez zmian, zdejmuje oznaczenie, odnotowuje operację w dzienniku aktywności grupy i wysyła członkowi e-mail.

5. **Członek znów czyta grupę.** Przy synchronizacji aplikacja członka odszyfrowuje nową kopię swoim nowym Kluczem Prywatnym. Pozycje grupy nie były ponownie szyfrowane - są czytelne od razu.

> ⚠ Wyjątek: reset właściciela grupy
>
> Jeżeli zresetował się **właściciel** grupy, a żaden inny członek z kluczem nie ma uprawnienia zarządzania użytkownikami, dostęp właścicielowi może przywrócić **każdy potwierdzony członek, który ma klucz grupy**. Bez tego grupa zostałaby na zawsze bez zarządzającego (własność przekazuje tylko właściciel). Uwzględnij to w procedurach organizacji. Jeśli zresetowany był jedynym członkiem z kluczem grupy, danych grupy nie da się już odczytać.

17

Porównanie ścieżek odzyskiwania dostępu

Sytuacja	Co zrobić	Skarbiec prywatny	Grupy	Wym
Pamiętasz Hasło Główne, zgubiłeś telefon z aplikacją 2FA	Logowanie kodem zapasowym	Zachowany	Zachowane	Hasło zapas
Zapomniałeś Hasła Głównego, masz klucz odzyskiwania	Odzyskanie konta kluczem	Zachowany	Zachowane	Klucz e-ma 2FA (włącz
Nie masz hasła ani klucza odzyskiwania	Reset konta	Usunięty	Po przywróceniu przez zarządzającego	Kod e + 2FA włącz + 7 d
Straciłeś hasło, klucz i drugi składnik	Brak ścieżki samoobsługowej	-	-	-

Dlaczego nie ma ścieżki, w której firma odzyskuje konto pracownika?

Niektóre menedżery haseł pozwalają organizacji odzyskać konto pracownika, co oznacza, że administrator ma techniczną możliwość odczytania jego skarbca prywatnego. W Inlockly™ świadomie z tego zrezygnowaliśmy: administrator może przywrócić wyłącznie dostęp do danych **grupy**, do których użytkownik i tak miał prawo. Skarbiec prywatny pozostaje prywatny - także wobec pracodawcy.

18

Udostępnianie danych

Inlockly™ chroni nie tylko dane pojedynczego użytkownika, ale także wymianę poufnych informacji z innymi. Zamiast przesyłać hasła w wiadomościach, użytkownicy udostępniają je we wspólnych skarbcach grup albo zaszyfrowanym linkiem z terminem ważności.

18.1 Rodzaje grup

Grupa	Dla kogo	Najważniejsze cechy
Rodzina	Bliscy, do 5 osób	Wspólny skarbiec, każdy ma też własny skarbiec prywatny
Zespół	Małe zespoły, od 5 do 10 miejsc	Foldery, uprawnienia, dziennik aktywności
Firma	Organizacje, bez górnego limitu miejsc	Działy, uprawnienia do folderów, role, rozbudowany dziennik

Każdy członek grupy zachowuje własny skarbiec prywatny, do którego nie mają dostępu inni członkowie ani administratorzy. Członkowie grupy firmowej mogą dodatkowo utworzyć własną grupę rodzinną.

18.2 Szyfrowanie w grupach

Skarbiec grupy szyfruje jej własny, losowy 256-bitowy klucz. Gdy nowa osoba dołącza do grupy, członek z odpowiednim uprawnieniem zatwierdza ją, a jego aplikacja szyfruje klucz grupy Kluczem Publicznym nowej osoby (RSA-OAEP) i przesyła tylko tę zaszyfrowaną kopię. Serwer nigdy nie zna klucza grupy.

Uprawnienia (kto widzi który folder, kto może edytować, udostępniać czy zarządzać członkami) egzekwuje serwer. Grupa jest zawsze wyznaczana po stronie serwera z zasobu, na którym działa żądanie - uprawnienie w jednej grupie nie pozwala na operacje w innej.

18.3 Dołączanie do grup

Do grupy dołącza się kodem dołączenia albo linkiem z zaproszenia wysłanego e-mailem przez członka z odpowiednim uprawnieniem. Kody i zaproszenia mają ograniczony czas ważności. Dołączenie wymaga zatwierdzenia

- dopiero wtedy nowy członek otrzymuje zaszyfrowaną kopię klucza grupy.

18.4 Udostępnianie linkiem

Pojedynczą pozycję można przekazać osobie spoza konta zaszyfrowanym linkiem (od planu Osobistego):

- treść pozycji jest szyfrowana (AES-256-GCM) **nowym, losowym kluczem** tylko dla tego linku;
- klucz jest częścią adresu **po znaku** `#` - przeglądarka nie wysyła tej części na serwer, więc serwer przechowuje tylko szyfrogram;
- link działa od godziny do 30 dni albo **jednorazowo** - wygasa po pierwszym otwarciu;
- dostęp można ograniczyć do wskazanych adresów e-mail - odbiorca potwierdza wtedy adres kodem;
- po wygaśnięciu lub zużyciu linku serwer usuwa szyfrogram, a w historii zostaje jedynie informacja o linku (bez treści); link można też w każdej chwili cofnąć.

18.5 Kopia zapasowa skarbca

Eksport do pliku `.inlockly` tworzy zaszyfrowaną kopię zapasową z **osobnym hasłem**: klucz powstaje funkcją Argon2id z losową solą i parametrami zapisanymi w pliku, a treść jest szyfrowana AES-256-GCM. Kopia otwiera się niezależnie od konta. Złe hasło i uszkodzony plik są nierozróżnialne.

> ⚠ Uwaga
>
> Eksport do niezaszyfrowanego pliku (np. CSV) zawiera wszystkie hasła w jawnej postaci. Używaj go tylko do przeniesienia danych i usuń plik zaraz po imporcie.

19

Funkcje pomocnicze a prywatność

Część funkcji aplikacji potrzebuje danych ze skarbca. Projektujemy je tak, aby te dane nie opuszczały urządzenia.

19.1 SafetyMeter™

SafetyMeter™ ocenia siłę haseł, wskazuje hasła powtórzone, witryny bez szyfrowanego połączenia oraz konta, na których warto włączyć 2FA. Sprawdzenie haseł w znanych wyciekach odbywa się **na urządzeniu**: aplikacja porównuje hasła z dołączoną listą około 100 tysięcy haseł z publicznych wycieków. Nie wysyłamy haseł ani ich skrótów do zewnętrznych usług. Wyniki są liczone w pamięci i nie trafiają na serwer.

19.2 Kody 2FA w pozycjach

Klucz 2FA zapisany przy loginie jest częścią zaszyfrowanej treści pozycji. Kod jednorazowy (RFC 6238) liczy aplikacja na urządzeniu - serwer nie zna klucza ani kodów.

19.3 Autouzupełnianie

Rozszerzenie przeglądarki i systemowe autouzupełnianie na telefonie wypełniają formularze lokalnie. Ustawienia autouzupełniania dla poszczególnych witryn są zaszyfrowane Kluczem Symetrycznym. Systemowe autouzupełnianie na telefonie odblokowuje skarbiec tą samą funkcją Argon2id co aplikacja.

19.4 Ikony witryn

Aplikacja pokazuje przy pozycjach ikony witryn. Pobiera je przez serwer pośredniczący Inlockly™ zamiast bezpośrednio z zewnętrznych usług, dzięki czemu lista Twoich witryn nie trafia do firm trzecich. Serwer pośredniczący poznaje jedynie nazwę witryny, o której ikonę prosi aplikacja. Ikony są zapamiętywane na urządzeniu, a w preferencjach można je całkowicie wyłączyć - wtedy aplikacja nie wysyła żadnych żądań.

© 2026 Inlockly™ · inlockly.com

19.5 Historia zmian pozycji

Poprzednie wersje pozycji są przechowywane zaszyfrowane tym samym kluczem co pozycja. Serwer zna jedynie czas i autora zmiany, potrzebne do synchronizacji i dziennika aktywności. Okres przechowywania historii zależy od planu.

20

Model zagrożeń i granice ochrony

Żaden system nie chroni przed wszystkim. Poniżej opisujemy, przed czym chroni Inlockly™, a w jakich sytuacjach ochrona zależy od użytkownika i jego urządzenia.

20.1 Przed czym chroni Inlockly™

Zagrożenie	Dlaczego dane pozostają bezpieczne
Wyciek bazy danych lub kopii zapasowej serwera	Baza zawiera szyfrogramy i skróty BCRYPT; klucze powstają wyłącznie z Hasła Głównego lub klucza odzyskiwania
Przejęcie serwera lub nieuczciwy pracownik	Serwer nie zna kluczy; może zablokować usługę, ale nie odczyta skarbców
Podsłuch połączenia	TLS, a pod nim dane już zaszyfrowane na urządzeniu
Podmiana zaszyfrowanych danych	Znacznik uwierzytelniający GCM – zmieniony szyfrogram nie zostanie przyjęty
Odgadywanie Hasła Głównego z wykradzionych danych	Argon2id (64 MiB na próbę) i BCRYPT na serwerze
Logowanie cudzym hasłem	Drugi składnik, alerty o nowym urządzeniu, limity prób
Sprawdzanie, czy ktoś ma konto	Te same odpowiedzi i czasy dla istniejących i nieistniejących adresów
Obejście 2FA wpisem do bazy	Pieczęć HMAC na wszystkim, co decyduje o dostępie

Zagrożenie	Dlaczego dane pozostają bezpieczne
Przejęcie skrzynki i drugiego składnika w celu resetu	7 dni karencji, codzienne powiadomienia i anulowanie jednym kliknięciem

20.2 Granice ochrony

- **Zainfekowane urządzenie.** Złośliwe oprogramowanie z pełnym dostępem do odblokowanego urządzenia (np. keylogger) może przechwycić Hasło Główne albo odczytać dane z odblokowanego skarbca. Dbaj o aktualność systemu i nie instaluj programów z niepewnych źródeł.
- **Słabe lub powtórzone Hasło Główne.** Argon2id zwiększa koszt ataku, ale nie zastąpi dobrego hasła.
- **Wyłudzenie (phishing).** Fałszywa strona może nakłonić do wpisania Hasła Głównego. Rozszerzenie Inlockly™ podpowiada dane tylko na właściwej domenie - jeśli podpowiedź się nie pojawia, sprawdź adres.
- **Utrata wszystkich czynników.** Bez Hasła Głównego, klucza odzyskiwania i drugiego składnika nie ma samoobsługowej ścieżki odzyskania konta.
- **Metadane.** Serwer zna informacje administracyjne opisane w rozdziale „Co widzi serwer" - w tym nazwy grup i folderów, czas zmian oraz adresy IP urządzeń.
- **Osoby z dostępem do grupy.** Członek grupy może odczytać i skopiować dane, do których ma dostęp. Odebranie dostępu nie usuwa kopii, które zdążył wykonać - po odejściu osoby zmień udostępnione jej hasła.
- **Sekret TOTP konta.** Drugi składnik w aplikacji uwierzytelniającej wymaga, by serwer znał jego sekret (szyfrowany kluczem serwera). Nie wpływa to na poufność skarbca.

21

Zalecenia

21.1 Dla użytkowników

- Wybierz długie, unikalne Hasło Główne i nie używaj go nigdzie indziej.
- Zachowaj klucz odzyskiwania poza Inlockly™ - wydrukowany albo na nośniku w bezpiecznym miejscu.
- Włącz uwierzytelnianie dwuskładnikowe - najlepiej aplikację uwierzytelniającą lub klucz YubiKey - i przechowuj kody zapasowe niezależnie od telefonu.
- Reaguj na e-maile bezpieczeństwa: nowe logowanie, wniosek o reset, użycie kodu zapasowego. Jeśli to nie Ty - anuluj wniosek i zmień Hasło Główne.
- Ustaw krótki czas automatycznej blokady na urządzeniach, z których korzystają też inni.
- Regularnie sprawdzaj SafetyMeter™ i zmieniaj hasła słabe, powtórzone i ujawnione w wyciekach.

21.2 Dla administratorów grup i firm

- Opracuj procedurę **potwierdzania tożsamości** przed przywróceniem dostępu po resecie konta - poza aplikacją, kanałem znanym wcześniej.
- Upewnij się, że **co najmniej dwie osoby** mają uprawnienie zarządzania użytkownikami i kopię klucza grupy. Reset jedynego zarządzającego utrudnia przywrócenie dostępu, a reset jedynej osoby z kluczem oznacza utratę danych grupy.
- Nadawaj uprawnienia według potrzeb: dostęp do folderów i działów tylko tym, którzy go potrzebują.
- Nie umieszczaj poufnych informacji w nazwach grup, działów i folderów - nie są szyfrowane.
- Po odejściu pracownika usuń go z grupy i zmień hasła, które były mu udostępnione.
- Przeglądaj dziennik aktywności grupy, szczególnie po resecie konta członka.
- Przypominaj członkom o zachowaniu klucza odzyskiwania - to najprostszy sposób, by nie tracić danych.

22

Słownik pojęć

Pojęcie	Znaczenie
Hasło Główne	Hasło znane tylko użytkownikowi; nigdy nie trafia na serwer
Klucz Główny	256-bitowy klucz wyprowadzony z Hasła Głównego funkcją Argon2id; tylko w pamięci urządzenia
Hash Hasła Głównego	Jednokierunkowy skrót Klucza Głównego wysyłany na serwer do uwierzytelnienia
Klucz Symetryczny	Losowy, 256-bitowy klucz konta; szyfruje skarbiec prywatny i Klucz Prywatny
Chroniony Klucz Symetryczny	Klucz Symetryczny zaszyfrowany Kluczem Głównym; przechowywany na serwerze
Para Kluczy RSA	Klucz Publiczny (jawny) i Klucz Prywatny; służą do przekazywania kluczy grup
Chroniony Klucz Prywatny	Klucz Prywatny zaszyfrowany Kluczem Symetrycznym; przechowywany na serwerze
Klucz grupy	Losowy, 256-bitowy klucz skarbca grupy; każdy członek ma własną zaszyfrowaną kopię
Klucz odzyskiwania	128 losowych bitów (28 znaków) do odzyskania konta po zapomnieniu Hasła Głównego
Klucz wrap / klucz auth	Klucze wyprowadzone z klucza odzyskiwania: pierwszy szyfruje kopię Klucza Symetrycznego, drugi jest dowodem dla serwera

Pojęcie	Znaczenie
Weryfikator	Wartość, po której serwer rozpoznaje poprawny klucz odzyskiwania, nie znając klucza
Skarbiec	Zbiór pozycji zaszyfrowanych tym samym kluczem: prywatny lub grupy
Pozycja	Pojedynczy wpis w skarbcu: login, karta, notatka itp.
Nonce (wektor inicjalizujący)	Losowa, jednorazowa wartość używana przy każdym szyfrowaniu AES-GCM
Znacznik uwierzytelniający	128-bitowa wartość GCM wykrywająca każdą zmianę szyfrogramu
Pieczęć HMAC	Podpis serwera chroniący dane decydujące o dostępie przed modyfikacją w bazie
Token sesji	Identyfikator zalogowanego urządzenia
Token zaufania	Zwalnia z kroku 2FA na zaufanym urządzeniu
Karencja	7 dni między wnioskiem o reset konta a możliwością jego dokończenia
CSPRNG	Kryptograficznie bezpieczny generator liczb losowych
Zero-knowledge (zerowa wiedza)	Model, w którym usługodawca nie ma kluczy do danych użytkownika

© 2026 Inlockly™ · inlockly.com

23

Standardy i źródła

- NIST FIPS PUB 197 - Advanced Encryption Standard (AES), https://csrc.nist.gov/pubs/fips/197/final
- NIST SP 800-38D - Galois/Counter Mode (GCM) and GMAC, https://csrc.nist.gov/pubs/sp/800/38/d/final
- ISO/IEC 18033-3 - Encryption algorithms, Part 3: Block ciphers
- IETF RFC 9106 - Argon2 Memory-Hard Function for Password Hashing, https://www.rfc-editor.org/rfc/rfc9106
- IETF RFC 8018 - PKCS #5: Password-Based Cryptography (PBKDF2), https://www.rfc-editor.org/rfc/rfc8018
- IETF RFC 5869 - HMAC-based Extract-and-Expand Key Derivation Function (HKDF), https://www.rfc-editor.org/rfc/rfc5869
- IETF RFC 2104 - HMAC: Keyed-Hashing for Message Authentication, https://www.rfc-editor.org/rfc/rfc2104
- IETF RFC 8017 - PKCS #1: RSA Cryptography Specifications v2.2, https://www.rfc-editor.org/rfc/rfc8017
- IETF RFC 6238 - TOTP: Time-Based One-Time Password Algorithm, https://www.rfc-editor.org/rfc/rfc6238
- IETF RFC 8446 - The Transport Layer Security (TLS) Protocol Version 1.3, https://www.rfc-editor.org/rfc/rfc8446
- Crockford's Base32, https://www.crockford.com/base32.html
- N. Provos, D. Mazières, „A Future-Adaptable Password Scheme" (BCRYPT), USENIX 1999
- OWASP Password Storage Cheat Sheet, https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html
- NCSC, „Most hacked passwords revealed as UK cyber survey exposes gaps in online security", https://www.ncsc.gov.uk/news/most-hacked-passwords-revealed-as-uk-cyber-survey-exposes-gaps-in-online-security
- W3C Web Cryptography API, https://www.w3.org/TR/WebCryptoAPI/

24

Historia zmian

24.1 v2.1 - 5 października 2026

Raport opisuje nowe ścieżki odzyskiwania dostępu i ma nową formę: wersję na stronie internetowej oraz PDF generowany z tej samej treści. Fundamentalne zasady ochrony danych i schemat kryptograficzny z wersji 2.0 pozostają bez zmian.

- **Klucz odzyskiwania** (nowy rozdział i diagram): 128-bitowy klucz tworzony przy rejestracji pozwala odzyskać konto po zapomnieniu Hasła Głównego bez utraty danych. Usunęliśmy zapisy o braku możliwości odzyskania dostępu.
- **Reset konta bez klucza odzyskiwania** (nowy rozdział i diagram): 7-dniowa karencja z powiadomieniami i możliwością anulowania; skarbiec prywatny jest usuwany, konto, plan i członkostwa zostają.
- **Przywrócenie dostępu do grup po resecie** (nowy rozdział i diagram) przez zarządzającego grupą, wraz z zasadą potwierdzania tożsamości i wyjątkiem dla właściciela grupy.
- **Uwierzytelnianie dwuskładnikowe:** dodane aplikacje uwierzytelniające (TOTP) i 10 kodów zapasowych, opis limitów prób, ochrony przed powtórzeniem i dostępności metod w planach.
- Diagram tworzenia konta uzupełniony o generowanie klucza odzyskiwania; opisy logowania zaktualizowane (tokeny sesji i zaufania pod pieczęcią HMAC, anulowanie oczekującego resetu).
- Nowe rozdziały: „Najważniejsze w skrócie", „Co widzi serwer, a czego nie", zestawienie parametrów kryptograficznych i hierarchia kluczy, udostępnianie linkiem i kopia zapasowa, funkcje pomocnicze a prywatność, model zagrożeń, zalecenia dla użytkowników i administratorów, słownik pojęć, standardy i źródła.
- Rozdział o infrastrukturze ograniczony do stanu bieżącego (serwery w Unii Europejskiej); szczegółowy opis architektury opublikujemy w kolejnej wersji.
- Diagramy w wersji polskiej i angielskiej, generowane z jednego modelu.

24.2 v2.0 - 24 września 2026

Druga wersja Raportu Bezpieczeństwa Inlockly™. Fundamentalne zasady ochrony danych pozostają bez zmian, zmieniła się ich realizacja kryptograficzna. Nowy schemat nie jest zgodny ze schematem opisanym w wersji 1.0.

- Klucz Główny wyprowadzamy funkcją Argon2id (64 MiB, 4 przebiegi, 4 wątki) zamiast PBKDF2-SHA256. Solą jest skrót SHA-256 adresu e-mail. PBKDF2 służy już wyłącznie do wyprowadzenia Hasha Hasła Głównego w jednym powtórzeniu.
- Dane i klucze szyfrujemy algorytmem AES-256-GCM, który wykrywa każdą modyfikację zaszyfrowanych danych. Przy każdym szyfrowaniu generowany jest nowy wektor inicjalizujący.
- Klucz Symetryczny ma długość 256 bitów.
- Kopie kluczy skarbców grup chronimy algorytmem RSA-OAEP (2048 bitów, SHA-256).
- Domyślny czas braku aktywności, po którym skarbce są blokowane, wynosi 5 minut.
- Zaktualizowaliśmy opis technologii (TLS zamiast SSL) i przepisów o ochronie danych, usunęliśmy opis ponownego generowania Klucza Symetrycznego oraz poprawiliśmy błędy redakcyjne.

24.3 v1.0 - 13 stycznia 2024

Pierwsze publiczne wydanie Raportu Bezpieczeństwa Inlockly™. Ten dokument pomoże Ci zrozumieć, w jaki sposób wdrożyliśmy wszelkie rozwiązania w Inlockly™, a także jak dbamy o Twoje dane i informacje poufne.