Analiza porównawcza mechanizmów uwierzytelniania

Bezpieczne transakcje elektroniczne i ochrona klientów

Mateusz Wirkijowski, Mieszko Makowski, Bartosz Grzybowski,
Patryk Motylski, Dawid Ryba, Mikołaj Pacek

1. Zakres analizy

Badane mechanizmy uwierzytelniania:

- Hasła (klasyczne);
- Dwuskładnikowa autoryzacja (2FA);
- Klucze sprzętowe (np. FIDO);
- Biometria;
- Single Sign-On (SSO);
- OAuth2/JWT.

Czynniki porównawcze:

- **Uniwersalność zastosowania**
  Czy dany mechanizm uwierzytelniania można zastosować w różnych miejscach? Czy mechanizm działa na komputerach, telefonach, stronach internetowych, aplikacjach mobilnych, itp.? Czy jest praktyczny w różnych sytuacjach?

- **Funkcjonalność z punktu widzenia użytkownika końcowego**
  Jak łatwo jest użytkownikowi korzystać z tego mechanizmu? Czy proces logowania jest prosty, szybki i wygodny? Użytkownicy nie lubią, gdy coś jest zbyt skomplikowane, *więc oceniamy, czy mechanizm jest przyjazny i łatwy do codziennego używania.*

- **Odporność na ataki**
  W jakim stopniu mechanizm jest odporny na ataki? Czy łatwo go "złamać" lub oszukać? Jak dobrze mechanizm chroni użytkowników przed atakami hakerów?

- **Aspekty bezpieczeństwa**
  W jakim stopniu dany mechanizm jest bezpieczny? Czy mechanizm oferuje użytkownikom wysoki poziom poufności przesyłanych danych?

- **Inne aspekty techniczne**
  Czy mechanizm wymaga specjalnego sprzętu, dodatkowego oprogramowania lub kosztownych rozwiązań? Jakie są wymagania techniczne? Czy wprowadzenie tego mechanizmu jest kosztowne lub trudne?

2. Założenia Projektowe

Główne cele merytoryczne:

- zakres funkcjonalny;
- źródła informacji;
- metody przeprowadzenia analizy;
- planowane efekty analizy.

3. Sposób realizacji

- Zebranie odpowiednich materiałów źródłowych;
- Określenie narzędzi koniecznych do wykonania;
- Określenie metodyki realizacji prac;
- Wykonanie indywidualnej analizy;

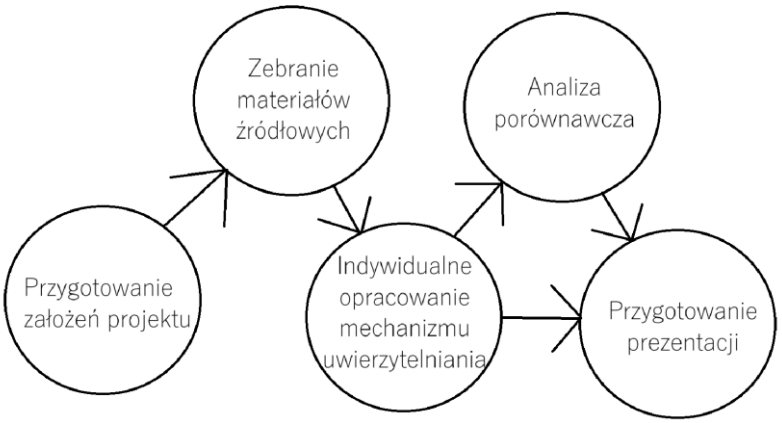
- Analiza porównawcza;
- Przygotowanie prezentacji;
- Prezentacja wykonanej analizy i oddanie projektu.

4. Opracowanie planu realizacji projektu z podziałem na zadania i „zasoby”

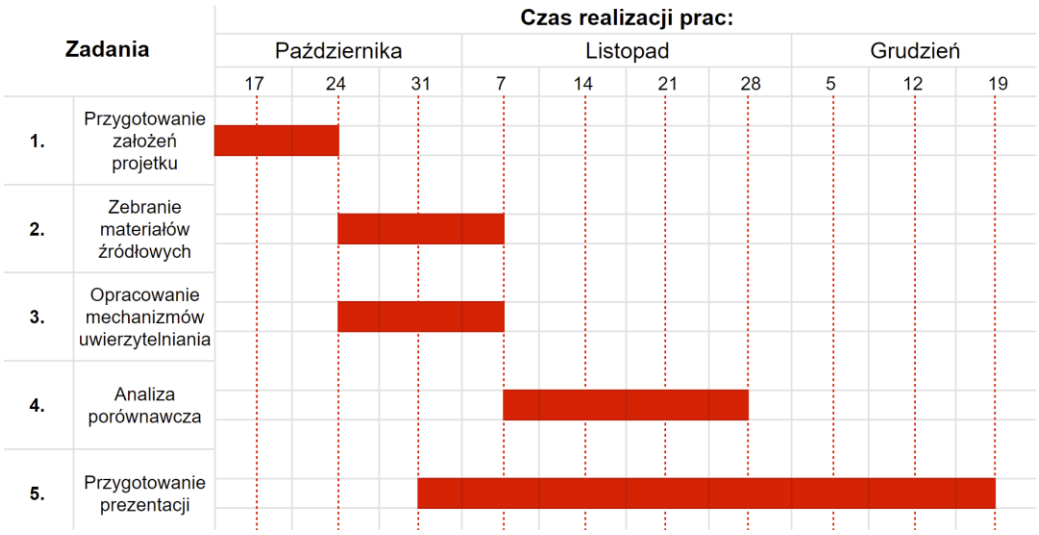
Podział zadań na etapie indywidualnej analizy danego mechanizmu:

- Hasła (klasyczne) - **Dawid Ryba**
- Dwuskładnikowa autoryzacja (2FA) - **Patryk Motylski**
- Klucze sprzętowe (np. FIDO) - **Mieszko Makowski**
- Biometria - **Bartosz Grzybowski**
- Single Sign-On (SSO) - **Mateusz Wirkijowski**
- OAuth2/JWT - **Mikołaj Pacek**

Wykres zadań i zależności między nimi (wykres Pert):



Wstępny harmonogram pracy (wykres Gantta):



5. Metody pracy i komunikacji

- Narzędzia do pracy grupowej - kanał na Discord;
- Komunikacja między członkami zespołu - kanał na Discord;
- Koordynator projektu - Mikołaj Pacek.

6. Kamienie milowe:

- **7.11.2024** – Opracowanie mechanizmów uwierzytelniania;
- **28.11.2024** – Analiza porównawcza mechanizmów uwierzytelniania;
- **19.12.2024** – Prezentacja projektu.

7. Analiza mechanizmów uwierzytelniania:

7.1. Hasła (klasyczne)

Opis:

Hasła klasyczne to najprostszy i najbardziej tradycyjny mechanizm uwierzytelniania, polegający na wpisaniu tajnego ciągu znaków, który zna jedynie użytkownik. Podstawowa idea hasła opiera się na zasadzie „coś, co wiesz” — użytkownik podaje swoje hasło, aby udowodnić swoją tożsamość. Hasła mogą składać się z liter, cyfr oraz symboli i są stosowane do zabezpieczania kont użytkowników, plików, a także zasobów cyfrowych w różnych systemach.

Opis zakresów funkcjonalnych:

- **Uniwersalność zastosowania**
  Hasła są najbardziej uniwersalnym i szeroko stosowanym mechanizmem uwierzytelniania. Można je używać na komputerach, telefonach, stronach internetowych, aplikacjach mobilnych, oraz w wielu systemach operacyjnych. Są wspierane niemal we wszystkich kontekstach cyfrowych, zarówno w aplikacjach lokalnych, jak i w chmurze.

  Ocena: Wysoka uniwersalność, ponieważ hasła są standardem od dekad i niemal każdy system obsługuje ten mechanizm.

- **Funkcjonalność z punktu widzenia użytkownika końcowego**
  Łatwość użytkowania: Tworzenie i wprowadzanie haseł jest proste, ale wymaga od użytkownika ich zapamiętania. Może to prowadzić do problemów, jeśli użytkownicy wybierają słabe, łatwe do zapamiętania hasła, które są mniej bezpieczne.

  Wygoda: Wprowadzanie haseł może być uciążliwe, zwłaszcza na urządzeniach mobilnych. Zbyt skomplikowane hasła są trudniejsze do wpisania i mogą frustrować użytkowników. Z tego powodu użytkownicy często stosują krótsze, powtarzalne hasła, co obniża bezpieczeństwo.

- **Odporność na ataki**
  Hasła są podatne na różnorodne ataki, takie jak phishing, brute force, keyloggery czy ataki typu dictionary. Jeśli użytkownicy stosują te same hasła na różnych platformach,

jeden wyciek może prowadzić do nieautoryzowanego dostępu do wielu usług. Ataki z wykorzystaniem słabych haseł, zwłaszcza tych krótkich lub popularnych (np. "123456"), są powszechne.

- **Aspekty bezpieczeństwa**
  W przypadku hasła, zabezpieczenia zależą głównie od jego złożoności i długości. Dobrze dobrane, długie hasła są trudniejsze do złamania, ale w praktyce użytkownicy często wybierają krótkie, łatwe do odgadnięcia hasła. Wiele serwisów stosuje hashowanie haseł, co podnosi poziom bezpieczeństwa, ale wycieki baz danych haseł mogą prowadzić do poważnych naruszeń bezpieczeństwa.
- **Inne aspekty techniczne**
  Koszt: Wdrożenie systemu opartego na hasłach jest bardzo tanie i nie wymaga specjalistycznego sprzętu ani oprogramowania. Niemal każdy system może obsługiwać hasła bez dodatkowych kosztów.

  Dodatkowe wymagania: Brak potrzeby specjalnego sprzętu, jednak konieczne są mechanizmy do zarządzania hasłami (np. resetowanie haseł, tworzenie odpowiednich polityk haseł).
  Ocena:

  Ocena: Bardzo niski koszt wdrożenia i brak dodatkowych wymagań sprzętowych lub technicznych.

7.2. Dwuskładnikowa autoryzacja (2FA)

Opis:

Uwierzytelnianie dwuskładnikowe (2FA, ang. Two-Factor Authentication) to mechanizm zabezpieczający dostęp do kont cyfrowych poprzez zastosowanie dwóch odrębnych warstw ochrony. Jego działanie polega na tym, że po wprowadzeniu hasła użytkownik musi dodatkowo potwierdzić swoją tożsamość za pomocą drugiego czynnika, na przykład kodu wysłanego na telefon. Takie podejście minimalizuje ryzyko nieautoryzowanego dostępu, ponieważ po przełamaniu przez atakującego jednej warstwy zabezpieczenia istnieje jeszcze jedna zupełnie odrębna, a złamanie dwóch naraz jest znacząco trudniejsze.

Możliwe formy wdrożenia:

- **Tokeny sprzętowe (Hardware tokens)**
  To fizyczne urządzenia, które generują jednorazowe kody (np. RSA SecurID). Użytkownik wpisuje kod generowany przez urządzenie podczas logowania. Tokeny sprzętowe są odporne na ataki online, ponieważ działają offline i nie są połączone z internetem. Chociaż bardzo bezpieczne, wymagają noszenia dodatkowego urządzenia, co może być niewygodne dla użytkowników.
- **Klucze bezpieczeństwa USB (USB security keys)**
  Klucze bezpieczeństwa, jak np. YubiKey lub Google Titan, działają w oparciu o protokoły takie jak FIDO U2F i są podłączane bezpośrednio do portu USB lub łączą się z urządzeniem przez NFC lub Bluetooth. Umożliwiają bezpieczną autoryzację

logowania przez jedno kliknięcie kluczu, eliminując potrzebę przepisywania kodów. Są wygodne i bardzo bezpieczne, choć wymagają fizycznego dostępu do klucza.

- **Weryfikacja SMS (SMS verification)**

  Użytkownik otrzymuje kod jednorazowy w wiadomości SMS, który następnie wpisuje podczas logowania. Ta metoda jest bardzo popularna i łatwo dostępna, ponieważ nie wymaga specjalnych aplikacji – wystarczy telefon. Warto jednak zauważyć, że jest podatna na ataki typu SIM swapping, gdzie atakujący przejmuje numer telefonu użytkownika.

- **Powiadomienia push (Push notifications)**

  Aplikacje, takie jak Duo Mobile czy Microsoft Authenticator, wysyłają na telefon użytkownika powiadomienie push z prośbą o potwierdzenie logowania. Wystarczy jedno kliknięcie, aby je zatwierdzić, co sprawia, że metoda ta jest szybka, wygodna i często bardziej bezpieczna niż SMS-y, ponieważ wymaga bezpośredniego dostępu do urządzenia mobilnego.

- **Aplikacje uwierzytelniające (Software tokens)**

  Klasyczne aplikacje do generowania kodów, takie jak Google Authenticator, Authy czy Microsoft Authenticator, tworzą jednorazowe kody (TOTP) ważne przez określony czas, które użytkownik wpisuje przy logowaniu. Aplikacje te działają offline, co zmniejsza ryzyko przechwycenia kodów przez atakujących, i są wygodną alternatywą dla SMS-ów.

- **Weryfikacja e-mail (Email-based verification)**

  Kod jednorazowy lub link potwierdzający wysyłany jest na adres e-mail użytkownika. Choć jest to mniej bezpieczne niż inne metody, e-mail może stanowić dodatkowy poziom uwierzytelnienia, jeśli jest stosowany razem z inną formą 2FA. Niestety, metoda ta jest narażona na ryzyko, jeśli konto e-mail użytkownika jest słabo zabezpieczone.

Opis zakresów funkcjonalnych:

- **Uniwersalność zastosowania**

  Dwuskładnikowe uwierzytelnianie można z łatwością zastosować w różnych środowiskach – od komputerów i telefonów po strony internetowe i aplikacje mobilne. Mechanizm ten jest elastyczny i sprawdza się w różnych sytuacjach, zapewniając dodatkowy poziom ochrony przy jednoczesnej uniwersalności zastosowania. Dzięki temu 2FA jest szeroko stosowane w bankowości, platformach społecznościowych, aplikacjach korporacyjnych oraz innych serwisach cyfrowych.

- **Funkcjonalność z punktu widzenia użytkownika końcowego**

  Dla użytkownika końcowego proces korzystania z 2FA jest zazwyczaj prosty i intuicyjny – wymaga jedynie dodatkowego kroku w logowaniu, np. wpisania kodu otrzymanego na telefon. Mechanizm ten jest zaprojektowany z myślą o wygodzie, dlatego najczęściej stosowane metody są szybkie i nieabsorbujące, co minimalizuje ryzyko frustracji użytkowników. Intuicyjność i prostota sprawiają, że 2FA jest przyjazne dla codziennego użytkowania.

- **Odporność na ataki**

  Mechanizm 2FA znacząco utrudnia atakującym uzyskanie dostępu do konta, nawet jeśli uda im się zdobyć hasło użytkownika. Dwuskładnikowa weryfikacja stanowi dodatkową barierę dla prób przejęcia konta i jest znacznie bardziej odporna na ataki typu phishing, brute-force oraz przechwytywanie danych logowania. Dzięki temu zapewnia użytkownikom wyższy poziom ochrony przed próbami oszustw i włamań. W celu zwiększenia ochrony można korzystać z MFA (multi-factor authentication), które działa tak samo jak 2FA tylko wykorzystuje więcej punktów weryfikacji użytkownika.

- **Aspekty bezpieczeństwa**

  2FA oferuje wysoki poziom poufności, gdyż dostęp do konta wymaga dwóch niezależnych składników autoryzacji. Dodatkowy element ochrony, taki jak kod jednorazowy, zwiększa poziom bezpieczeństwa danych przesyłanych w procesie logowania, co zapewnia użytkownikom lepszą ochronę przed wyciekiem wrażliwych informacji. W połączeniu z szyfrowaniem przesyłanych danych 2FA gwarantuje poufność i integralność informacji użytkownika.

- **Inne aspekty techniczne**

  Wprowadzenie 2FA nie wymaga zazwyczaj kosztownego sprzętu ani skomplikowanego oprogramowania. W wielu przypadkach wystarczą podstawowe narzędzia, takie jak aplikacja mobilna do generowania kodów lub możliwość odbierania wiadomości SMS. Implementacja 2FA jest również relatywnie prosta, a większość platform i aplikacji oferuje gotowe rozwiązania, które można łatwo dostosować do indywidualnych potrzeb, bez nadmiernych kosztów lub technicznych barier wdrożeniowych.

- **Orientacyjny koszt wdrożenia**

  Koszt wdrożenia 2FA zależy od metody. Aplikacje mobilne są darmowe, a integracja kosztuje od 500 do 5000 zł. Weryfikacja SMS wiąże się z opłatą 0,05–0,20 zł za wiadomość, co przy 10 000 logowań miesięcznie daje 500–2000 zł. Tokeny sprzętowe kosztują 150–300 zł za sztukę, a powiadomienia push generują miesięczny koszt kilkudziesięciu złotych i jednorazową integrację za 500–5000 zł. Koszty można łatwo dostosować do potrzeb firmy.

7.3. Klucze sprzętowe (np. FIDO)

Opis:

Klucze sprzętowe to zewnętrzne urządzenia najczęściej przypominające pendrive’a. Jest stosowany do potwierdzania tożsamości użytkownika podczas logowania, co dodaje dodatkowy poziom bezpieczeństwa w procesie uwierzytelniania wieloskładnikowego (MFA). Często korzysta z protokołów takich jak U2F lub FIDO2, które są szeroko wspierane przez różne usługi online.

Klucze sprzętowe można podzielić na dwa sposoby według:

A. Rodzaje hasła:

- **Jednorazowe hasła (OTP - One-Time Passwords):**

Klucz generuje jednorazowy kod, który jest ważny tylko przez krótki czas lub do jednorazowego użycia.Kod jest przesyłany na żądanie lub generowany automatycznie po podłączeniu klucza. Metoda skutecznie chroni przed przechwyceniem hasła, ponieważ nawet jeśli kod zostanie ujawniony, nie może zostać ponownie użyty.

- **Kryptografia klucza publicznego (PKI - Public Key Infrastructure):**

  Klucze sprzętowe generują parę kluczy: klucz publiczny i klucz prywatny.Klucz publiczny jest rejestrowany na serwerze uwierzytelniającym, natomiast klucz prywatny pozostaje bezpiecznie przechowywany w kluczu sprzętowym. Podczas uwierzytelniania klucz sprzętowy potwierdza tożsamość użytkownika, podpisując kryptograficznie wiadomość (challenge) serwera, co potwierdza autentyczność użytkownika.

- **PIN lub hasło dodatkowe (opcja w niektórych kluczach):**

  Użytkownik może być zobowiązany do wprowadzenia dodatkowego PIN-u lub hasła, aby aktywować klucz sprzętowy. Jest to dodatkowa warstwa zabezpieczenia, która chroni przed nieautoryzowanym dostępem w przypadku fizycznej kradzieży klucza. Hasło lub PIN musi być znane użytkownikowi i dodaje kolejną warstwę zabezpieczenia do procesu logowania.

- **Uwierzytelnianie biometryczne (dostępne w wybranych kluczach):**
- Niektóre klucze sprzętowe oferują funkcję rozpoznawania odcisku palca lub innej biometrii. Użytkownik musi potwierdzić swoją tożsamość odciskiem palca, co zwiększa bezpieczeństwo fizyczne klucza. Funkcja biometryczna eliminuje ryzyko użycia klucza przez osobę nieuprawnioną, nawet w przypadku jego kradzieży.

B. Sposoby łączenia klucza z urządzeniem:

USB-A Najczęściej stosowane w laptopach i komputerach stacjonarnych.
USB-C: Popularny w praktycznie wszystkich nowych modelach laptopów i telefonów
Lightning: Dostosowane do starych urządzeń Apple.
NFC: Klucze wyposażone w NFC umożliwiają logowanie zbliżeniowe na urządzeniach mobilnych.

Podatności:

Utrata

Najprostszą podatnością związaną z każdym urządzeniem przechowującym hasła jest kradzież lub zgubienie tego urządzenia. Ryzyko takiej sytuacji można zmniejszyć poprzez zastosowanie fizycznych środków ochrony, takich jak zamki, elektroniczne „smycze” czy czujniki obecności oraz alarmy. Skradzione tokeny mogą stać się bezużyteczne dzięki użyciu uwierzytelniania dwuskładnikowego. Zwykle, aby się uwierzytelnić, należy wprowadzić osobisty numer identyfikacyjny (PIN) wraz z danymi dostarczonymi przez token w momencie użycia tokenu.

Ataki

Każdy system umożliwiający użytkownikom uwierzytelnianie przez niezaufaną sieć (taką jak internet) jest podatny na ataki typu „man-in-the-middle” (MitM). W tego rodzaju ataku haker działa jako „pośrednik” między użytkownikiem a prawdziwym systemem, uzyskując dane uwierzytelniające od prawdziwego użytkownika, a następnie przekazując je do systemu uwierzytelniania jako własne. Ponieważ dane tokena są matematycznie poprawne, proces uwierzytelnienia przebiega pomyślnie, a oszust uzyskuje dostęp. W 2006 roku Citibank padł ofiarą tego rodzaju ataku, gdy jego użytkownicy biznesowi korzystający z tokenów sprzętowych stali się celem dużej operacji phishingowej prowadzonej przez ukraińskich przestępców.

Złamanie kodów

W 2012 roku zespół badawczy Prosecco w INRIA Paris-Rocquencourt opracował skuteczną metodę wyodrębniania klucza tajnego z kilku kryptograficznych urządzeń zgodnych ze standardem PKCS #11. Wyniki tych badań zostały udokumentowane w raporcie technicznym INRIA RR-7944, ID hal-00691958, i opublikowane na konferencji CRYPTO 2012.

Opis zakresów funkcjonalnych:

Klucze sprzętowe umożliwiają bezpieczne uwierzytelnienie i są kompatybilne z wieloma platformami i systemami, takimi jak Windows, macOS, Android, iOS oraz systemy webowe (np. Google, Microsoft, Facebook). Wspierają protokoły FIDO U2F i FIDO2, co pozwala na użycie tego samego klucza z różnymi usługami online. Niektóre modele mogą również pełnić funkcje kryptograficzne, takie jak generowanie kluczy SSH, co jest przydatne w środowiskach programistycznych.

Odporność na ataki:

Weryfikacja oparta o klucz sprzętowy uniemożliwia pomyślne podszycie się pod siebie osobom, które weszły w posiadanie Twoich danych.

- **Phishing:** Klucze sprzętowe eliminują ryzyko phishingu, ponieważ korzystają z unikalnych kodów kryptograficznych, które nie mogą zostać ponownie użyte ani przekazane.
- **MitM (Man-in-the-Middle):** Klucz komunikuje się bezpośrednio z serwerem, co eliminuje ryzyko przechwycenia poświadczeń przez osobę trzecią.
- **Zgubienie lub kradzież klucza:** Jeśli klucz zostanie zgubiony lub skradziony, osoba atakująca nadal musi znać login i hasło. Większość usług oferuje także opcję zapasowych metod dostępu, takich jak aplikacje uwierzytelniające lub zapasowy klucz sprzętowy.

Inne aspekty techniczne:

- **Wymogi regulacyjne:** Klucze sprzętowe zgodne z FIDO są często stosowane w organizacjach objętych rygorystycznymi wymaganiami regulacyjnymi (np. RODO, HIPAA).

- **Kompatybilność:** Wiele kluczy sprzętowych jest kompatybilnych z różnymi systemami operacyjnymi oraz urządzeniami, co umożliwia szerokie zastosowanie, zarówno dla użytkowników indywidualnych, jak i w organizacjach.
- **Konfiguracja i zarządzanie:** Klucze sprzętowe mogą być zarządzane centralnie w firmach, gdzie istnieje potrzeba zarządzania poświadczeniami pracowników.

7.4. Biometria

Opis:

Biometria to dziedzina nauki zajmująca się pozyskiwaniem, analizą, porównywaniem oraz obliczeniami dotyczącymi mierzalnych cech ludzkiego ciała i zachowań. W technologii informatycznej, pełni rolę mechanizmu identyfikacji i uwierzytelniania użytkowników. Umożliwia jednoznaczne rozpoznanie osoby na podstawie jej unikalnych cech. Identyfikatory biometryczne obejmują charakterystyczne cechy fizjologiczne oraz behawioralne, co znacząco rozszerza możliwości ich zastosowania w różnych systemach.

Wybrane identyfikatory biometryczne:

- **DNA** – zawiera indywidualny kod genetyczny, umożliwiający replikację komórek i syntezę białek potrzebnych do podtrzymania życia.
- **Uszy** – kształt i szczegóły budowy ucha wykazują unikalne cechy, które mogą być wykorzystywane do identyfikacji.
- **Tęczówka oka** – kolorowa część oka odpowiadająca za regulację wielkości źrenicy, zawiera niepowtarzalne wzory, umożliwiające precyzyjną identyfikację.
- **Siatkówka oka** – znajduje się w tylnej części oka i zawiera unikalny układ naczyń krwionośnych, co pozwala na jednoznaczną identyfikację konkretnej osoby.
- **Twardówka oka** – biała część oka, na której można obserwować charakterystyczną sieć naczyń krwionośnych, widoczną podczas ruchów gałek ocznych.
- **Twarz** – charakterystyki twarzy, takie jak odległość między punktami anatomicznymi, pozwalają na identyfikację. Do analizy wybierane są cechy stałe, które nie zmieniają się z wiekiem ani pod wpływem oświetlenia.
- **Geometria palców** – pomiar obejmuje cechy takie jak kształt, długość, szerokość, grubość oraz odległości między palcami.
- **Odcisk palca** – unikalne linie papilarne zawierające przypadkowe nieregularności, takie tak zakończenia i rozdwojenia, stanowią stabilną podstawę do identyfikacji.
- **Geometria dłoni** – obejmuje nie tylko geometrię palców, ale także powierzchnię dłoni i jej profil boczny.
- **Bicie serca** – niezależnie od tętna i poziomu wysiłku charakteryzuje się indywidualnym rytmem wynikającym z kształtu serca i rozmieszczenia zastawek; pozostaje stabilne przez całe życie, chyba że zmieni je choroba lub poważny epizod kardiologiczny.
- **Podpis** – odręczny podpis może być analizowany statycznie (gotowy podpis) lub dynamicznie (podczas jego wykonywania), co pozwala na identyfikację osoby.

- **Głos** – barwa i sposób mówienia użytkownika są unikalne dzięki długości strun głosowych, kształtowi gardła i jamy ustnej, jak również cechom behawioralnym, takim jak akcent czy styl mówienia.
- **Pisanie na klawiaturze** – analizowane są cechy takie jak czas potrzebny na wybranie, naciśnięcie i zwolnienie klawiszy lub ich sekwencji, typowe, powtarzające się błędy oraz rytm pisania.

Opis zakresów funkcjonalnych:

- **Uniwersalność zastosowania**

  Uwierzytelnianie biometryczne może stanowić bardzo uniwersalne rozwiązanie, w zależności od wybranych identyfikatorów biometrycznych. Mechanizmy takie jak rozpoznawanie twarzy, odcisków palców czy głosu znajdują szerokie zastosowanie w różnych środowiskach i na wielu urządzeniach. Są zintegrowane zarówno z komputerami, jak i smartfonami, a także stosowane na stronach internetowych i w aplikacjach mobilnych, co pozwala na łatwą i praktyczną identyfikację użytkownika podczas logowania do systemu. Oferują one dosyć bezproblemową i praktyczną identyfikację użytkownika, który próbuje zalogować się do systemu.

- **Funkcjonalność z punktu widzenia użytkownika końcowego**

  Biometria jest stosunkowo przyjazna dla użytkowników końcowych, ponieważ wymaga minimalnej interakcji z ich strony. Współczesne mechanizmy wdrażane do codziennego użytku są wysoce zautomatyzowane – wystarczy przyłożyć palec do czytnika linii papilarnych lub spojrzeć na ekran w celu rozpoznania twarzy, aby sfinalizować proces uwierzytelniania. Rozwiązania dostępne na rynku komercyjnym są szybkie i wygodne, eliminując konieczność zapamiętywania haseł lub generowania tokenów przez użytkownika. Wysoka popularność technologii biometrycznych wynika z ich prostoty i łatwości w codziennym używaniu.

  Niestety, logowanie biometryczne jest podatne na zakłócenia, takie jak skaleczenie palca (w przypadku skanowania linii papilarnych) lub słabe oświetlenie (w przypadku rozpoznawania twarzy), co może utrudnić lub uniemożliwić korzystanie z tej formy uwierzytelniania. Dlatego biometrię najczęściej stosuje się jako preferowaną metodę, jednak w przypadku jej niepowodzenia użytkownik ma możliwość zalogowania się za pomocą klasycznego hasła lub kodu PIN.

- **Odporność na ataki**

  Biometria zapewnia stosunkowo wysoki poziom odporności na ataki, ponieważ opiera się na unikalnych cechach danego użytkownika. Podrobienie odcisku palca czy oszukanie systemu rozpoznawania twarzy jest trudne i często wymaga zaawansowanej technologii. Niektóre systemy mogą być podatne na spoofing za pomocą zdjęć lub modeli, jednak nowoczesne rozwiązania stosują dodatkowe zabezpieczenia, takie jak detekcja żywego obrazu, co znacznie utrudnia włamanie do systemu.

  Dane biometryczne są zazwyczaj przechowywane lokalnie w formie zaszyfrowanej, co utrudnia ich przejęcie i zwiększa poziom bezpieczeństwa. Wadą jest jednak fakt, że w przypadku wycieku takich danych użytkownik nie może ich zmienić, jak w przypadku hasła, ponieważ są one bezpośrednio powiązane z jego unikalnymi cechami fizycznymi lub behawioralnymi.

- **Aspekty bezpieczeństwa**

  Poziom bezpieczeństwa uwierzytelniania oferowany przez rozwiązania biometryczne można ocenić jako wysoki. Powiązanie z unikalnymi cechami użytkownika sprawia, że rozwiązanie to zapewnia solidną ochronę. Decyzja o tym, czy użytkownik jest faktycznie tym, za kogo się podaje, opiera się na prawdopodobieństwie obliczonym przez odpowiedni algorytm oraz ustalonym progu.

  Przy ocenie bezpieczeństwa tego rozwiązania warto zwrócić uwagę na wskaźniki FAR (False Acceptance Rate) oraz FRR (False Rejection Rate). FAR odpowiada za procent przypadków, w których użytkownicy zostali błędnie zaakceptowani przez system, natomiast FRR wskazuje procent przypadków, w których użytkownicy zostali błędnie odrzuceni. Podczas projektowania algorytmu biometrycznego i ustalania progu należy uwzględnić te wskaźniki, aby osiągnąć równowagę między wygodą użytkowania (niska liczba fałszywych odrzuceń) a bezpieczeństwem systemu (niska liczba fałszywych akceptacji).

- **Inne aspekty techniczne**

  Większość dostępnych dziś na rynku urządzeń mobilnych i komputerów jest wyposażona w komponenty umożliwiające wykorzystanie rozwiązań biometrycznych. Kamery oraz skanery linii papilarnych stały się na tyle powszechne, że od dłuższego czasu stanowią standard rynkowy. Niemniej jednak chęć zastosowania bardziej zaawansowanych form uwierzytelniania biometrycznego, takich jak skanery tęczówki, może wymagać specjalistycznego sprzętu, co podnosi koszty wdrożenia takich rozwiązań. Wymogi techniczne różnią się w zależności od wybranych technologii, jednak wiele rozwiązań jest już na tyle popularnych i powszechnych, że ich zastosowanie nie wiąże się z nadmiernymi inwestycjami.

- **Producenci i ceny przykładowych rozwiązań**
  - Skanery linii papilarnych:
    - Tańsze rozwiązanie:
      Futronic Technology Co. Ltd.
      Model: FS80H USB Fingerprint Scanner
      Cena: około 150 zł
    - Drogie rozwiązanie:
      Suprema
      Model: BioLite N2
      Cena: 3000 zł+
  - Skanery siatkówki:
    - Tańsze rozwiązanie:
      IrisAccess (LG Electronics)
      Model: LG4000
      Cena: około 1000 zł
    - Droższe rozwiązanie:
      Iris ID
      Model: iCAM TD100
      Cena: około 7500 zł
  - Testy DNA:
    - Tańsze rozwiązanie:
      MyHeritage

Produkt: Home DNA Kit
Cena: 200 - 400 zł

- Droższe rozwiązanie:
  Centrum Badań DNA
  Produkt: Badanie całogenomowe
  Cena: 7500 zł+

7.5. Single Sign-On (SSO)

Opis

SSO, czyli Single Sign-On to system uwierzytelniania, który pozwala użytkownikom na dostęp do kilku aplikacji oraz usług przy użyciu jednego zestawu danych logowania np. (jednej nazwy użytkownika i hasła). Dzięki tej metodzie wystarczy, że użytkownik zaloguje się raz i uzyskuje dostęp do różnych zasobów, które są zintegrowane z system SSO, co pozwala na łatwiejszą pracę zarówno użytkownikom jak i administratorom zarządzającym dostępami.

Opis zakresów funkcjonalnych:

- **Uniwersalność zastosowania**
  SSO, jest powszechnie stosowany w wielu środowiskach korporacyjnych ze względu na łatwość użycia i przyspieszenie pracy. Jest kompatybilny z wieloma aplikacjami, usługami oraz systemami zarówno lokalnymi jak i chmurowymi.

  Single Sign-On umożliwia integrację różnych narzędzi firmowych i usług SaaS, co w połączeniu z jego szerokim wsparciem, wykorzystywany jest w wielu systemach takich jak Microsoft, Google, czy w innych systemach ERP i CRM.

  Ocena: SSO jest bardzo uniwersalnym oraz użytecznym narzędziem, co sprawia, że jest bardzo szeroko wykorzystywane przez firmy.

- **Funkcjonalność z punktu widzenia użytkownika końcowego:**
  SSO, zapewnia wygodę, pozwalając na dostęp do wielu aplikacji, eliminując potrzebę logowania się do każdej aplikacji z osobna.

  Z punktu widzenia użytkownika, wystarczy się zalogować raz, na początku pracy, by uzyskać dostęp do wszystkich potrzebnych zasobów.

  Nie tylko upraszcza to codzienną pracę, lecz również redukuje potrzebę zapamiętywania dużej ilości haseł, co zmniejsza ryzyko stosowania słabych i powtarzalnych haseł.

  Warto zaznaczyć, że SSO łączy się z innymi sposobami uwierzytelnienia (np MFA oraz 2FA), co sprawia, że jest jeszcze bardziej bezpiecznie, minimalizując tym samym wszelkie skomplikowane procesy związane z logowaniem dla użytkownika.

  Łatwość użytkowania i wygoda: Użytkowanie z SSO jest bardzo proste, wystarczy zalogować się w systemie, a wszystkie dostępy do zasobów z nim zintegrowanych zostaną przekazane użytkownikowi

- **Odporność na ataki:**
  Naturalnie nasuwającym się pytaniem odnośnie SSO, jest pytanie o ryzyko ataku, ponieważ jedno włamanie do systemu pozwoli atakującemu na przejęcie dostępów do wszystkich zintegrowanych zasobów.

  Właśnie z tego powodu SSO, jest często wsperane dodatkowym zabezpieczeniem w postaci np MFA, które znacznie podnosi poziom bezpieczeństwa.

  Warto również zaznaczyć, że systemy SSO są także podatne na ataki phishingowe oraz MiTM, lecz stosowanie silnego zabezpieczenia (co jest ułatwiane dzięki potrzebie zapamiętania jednego hasła zamiast kilku do każdego z osobnych zasobów) może ograniczyć ryzyka wynikające z tego typu ataków.
- **Aspekty bezpieczeństwa:**
  SSO wprowadza pewne korzyście w zakresie bezpieczeństwa dzięki centralnemu zarządzaniu uwierzytelnieniem. Pozwala ono na lepsze kontrolowanie dostępu do zasobów, a także szybkie reagowanie w przypadku ewentualnych naruszeń. Administratorzy również mają ułatwiony sposób monitorowania działań użytkowników. Warto jednak pamiętać, by sam system SSO, był odpowiednio zabezpieczony i chroniony przed wyciekiem danych np za pomocą MFA.
- **Inne aspekty techniczne:**
  Zdecydowanym minusem stosowanie SSO, może być koszt wdrożenia tego systemu, ponieważ jest on wyższy niż w przypadku stosowania tradycyjnych haseł. Wymaga on bardziej zaawansowanego oprogramowania i konfiguracji, by działał poprawnie. Dodatkowo systemy SSO wymagają integracji z aplikacjami, co wiąże się z kosztami utrzymania oraz specjalnych zasobó IT, by zarządzać dostępami i monitorować działania użytkowników.
  Mimo wszystko warto pamiętać, że zwiększenie wydajności pracy może jak najbardziej przyczynić się do zwrotu poniesionych kosztów w związku z wdrożeniem i utrzymaniem systemu.
- **Orientacyjny Koszt Wdrożenia**
  Jeśli chodzi o koszt wdrożenia usługi SSO, to jest bardzo duży rozstrzał między poszczególnymi dostawcami. Nie brakuje usługodawców, którzy za implementacje danego oprogramowania SSO, liczą sobie 2x, 3x a nawet 4x krotność tego co konkurencja.

  Dzięki temu, że Single Sign-on oferuje wysokie standardy bezpieczeństwa, jest proste w użyciu i zarządzaniu, a także z łatwością pozwala na integracje z istniejącymi systemami, to nic dziwnego, że niektóre przedsiębiorstwa i tak decydują się przepłacać za tę usługę.

  Oczywiście warto zaznaczyć, że koszt wacha się w zależności od zapotrzebowania, wielkości firmy, typu usługi oraz czasu przeznaczonego na implementację. Jeśli chodzi o SSO chmurowe, to najbardziej bazowe rozwiązania szacuje się pomiędzy 2-6 za użytkownika miesięcznie, podczas, gdy rozwiązania dostosowane do personalnych potrzeb danej firmy oscylują w kwotach 8-12 za użytkownika miesięcznie.

Inaczej ma się sytuacja, gdy usługę SSO, chcemy zaimplementować on-site. W takim wypadku koszty wzrastają kwot 10 000$ - 50 000$. Dokładny cennik zależy w zupełności od wymagań systemu.

7.6. OAuth2/JWT

Opis

OAuth2 to przede wszystkim protokół autoryzacji, który umożliwia aplikacjom dostęp do zasobów użytkownika w innej usłudze, bez konieczności podawania loginu i hasła do tej aplikacji.
Aplikacja kliencka gdy chce się połączyć z zewnętrzną usługą, wysyła do serwera autoryzacji (np. Google) żądanie o połączenie. Następnie użytkownik uwierzytelnia się za pomocą loginu hasła. Jeśli doszło już wcześniej do uwierzytelnienia, można wykorzystać np. **token** szybkiego uwierzytelnienia użytkownika i autoryzacji do zasobów.

JWT (JSON Web Token) jest stosowany do uwierzytelniania bądź autoryzacji w aplikacjach webowych. Jest samowystarczalny, nie wymaga osobnego serwera. Przy początkowym uwierzytelnieniu (np. za pomocą loginu i hasła), serwer aplikacji tworzy token i wysyła go do klienta. Klient przechowuje token w swojej pamięci lokalnej, gdzie bezpiecznie jest przechowywany (klient nie ma do niego dostępu np. poprzez kod). Przy następnych połączeniach z serwerem, nie jest wymagane ponowne logowanie, wystarczy, że klient przekaże w nagłówku JWT token. Dzięki temu JWT jest bezstanowe i nie wymaga sesji.

Opis zakresów funkcjonalnych

- **Uniwersalność zastosowania**

  OAuth 2.0 i JWT są niezwykle uniwersalne, stosowane w szerokim spektrum aplikacji – od webowych, przez mobilne, po API. Dzięki temu, że są obsługiwane przez wielu dostawców tożsamości, (np. Google, Facebook, Microsoft), umożliwiają łatwą integrację z zewnętrznymi usługami bez konieczności tworzenia nowego systemu logowania. OAuth 2.0 doskonale sprawdza się w przypadku aplikacji, które wymagają zdalnej autoryzacji i współpracy między wieloma systemami. Z kolei JWT, dzięki swojej lekkości i bezstanowości, jest idealnym rozwiązaniem w architekturach rozproszonych, gdzie sesje użytkowników muszą być przenoszone między różnymi komponentami systemu.

  Ocena: Wysoka uniwersalność, ponieważ implementacja obu technologii jest prosta i nie wymagają wielu zasobów.

- **Funkcjonalność z punktu widzenia użytkownika końcowego**

  OAuth 2.0 z JWT znacząco poprawia doświadczenie użytkownika, oferując logowanie jednokrotne (SSO) oraz integrację z istniejącymi kontami użytkownika u dostawców zewnętrznych,. Użytkownicy nie muszą tworzyć kolejnych kont ani zapamiętywać nowych haseł, co zwiększa wygodę i przyspiesza proces logowania - wystarczy kilka kliknięć. Proces ten, choć bardzo funkcjonalny, może być mniej intuicyjny w przypadku aplikacji o prostym charakterze, gdzie standardowy formularz

logowania może wystarczyć. Dodatkowe kroki, takie jak przekierowanie do strony autoryzacji, mogą wydłużyć proces logowania, ale rekompensuje to wzrost bezpieczeństwa.

Ocena: Średnia funkcjonalność, choć JWT jest nieodczuwalny dla użytkownika końcowego, dla mniejszych aplikacji OAuth wraz z swoimi przekierowaniami może komplikować proces logowania.

- **Odporność na ataki**

  OAuth 2.0 i JWT oferują zaawansowane mechanizmy obrony przed atakami, takimi jak Cross-Site Request Forgery (CSRF) czy man-in-the-middle (MitM), jeśli są poprawnie wdrożone. Krótkoterminowe tokeny i precyzyjne zakresy (scopes) dostępu znacząco ograniczają ryzyko nadmiernych uprawnień. Jednak te technologie są podatne na specyficzne ataki, takie jak wyciek tokenu przez XSS lub manipulację algorytmem podpisu, jeśli nie są właściwie zabezpieczone.

  Ocena: Dosyć wysoka, choć wymagają dobrej implementacji, aby zapewnić odporność.

- **Aspekty bezpieczeństwa**

  OAuth 2.0 i JWT zapewniają silne mechanizmy uwierzytelniania i autoryzacji, w tym możliwość stosowania precyzyjnych zakresów dostępu do zasobów oraz zaawansowane algorytmy podpisywania i szyfrowania. Wadą JWT jest brak natywnego mechanizmu unieważniania tokenów, co może prowadzić do sytuacji, w której nieważne tokeny pozostają aktywne do czasu wygaśnięcia. W porównaniu z tradycyjnymi sesjami, gdzie sesja może być zakończona natychmiast po wylogowaniu, OAuth 2.0 wymaga bardziej złożonych mechanizmów zarządzania tokenami, takich jak blacklisty.

- **Inne aspekty techniczne**

  OAuth 2.0 i JWT są wyjątkowo wydajne w architekturach rozproszonych, gdzie bezstanowość tokenów JWT eliminuje potrzebę utrzymywania sesji na serwerze, co ułatwia skalowanie. Implementacja tych mechanizmów wymaga jednak dodatkowej infrastruktury, w tym serwerów autoryzacyjnych oraz odpowiedniego zarządzania tokenami i kluczami kryptograficznymi. To zwiększa złożoność i koszty utrzymania, zwłaszcza w porównaniu z prostszymi systemami opartymi na ciasteczkach sesyjnych, które choć mniej skalowalne, są łatwiejsze do wdrożenia i obsługi w małych projektach.

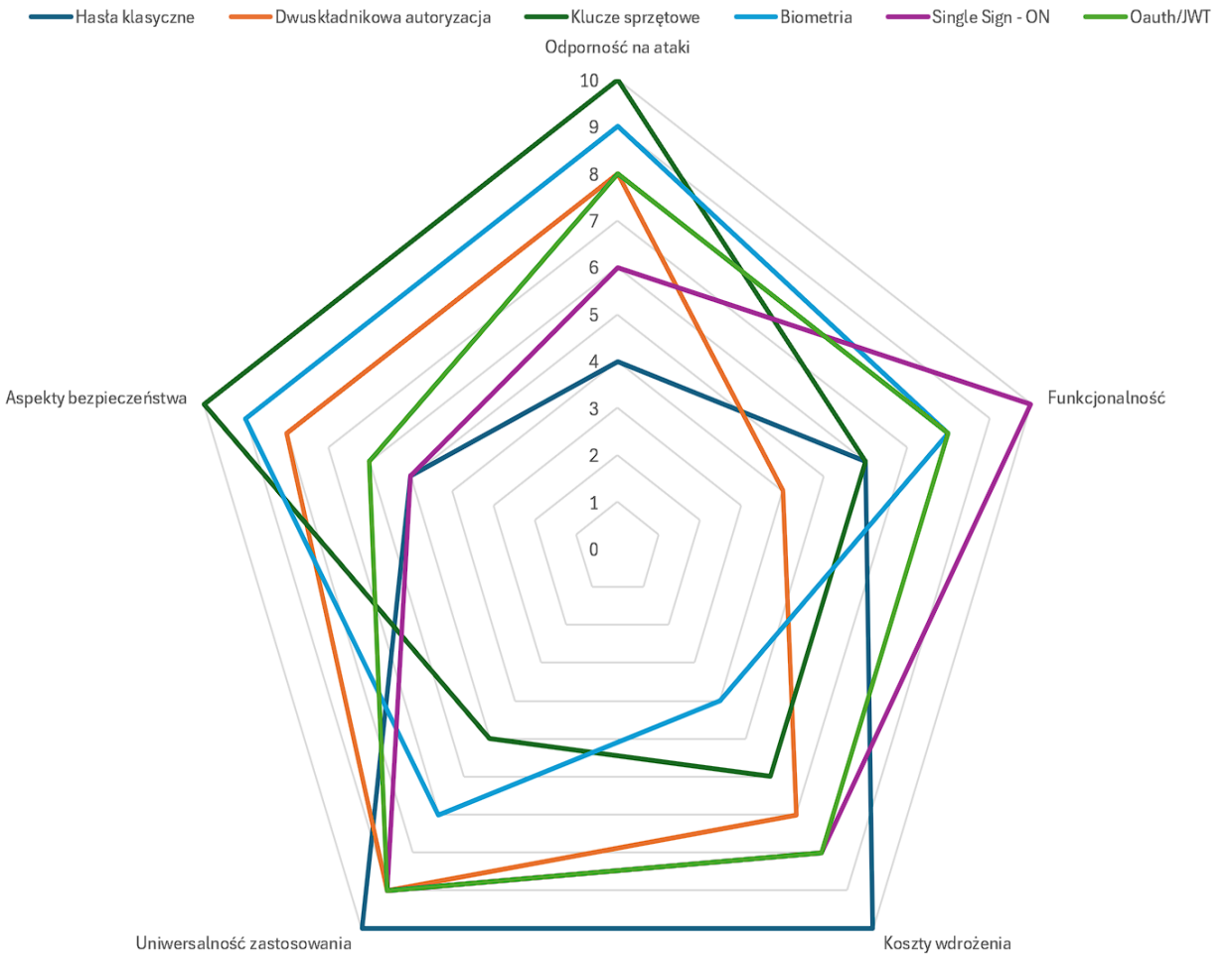
- **Orientacyjny Koszt Wdrożenia**

  Zarówno OAuth 2.0 jak i JWT są darmowymi narzędziami darmowymi, dlatego nie trzeba dokonywać ich zakupu. Koszt implementacji takiego zastosowania różni się od wielkości aplikacji, w jakiej chcemy to wykorzystać. Waha się pomiędzy 1000$ a 5000$.

8. Porównanie mechanizmów uwierzytelniania

Ocena mechanizmów

	Hasła klasyczne	Dwuskładnikowa autoryzacja	Klucze sprzętowe	Biometria	Single Sign - ON	Oauth/JWT
Odporność na ataki	4	8	10	9	6	8
Funkcjonalność	6	4	6	8	10	8
Koszty wdrożenia	10	7	6	4	8	8
Uniwersalność zastosowania	10	9	5	7	9	9
Aspekty bezpieczeństwa	5	8	10	9	5	6



Typowe miejsca zastosowania

Hasła (klasyczne):

- Logowanie do kont e-mail (np. Gmail, Outlook)
- Portale społecznościowe (np. Facebook, Twitter)
- Platformy e-commerce (np. Allegro, Amazon)
- Systemy operacyjne (np. Windows, macOS)

Dwuskładnikowa autoryzacja (2FA):

- Bankowość internetowa i mobilna
- Platformy chmurowe (np. Google Workspace, AWS)
- Logowanie do gier online (np. Steam, Xbox Live)
- Korporacyjne systemy dostępu

Klucze sprzętowe (np. FIDO):

- Systemy korporacyjne (np. VPN, CRM)
- Platformy programistyczne (np. GitHub, GitLab)
- Logowanie do usług wspierających FIDO2 (np. Google, Microsoft)

Biometria:

- Odblokowywanie urządzeń mobilnych
- Logowanie do systemów operacyjnych (np. Windows Hello)
- Płatności mobilne (np. Apple Pay, Google Pay)
- Systemy zabezpieczające dane w sektorze medycznym i bankowym

Single Sign-On (SSO):

- Platformy korporacyjne (np. Slack, Salesforce)
- Systemy edukacyjne (np. platformy uczelniane)
- Aplikacje ERP i CRM (np. SAP, Oracle)

OAuth2/JWT:

- Logowanie do aplikacji zewnętrznych („Zaloguj się przez Google”)
- Autoryzacja w aplikacjach korzystających z API
- Komunikacja między mikroserwisami
- Integracje z platformami społecznościowymi (np. Facebook API)

Podsumowanie porównania

W kwestii **odporności na ataki** na wyróżnienie zasługują **Biometria** i **Klucze sprzętowe**. Biometria wykorzystuje uniwersalne cechy ludzkie do uwierzytelniania, z kolei klucze sprzętowe dzięki swojej fizycznej formie chronią przed wykradnięciem danych przez nieautoryzowaną osobę.
W kwestii odporności na ataki odradzanie jest wykorzystanie Haseł klasycznych.

Największą **funkcjonalnością** wykazała się technologia **Single Sign - ON, Biometria** i **Oauth 2.0 wraz z JWT**. Wszystkie trzy mają ponadprzeciętną funkcjonalnością i wygodą dla użytkownika. Zdecydowanie ułatwiają uwierzytelnienie i autoryzację użytkownikowi końcowemu poprzez prostotę tych procesów.
Klucze sprzętowe wymagają fizycznego nośnika, co może stanowić problem. Dwuskładnikowa weryfikacja nie kojarzy się pozytywnie użytkownikom końcowym, zawsze wiąże się z przedłużeniem, a nie skróceniem procesu weryfikacji.

Najniższe **koszty wdrożenia** mają **Hasła klasyczne** - jest to technika często domyślnie wprowadzona w wielu urządzeniach, narzędziach, aplikacjach czy bibliotekach programistycznych. Niskie koszty ponosi się za techniki nie stosujące fizycznych urządzeń, czyli **Oauth/JWT**, **SSO** i **2FA**.
Najwyższe koszty wdrożenia ma zdecydowanie Biometria - urządzenia fizyczne stosowane do autoryzacji często potrafią przekroczyć ponad tysiąc złotych za sztukę.

Większość z przedstawionych w porównaniu technologii cechuje bardzo wysoka **uniwersalność zastosowania**.
Bardziej ograniczone zastosowanie mają Klucze sprzętowe i Biometria. Przez swoją głównie fizyczną formę, nie są w stanie obsłużyć autoryzacji i uwierzytelnienia w świecie cyfrowym.

Ponownie w kwestii **aspektów bezpieczeństwa** wygrywają **Biometria** i **Klucze sprzętowe**. Dzięki swojej fizycznej formie są one bardziej bezpieczne niż pozostałe techniki. **Dwuskładnikowa autoryzacja** również zasługuje na wyróżnienie w tej kwestii - jej istota podwójnego procesu autoryzacji zdecydowanie zwiększa bezpieczeństwo chronionych zasobów.

9. Bibliografia:

Hasła (klasyczne):

https://pl.wikipedia.org/wiki/Has%C5%82o_(kryptografia)
https://www.komputerswiat.pl/artykuly/redakcyjne/bezpieczne-hasla-wszystko-co-zawsze-chcieliscie-wiedziec/45m0dlh

Dwuskładnikowa autoryzacja (2FA):

https://www.microsoft.com/en-us/security/business/security-101/what-is-two-factor-authentication-2fa
https://www.websiterating.com/pl/blog/password-managers/what-is-2fa-mfa/
https://us.norton.com/blog/privacy/what-is-2fa
https://www.authy.com/what-is-2fa/
https://www.syteca.com/pl/blog/wieloskladnikowe-uwierzytelnienie
https://chronpesel.pl/ochrona-danych-osobowych/jak-dziala-uwierzytelnianie-dwuskladnikowe-i-dlaczego-warto-z-niego-korzystac
https://www.bankier.pl/smart/2fa-co-to-jak-ustawic-uwierzytelnianie-dwuskladnikowe

Klucze sprzętowe:

https://techlord.pl/klucze-u2f-co-to-klucz-sprzetowy-usb-dlaczego-warto-go-miec-n-43.html/
https://geex.x-kom.pl/lifestyle/klucze-sprzetowe-yubico/
https://en.wikipedia.org/wiki/Security_token
https://faktysatakie.pl/klucz-sprzetowy-co-to-jest-i-jak-dziala/
https://www.sciencedirect.com/topics/computer-science/hardware-token
https://www.upguard.com/blog/hard-tokens

Biometria:

https://en.wikipedia.org/wiki/Biometrics
https://www.biometricsinstitute.org/what-is-biometrics
https://recogtech.com/en/insights-en/far-and-frr-security-level-versus-ease-of-use

Single Sign-On (SOO)

https://blog.avatier.com/single-sign-on-sso-simplifying-user-access-and-enhancing-security/
https://www.oracle.com/security/identity-management/single-sign-on-sso/
https://frontegg.com/guides/single-sign-on-sso

https://chatgpt.com/backend-api/bing/redirect?query=SSO+authentication+model+how+it+works+security+level+implementation+costs+resistance+to+attacks
https://auth0.com/blog/what-is-and-how-does-single-sign-on-work/

OAuth2/JWT:

https://auth0.com/intro-to-iam/what-is-oauth-2
https://frontegg.com/blog/oauth-vs-jwt
https://en.wikipedia.org/wiki/JSON_Web_Token
https://www.pullrequest.com/blog/common-security-issues-in-implementing-oauth-2-0-and-how-to-mitigate-them/
https://www.secopsolution.com/blog/jwt-weaknesses