Podstawy kryptografii

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. letni 2024/2025

inf.ug.edu.pl/~amb/

Uwierzytelnianie

Metody uwierzytelniania

- Cztery metody
  – wiedza – hasło, PIN, zawołanie-odzew, protokoły wiedzy zerowej, ...
  – posiadanie przedmiotu – token, karta, klucz (fizyczny przedmiot) ..
  – cechy indywidualne i niepowtarzalne – wzorzec siatkówki oka, odcisk palca, kształt twarzy ...
  – jw. ale dotyczące zachowań – podpis odręczny, głos, sposób chodzenia, pisania ...
- Problemy
  – wiedza – można nie zauważyć kompromitacji
  – posiadanie przedmiotu – nie ma szans na backup
  – cechy indywidualne i niepowtarzalne – nie mogą być zmienione

Metody uwierzytelniania – hasło

- Hasło
  – przesyłane jawnie, tylko dla lokalnych zastosowań
  – przechowywanie hasła w jawnej postaci umożliwia kradzież przez administratora
  – serwer przechowuje skrót i porównuje ze skrótem hasła podanego
  – lepiej $\langle s, h(pass||s)\rangle$ gdzie h funkcja skrótu, s losowe ziarno
  – każde z tych rozwiązań uniemożliwia odzyskanie hasła
- Połączenie z challenge-response
  – użytkownik musi podać hasło+liczba jednorazowa

Atak na hasło

- Kradzież
  – podglądanie, kopie papierowe, wirus przechwytujący hasło
- Podstęp (phishing)
- Zgadywanie hasła
  – brutalne (wszystkie hasła)
  – słownikowe
  – ale serwer nie pozwala wykonać zbyt wielu prób
  – atak jest łatwiejszy po zdobyciu zbioru skrótów haseł
  – tablice tęczowe *blue tables*: przygotowane skróty haseł ze słownika
  – szukanie elementu w $\{h(p)|p \in \text{hasło}\} \cap \{h(w)|w \in \text{słowo}\}$
- Hasła dawne mogą być skompromitowane
- Hasła domyślne

Atak na hasło – obrona

- Hasła spoza słownika
  – testowe łamanie podczas generowania hasła przez użytkownika
- Hasła długie
- Postać syntaktyczna
  – długość
  – kilka rodzajów znaków
- Zmiana hasła
  – okresowa (czy wolno wracać do dawnych?)
  – obowiązek haseł jednorazowych

Atak na hasło – obrona, c.d.

- Solenie (salted) haseł – skróty wraz z dodatkowym ciągiem
  – $\langle s, h(pass||s)\rangle$ gdzie h funkcja skrótu, s losowe ziarno
  – uniemożliwiają użycie tablic tęczowych
  – ale nie zmieniają bezpieczeństwa ataku na pojedyncze hasło
- Dalsze utrudnienie solenia, przechowuje się $\langle s, h(pass||s||s')\rangle$ takie, że $h(pass||s||s')$ ma na początku k bitów zerowych
  – k jest ustalonym parametrem bezpieczeństwa, sensowne wartości od 10 tys. do wielu milionów
  – w momencie ustawiania hasła trzeba wykonać pewną pracę
  – przetestować ok. k wartości ziarna s' tak by uzyskać żądaną własność
  – taką samą pracę musi wykonywać atakujący testując każde z możliwych haseł
  – pomysł *proof of work*
- Łamanie haseł może przebiegać w tempie 100 mld testów na sek.
  – żądanie dowodu pracy zwalnia to tempo tysiące czy miliony

Dodatkowe zasady uwierzytelniania

- Single sign-on uwierzytelnianie w systemie następuje raz
  – zmniejsza szanse na przechwycenie, zachęca do trudniejszych haseł
  – jeden serwer uwierzytelniający w systemie
- Hasła jednorazowe
  – są przekazywane innym kanałem (np. sms)
  – lista skrótów $h_n(s), n = 0, 1, \ldots, 100$, przechowywanie wartości licznika i odpowiedniego skrótu
  – funkcja generująca hasło $G(X, p, t)$ w zależności od użytkownika X, hasła p oraz znacznika czasu t
  – zawołanie–odzew, żądający wysyła zawołanie x, uwierzytelniający się potrafi wyprodukować odzew $G(x, p)$
  – de facto jest to podpis (ślepy) na żądanie
  – często dedykowane urządzenia (token)

Klucze

Klucze – długość

- „Moc" szyfrowania powinna zależeć tylko od długości klucza
- Moc przestrzeni kluczy: najczęściej 2^n, klucze n-bitowe
- Atak brutalny – trzeba przeszukać wszystkie klucze
  – możliwość wykorzystania równoległości (specjalna architektura, sieć komputerów)
  – trzeba mieć informację, że klucz został znaleziony
  – typowy problem dla kryptografii symetrycznej
- Atak urodzinowy: szukamy powtórzenia się elementu – trzeba porównać zbiory wielkości $2^{\frac{n}{2}}$
- Kryptografia asymetryczna – problemy matematyczne o pewnej złożoności
  – łamanie RSA oraz problemy DH (logarytm dyskretny): $exp(n^{\frac{1}{3}})$
  – krzywe eliptyczne i podobne – łamanie trudniejsze niż RSA

Klucze – długość, c.d.

- Kryptografia krzywych eliptycznych – kilkukrotnie krótsze klucze niż w RSA/DH
- Nowe, nieprawdopodobne techniki:
  – DNA: „obliczenia" biochemiczne
  – komputer kwantowy – znika wykładnicza złożoność
    - przez długi czas szczytowym osiągnięciem: $15 = 3 \cdot 5$
    - dziś również $56153 = 241 \cdot 233$
    - w 2022 anonsowano rozkład liczby 48-bitowej
      – ale nie jest jasne, czy rzeczywiście jest to prawdą
    - praktyczne zastosowania ciągle są niejasne

`https://postquantum.com/post-quantum/breaking-rsa-quantum-hype/` `https://phys.org/news/2025-05-quantum-rsa-encryption-qubits`

Klucze – jakość

- Bajt $\neq$ 8 bitów
  – ascii $= 95$ znaków
  – litery (bez rozróżnienia wielkości)+cyfry: jeszcze mniej
- Atak słownikowy
  – dane użytkownika, imiona, nazwy geograficzne, łatwe kombinacje znaków na klawiaturze
  – warianty z liczbami, znakami specjalnymi
  – litera w słowie angielskim ≈ 1.3 bita
- Przeciwdziałanie:
  – dłuższe hasła, pary słów, passphrase zamiast password
  – ale niektóre implementacje mogą obcinać dłuższe hasła
  – przechowywanie skrótów haseł wraz z ziarnem i/lub wymogiem dowodu pracy
- Klucze losowe: generator liczb losowych

Długości kluczy

klucz symetryczny	klucz publiczny RSA, DH
56	384
64	512
80	768
112	1792
128	2304

- funkcje skróty powinny być dwa razy dłuższe niż klucze analogicznej siły (atak urodzinowy)
- klucze publiczne w technologii krzywych eliptycznych mogą być ok. trzy razy krótsze niż RSA i logarytm dyskretny
- dziś uznaje się za bezpieczne długości kluczy 120 bitów dla kryptografii symetrycznej i 2048 bitów dla kryptografii asymetrycznej

Problem uzgodnienia/dystrybucji kluczy

- Dystrybucja – przekazanie klucza drugiej stronie
  – obie strony mają wspólny klucz
  – ale podczas przekazywania klucz może być podsłuchany
- Uzgodnienie – po wykonaniu protokołu obie strony mają wspólny klucz
  – możliwe dopiero w kryptografii asymetrycznej (DH)
- Kryptografia symetryczna, problemy
  – konieczne przekazywanie klucza
  – odrębny klucz dla każdej pary uczestników
- Kryptografia asymetryczna, problemy
  – radykalnie mniejsza wydajność
  – problem uwierzytelniania

Przekazywanie kluczy

- Uczestnicy, często w obecności zaufanego centrum
- Wstępna dystrybucja – przed rozpoczęciem komunikacji
  – wymagany jest bezpieczny kanał
  – klucze długoterminowe, klucze do szyfrowania kluczy
  – mogą być obliczane (w kryptografii asymetrycznej)
- Przekazywanie kluczy sesyjnych – w momencie wystąpienia potrzeby
  – służą do szyfrowania wiadomości oraz do MAC
  – bezpiecznym kanałem jest wcześniej uzgodniony klucz do szyfrowania kluczy
  – funkcja szyfrowania może być b. powolna/sztucznie zwalniana
  – utrudnia to atak siłowy na klucz sesyjny
- Uzgadnianie kluczy – nie ma potrzeby wstępnej dystrybucji ani zaufanego pośrednika

Dystrybucja kluczy, c.d.

- Połączenie przekazania/uzgodnienia klucza z uwierzytelnieniem
  – sprawdzenie klucza sesyjnego: możliwość podstawienia (man-in-the-middle), błędu transmisji, błędu odszyfrowania
  – podpis pod kluczem, bezpieczny skrót
- Atak ze znanym kluczem sesyjnym
- Perfect forward secrecy: złamanie klucza długoterminowego nie pozwala na znalezienie dawnych kluczy sesyjnych
  – np. losowe klucze sesyjne

Aktualizacja kluczy

- Generowanie nowego klucza dla nowej sesji
- Zapobiega kryptoanalizie z dużą ilości kryptogramu
- Dowolna funkcja jednokierunkowa – np f. skrótu
- Ew. kompromitacja klucza nie uchroni przed kompromitacją przyszłych kluczy, ale przeszłe pozostaną bezpieczne
- Funkcja zależąca również od klucza używanego tylko do aktualizacji
- Chroni również przyszłe klucze sesyjne

Przechowywanie kluczy

- Fizyczny sprzęt, karta kryptograficzna itp.
- Zaszyfrowany plik – konieczność hasła do pliku
- Powierzanie klucza – np. centrum certyfikacji może przechowywać kopię zapasową klucza prywatego
- Podział klucza – $k = \bigoplus k_i$ wszystkie udziały k_i są losowe, klucz można odtworzyć z wszystkich udziałów ale mniejszy zestaw niczego nie ujawnia
- Schematy progowe
  - w uczestników zna udziały klucza,
  - klucz może odtworzyć $t \leqslant w$ uczestników
  - $t-1$ uczestników ma zerową wiedzę na temat klucza

Schemat progowy Shamira

- w uczestników, każdy z nich ma mieć udział w kluczu, dowolnych t uczestników może odtworzyć klucz ale $t-1$ nie może
- Przygotowanie: liczba pierwsza $p > w$, różne elementy $x_i \in \mathbb{Z}_p^*$, $i = 1, \ldots, w$ ujawnione uczestnikom
- Dla klucza K Tadeusz wybiera $t-1$ losowych elementów $a_i \in \mathbb{Z}_p$, razem z $a_0 = K$ definiują one wielomian $a(X) = \Sigma_{j=0}^{t-1}\, a_j \cdot X^j$ stopnia $t-1$
- Każdy z uczestników otrzymuje udział $y_i = a(x_i)$, $i = 1, \ldots, w$
- Odtworzenie klucza: jeśli znamy dowolnych t punktów (x_i, y_i) położnych na wykresie wielomianu stopnia $t-1$, to twierdzenie interpolacyjne Lagrange'a pozwala jednoznacznie obliczyć wielomian
  – w szczególności jego wyraz wolny $a_0 = K$
- Bezpieczeństwo: jeśli znamy tylko $t-1$ punktów, to każdy kandydat na a_0 jest dopuszczalny

Przykład uzgadniania klucza – Schemat Blooma

- Założenia: zaufany pośrednik generujący pary kluczy
  – każdy uczestnik otrzymuje klucz publiczny r_U
  – każdy uczestnik otrzymuje dwa tajne klucze $a + b \cdot r_U$, $b + c \cdot r_U$
  – liczby a, b, c są te same w całym systemie, znane tylko Tadeuszowi
- Jeśli Alicja i Bolek chcą uzgodnić klucz, to Alicja oblicza $(a + b \cdot r_A) + (b + c \cdot r_A) \cdot r_B$ i analogicznie Bolek
- Wady systemu
  – klucze są stałe, nie ma pojęcia klucza sesyjnego
  – Ewa i Mariola w porozumieniu mogą rozwiązać układ równań, znaleźć stałe a, b, c używane przez Tadeusza i znaleźć klucze tajne wszystkich uczestników
- Można podnieść liczbę nieuczciwych uczestników, ale zawsze duża koalicja może złamać system