Podstawy kryptografii

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. letni 2023/2024

inf.ug.edu.pl/~amb/

Schematy podpisu cyfrowego

Podpis cyfrowy, podstawy

- Cel: (1) przekonać odbiorcę o autentyczności dokumentu
  – dodatkowo, (2) przekonać też stronę trzecią
- W kryptografii symetrycznej (1) jest łatwe: Alicja i Bolek mają wspólny i tajny klucz, Bolek wie, że dokument zaszyfrowany musi pochodzić od Alicji
  – ale i Alicja i Bolek mogą sporządzić ten sam dokument, sąd nie ma podstaw wierzyć, że autorem nie jest Bolek
  (2) nie jest spełnione
- Cechy (pożądane): (1) niemożność sfałszowania podpisu
  – (2) niemożność przeniesienia podpisu
  – (3) niemożność zmiany podpisanego dokumentu
  – dodatkowo: łatwość identyfikacji osoby składającej podpis, łatwość weryfikacji podpisu

Podpis cyfrowy, kryptografia symetryczna

- Protokół z kluczem symetrycznym:
  – zaufany arbiter Tadeusz zna tajne klucze wszystkich uczestników
  – Alicja przesyła zaszyfrowany dokument do arbitra z informacją, że adresatem jest Bolek: $\langle K_A(M), B\rangle$
  – ten odszyfrowuje, szyfruje i przesyła do Bolka z informacją o autorze: $K_B(\langle M, A\rangle)$
  – albo przesyła $K_B(\langle M, A\rangle)$ do Alicji, by to ona przesłała
  – Bolek wierzy Tadeuszowi, że wiadomość jest od Alicji, w razie potrzeby powoła się na Tadeusza
  – problemy: potrzebny jest zaufany pośrednik,
- zna on wszystkie klucze (niebezpieczeństwo kompromitacji),
- podpis jest przeznaczony tylko dla jednego odbiorcy
- duże wymagania obliczeniowe
  - ale można operować wyłącznie na skrótach

Podpis cyfrowy a szyfrowanie w kryptografii asymetrycznej

- Dane są dwie funkcje: $F(k, \ldots)$ oraz $G(\ell, \ldots)$ wzajemnie odwrotne: dla wszystkich k, ℓ, m zachodzi $G(\ell, F(k, m)) = m$
  – szyfrowanie: $F(k, \ldots)$ szyfruje kluczem publicznym k, $G(\ell, \ldots)$ odszyfrowuje kluczem prywatnym ℓ
  – podpis: $F(k, \ldots)$ podpisuje kluczem prywatnym k, $G(\ell, \ldots)$ weryfikuje kluczem publicznym ℓ, $G(\ell, F(k, m)) == m$
  – albo podpis $G(\ell, \ldots)$ i weryfikacja za pomocą $F(k, \ldots)$
- Czy to zawsze możliwe?:
  – klucze niekoniecznie można zamieniać rolami
  – operacje z kluczem publicznym i prywatnym nie muszą być przemienne

Podpis cyfrowy w kryptografii asymetrycznej

- Para kluczy, prywatny i publiczny (s, p)
  – podpisywanie kluczem prywatnym może być niedeterministyczne, $Sig(s, m)$
  – weryfikacja kluczem publicznym musi dawać wynik T/F, $V(p, Sig(s, m)) = T$, $V(p, m_1) = F$ jeśli m_1 nie jest postaci $Sig(s, m)$ dla pewnego m.
- Bezpieczeństwo (atak egzystencjalny):
  – Mariola zna klucz publiczny
  – i ma dostęp do urządzenia podpisującego
  – wygrywa, jeśli potrafi przedstawić jakąkolwiek podpisaną wiadomość (wcześniej nie podpisaną przez urządzenie)
  – nie wiemy, czy ta wiadomość jest jej przydatna

Podpis cyfrowy a MAC

MAC	**Podpis cyfrowy:**
odbiorca musi mieć wspólny klucz z nadawcą	każdy może zweryfikować podpis
dla każdego odbiorcy musi być odrębny MAC	dokument jest podpisany raz dla wszystkich
odbiorca sam ma pewność, ale nie może jej przekazać	weryfikacja jest dostępna wszystkim
MAC nie wiąże się ze zobowiązaniem	podpisu nie można się wyprzeć (jeśli z góry ustalono związek z kluczem publicznym)

Algorytm RSA

- Przygotowanie Alicji:
  – wybiera duże liczby pierwsze p i q, oblicza $N = p \cdot q$
  – wybiera e i d takie, że $e \cdot d = 1 \mod \varphi(n)$, $\varphi(n) = (p-1) \cdot (q-1)$
  – klucz publiczny: (N, e)
  – klucz prywatny: (N, d)
- Podpis Alicji:
  – przesyła do Bolka $S((N, d), m) = m^d \mod N$, m musi spełniać $m < N$, $NWD(m, N) = 1$
- Weryfikacja przez Bolka:
  – pobiera klucz publiczny Alicji
  – oblicza $V((N, e), s) = s^e = m^{d \cdot e} = m^{(1+\lambda \cdot \varphi(n))} = m \mod N$,
  – znajduje m i przekonuje się, że tylko właścicielka klucza prywatnego była w stanie tak zaszyfrować
- Każdy może przeprowadzić same kroki co Bolek

Słabości RSA

- Nieodporność na fałszerstwo egzystencjalne
  – Mariola może wybrać dowolne y_1, obliczyć $m_1 = y_1^e \mod N$ i twierdzić, że Alicja podpisała wiadomość m_1
  – co prawda wiadomość m_1 będzie prawie na pewno bezsensowna i bezużyteczna
- Nieodporność na fałszerstwo z wybranym tekstem
  – Mariola zdobywa dwa podpisy na wiadomościach m_1 oraz $m_2 = \frac{m}{m_1}$ i łatwo oblicza podpis m: $m^d = m_1^d \cdot m_2^d$
  – podpis pod nieznanymi dokumentami musi być stosowany ostrożnie
  – ale jest stosowany w różnych protokołach uwierzytelniania

RSA, podpis ślepy

- Bolek chce podpisu Alicji pod dokumentem m jej nieznanym
  – np. patent w biurze patentowym
- Algorytm Chauma
  – Alicja przygotowuje parę kluczy, $\langle N, e\rangle$, $\langle N, d\rangle$
  – Bolek wybiera losowe k i przesyła do podpisu $m \cdot k^e \mod N$
  – Alicja podpisuje $y = (m \cdot k^e)^d = m^d \cdot k \mod N$
  – Bolek oblicza $y/k = m^d \mod N$ - otrzymuje podpisany dokument m
- Alicja nie wie co podpisała
  – wniosek: taki podpis może być składany jedynie za pomocą pary kluczy przeznaczonej do składania podpisu ślepego

Podpis skrótu jako zasada ogólna

- Kryptografia asymetryczna jest mało wydajna
  – dokumenty są znacząco dłuższe niż tysiące bitów
  – rozwiązanie: podpisywanie jedynie skrótu dokumentu
  – sam dokument nie da się odtworzyć z podpisanego skrótu, musi być dołączany do przesyłki
- dokument m, klucz prywatny s, podpis $\langle m, S(s, h(m))\rangle$
- Tw.: jeśli schemat podpisu jest bezpieczny oraz skrót jest bezkolizyjny, to schemat podpisu skrótu jest bezpieczny
  – nie da się utworzyć podpisanego skrótu
  – nie da się przenieść podpisu z innego dokumentu
- Uwaga: bezpieczeństwo bardzo mocno zależy od (silnej) bezkolizyjności
  – dwa dokumenty o tym samym skrócie mają ten sam podpis

Funkcje skrótu - atak urodzinowy

- Prawdopodobieństwo, że dwie osoby spośród n osób mają urodziny tego samego dnia
  – dla $n \geqslant 22$ jest $> \frac{1}{2}$
- Prawdopodobieństwo, że dwie spośród r losowych liczb z zakresu $0 \ldots n$ są równe jest $1 - e^{\frac{-r^2}{n}}$
  – jeśli zakres jest 50 bitowy, to wystarczy wygenerować 2^{30} liczb by praktycznie na pewno było powtórzenie
- Alicja przygotowuje dwie wersje dokumentu, w każdej dokonuje 2^{30} małych modyfikacji, znajduje dwie wersje o identycznej funkcji skrótu i przekonuje Bolka do złożenia podpisu pod jedną wersją
  – de facto fałszuje podpis Bolka
- Funkcje skrótu powinny być dwa razy dłuższe niż się wydaje
- Bolek może dokonać małej modyfikacji dokumentu przed jego podpisaniem

Podpis skrótu w RSA

- Podpisany dokument: $\langle m, H(m)^d \mod N \rangle$
  – weryfikacja: czy $H(m) = s^e \mod N$?
- Fałszerstwo egzystencjalne
  – po obliczeniu $s^e \mod N$ dla dowolnego s trzeba znaleźć m o danym skrócie $H(m)$ - zadanie praktycznie niewykonalne
- Fałszerstwo z dwoma podpisami
  – nie jest łatwo znaleźć dwie wiadomości o znanym z góry iloczynie skrótów
- Nie ma jednak dowodu, że podpis RSA nawet z bezkolizyjnym skrótem jest bezpieczny

Schemat ElGamala

- Przygotowanie Alicji:
  – wybiera liczbę pierwszą p, generator g, wykładnik $a < p - 1$
  – klucz publiczny: $(p, g, \alpha = g^a \mod p)$
  – klucz prywatny Alicji: a
- Alicja podpisuje wiadomość $m < p$:
  – losuje k, t.ż. $NWD(k, p-1) = 1$,
  – oblicza $r = g^k \mod p$ oraz $s = k^{-1} \cdot (m - a \cdot r) \mod p - 1$
  – podpisem jest cała trójka $\langle m, r, s \rangle$
- Bolek weryfikuje podpis na podstawie klucza publicznego:
  – sprawdza równość $\alpha^r \cdot r^s = g^m \mod p$
- Uzasadnienie: $s \cdot k + a \cdot r = m \mod p - 1$
  – a więc $g^m = g^{s \cdot k + a \cdot r} = r^s \cdot \alpha^r \mod p$
- Niedeterminizm:
  – ta sama wiadomość może mieć wiele różnych podpisów

Schemat ElGamala, bezpieczeństwo

- Mariola chce sfałszować podpis pod inną wiadomością m
  – k jest dowolne, więc r też, α jest znane, szuka s takiego, że $\alpha^r \cdot r^s = g^m \mod p$
  – czyli rozwiązuje problem logarytmu dyskretnego $r^s = \alpha^{-r} \cdot g^m \mod p$
  – może zacząć od wyboru s i szukać r, ale to też sprowadzi się do problemu logarytmu dyskretnego
- Nie wiadomo, czy wspólne szukanie r i s ułatwi zadanie

Schemat ElGamala, losowość

- Alicja podpisała dwie wiadomości m_1 i m_2 z tą samą wartością losową k
  – a więc część r w obu podpisach jest identyczna
  – Ewa widzi to i wie, że $s_1 \cdot k - m_1 = s_2 \cdot k - m_2 \mod p - 1$
  – czyli $(s_1 - s_2) \cdot k = m_1 - m_2 \mod p - 1$
  – k daje się obliczyć, być może niejednoznacznie
  – z równania $a \cdot r = m_1 - k \cdot s_1 \mod p - 1$ można obliczyć a, również być może jest kilka rozwiązań
- Tzn. system jest całkowicie skompromitowany, znany jest klucz prywatny i można fałszować wszystkie podpisy

Digital Signature Algorithm

- Standard opracowany w 1991 r. przyjęty w 1994 r.
- Opiera się na problemie logarytmu dyskretnego
  – a więc nie da się bezpośrednio użyć do szyfrowania
  – wymaga przekazania oryginału wiadomości (jak ElGamal)
- Algorytm jest bardziej skomplikowany
  – lecz szybszy w działaniu (dwa potęgowania zamiast trzech przy weryfikacji)
  – bezpieczniejszy, wymaga by $p - 1$ miało duży dzielnik pierwszy
- częścią standardu jest funkcja skrótu SHA-1 specjalnie zaprojektowana z tej okazji
  – zastąpiona najpierw przez SHA-2, a obecnie przez SHA-3
- Formalnie, nie ma żadnego dowodu bezpieczeństwa schematu