Ministerstvo financií
Slovenskej republiky

EFQM
Recognised for excellence
5 star - 2012

Digitálne podpisy

Michal Rjaško

KPMG
cutting through complexity™

Digitálne podpisy

Michal Rjaško

Obsah

- Úvod, čo sú digitálne podpisy
- Bezpečnosť podpisových schém
- Najpoužívanejšie podpisové schémy
  - RSA, RSA-FDH, RSA-PKCS #1 v1.5, RSA-PSS
  - El-Gamal
  - (EC)DSA,
- Použitie a implementačné problémy

Úvod

- „elektronické podpisy“ v legislatíve
- „digitálne podpisy“ v kryptológii
  - „digitálne podpisy“ $\subset$ „elektronické podpisy“

Iba nejaký elektronický „tag“
skoro žiadne garancie – rozhoduje súd,

Dosiahnuteľné len s pomocou kryptológie, poskytujú určité „matematické“ garancie

Digitálne vs. vlastnoručné podpisy

- Digitálny podpis – ekvivalent vlastnoručného podpisu?
- Vlastnoručný podpis:
  - Identifikácia podpisujúceho, nepopierateľnosť autorstva
  - Nefalšovateľnosť
  - Potvrdenie dokumentu podpisujúcim (informovaný súhlas)
- V digitálnom svete je
  - jednoduché kopírovanie
  - jednoducho pozmeníme dokument

    ⇒ Digitálny podpis musí závisieť na dokumente

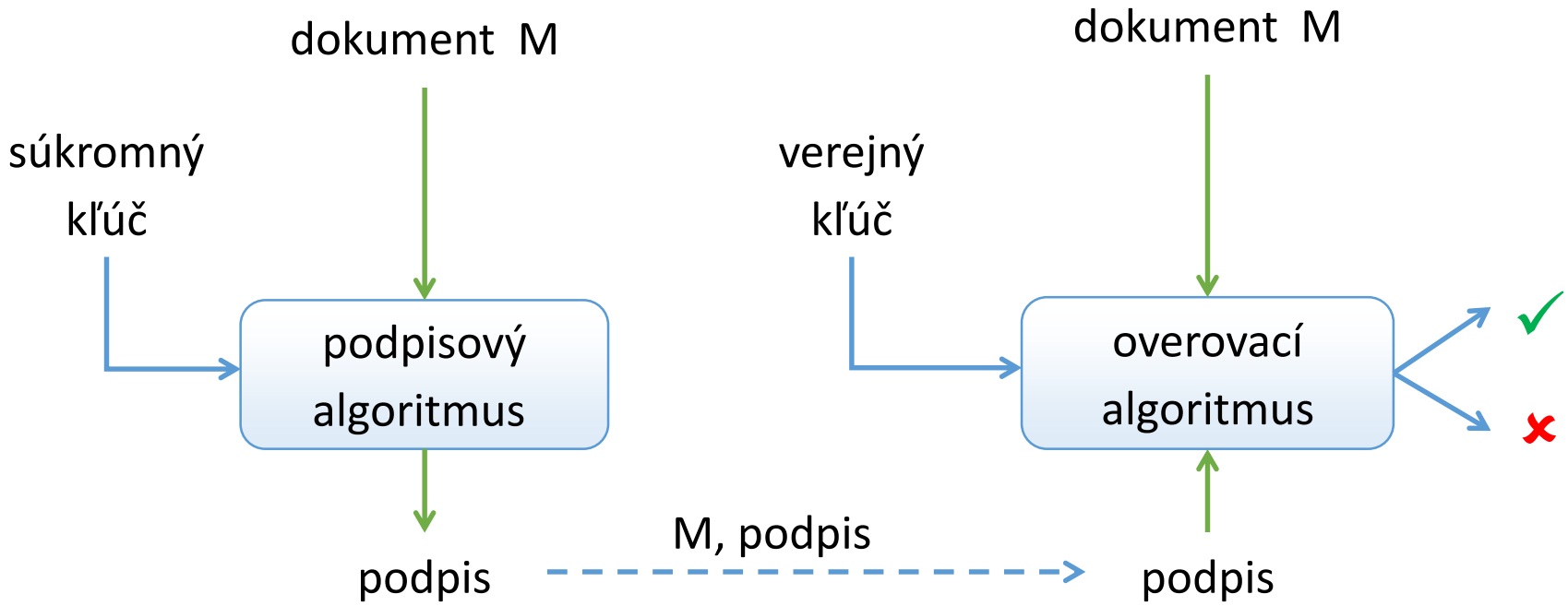
Digitálne vs. vlastnoručné podpisy

- Digitálne podpisy:
  - Identifikácia podpisujúceho, nepopierateľnosť autorstva
  - Nefalšovateľnosť
  - Potvrdenie dokumentu podpisujúcim (informovaný súhlas) ?
  - Autentickosť a integrita podpísaných údajov
  - Každý môže overiť podpis (zvyčajne)
- Niektoré vlastnosti nie je možné dosiahnuť len pomocou podpisových schém
  - PKI, zákony, predpisy (nie sú predmetom tejto prednášky)

Digitálne podpisy

- Identifikácia podpisujúceho
  - Crypto + PKI + bezpečný hardware
- Nefalšovateľnosť
  - Crypto
- Potvrdenie dokumentu podpisujúcim, nepopierateľnosť autorstva
  - Ťažké dosiahnuť v praxi
  - Crypto + zákon + predpisy + dôveryhodný harware a software
- Autentickosť a integrita podpísaných údajov
  - Crypto
- Každý môže overiť podpis (zvyčajne)
  - Crypto

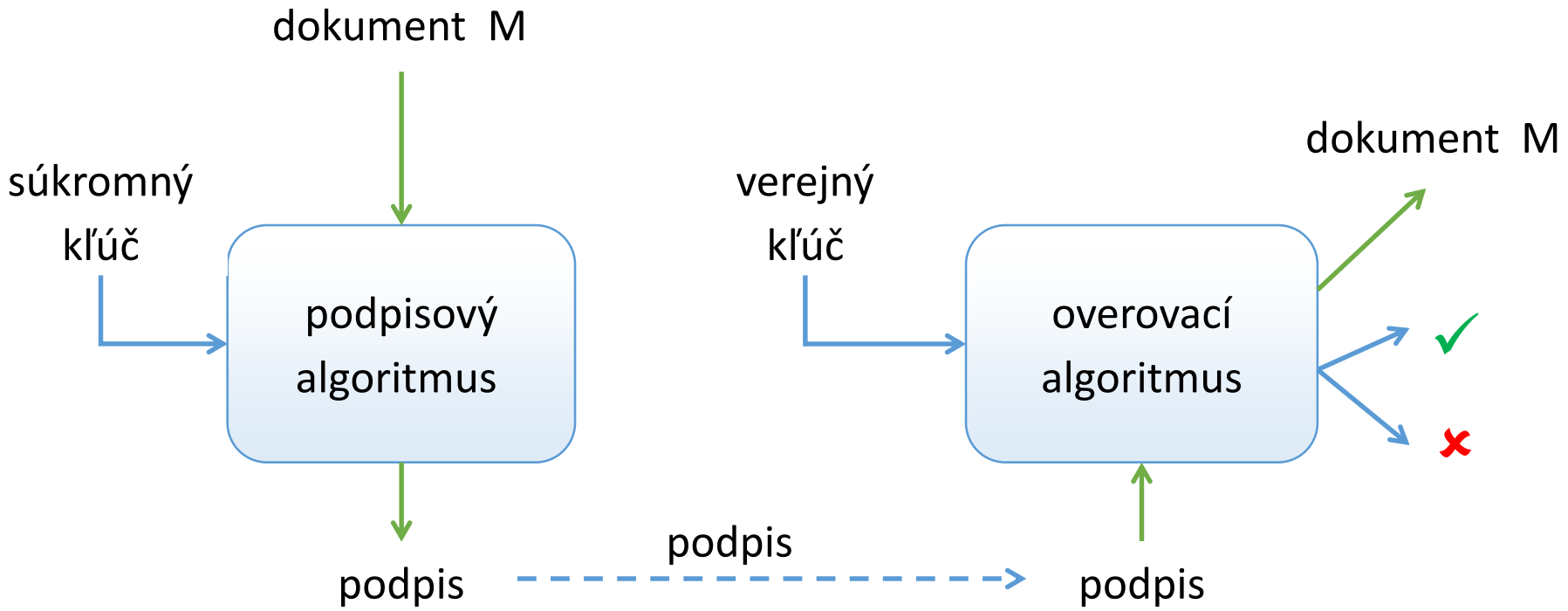
Schémy digitálnych podpisov
s apendixom



Asymetrická konštrukcia

- Súkromný kľúč – podpisovanie
- Verejný kľúč - overovanie

Schémy digitálnych podpisov
s rekonštrukciou správy

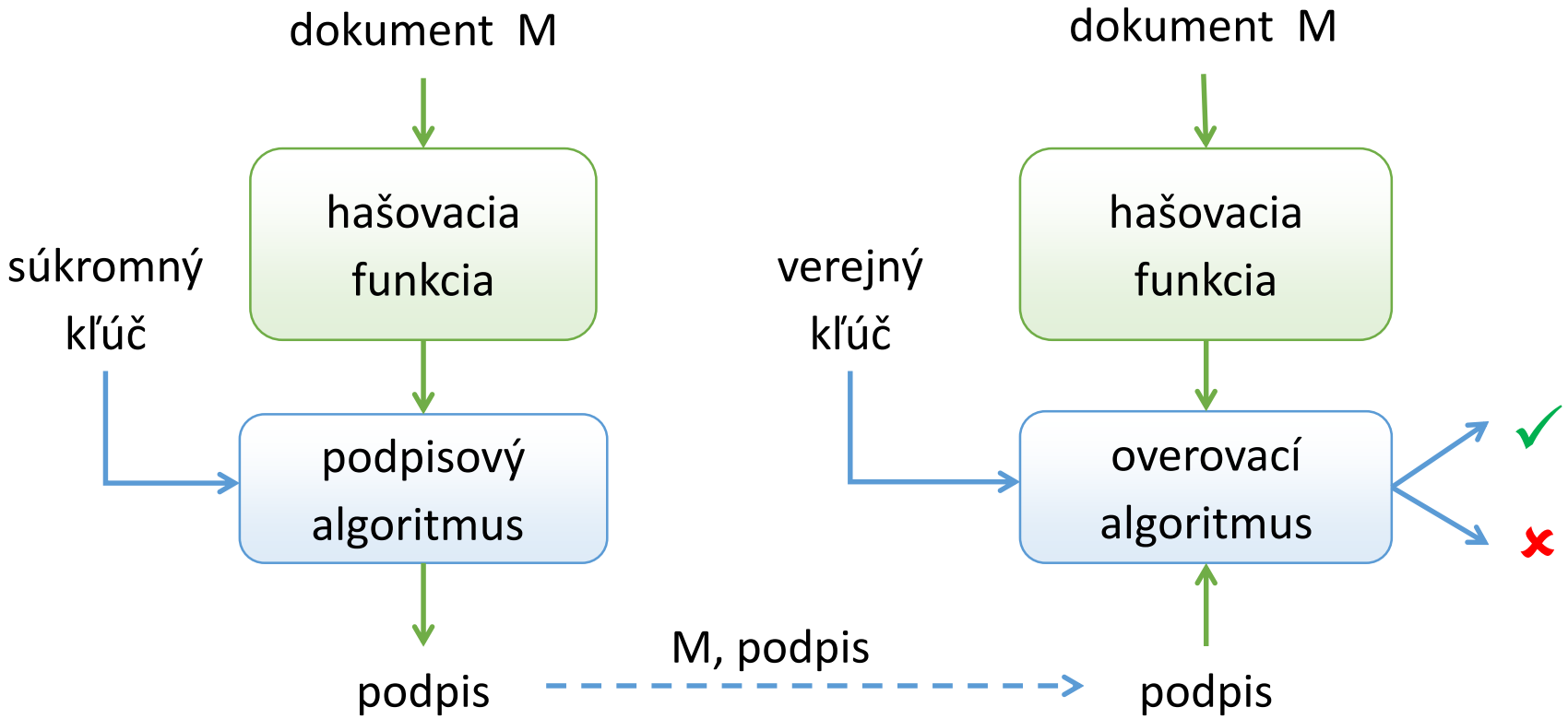


Asymetrická konštrukcia

- Súkromný kľúč – podpisovanie
- Verejný kľúč - overovanie

Schémy digitálnych podpisov

najčastejšia forma

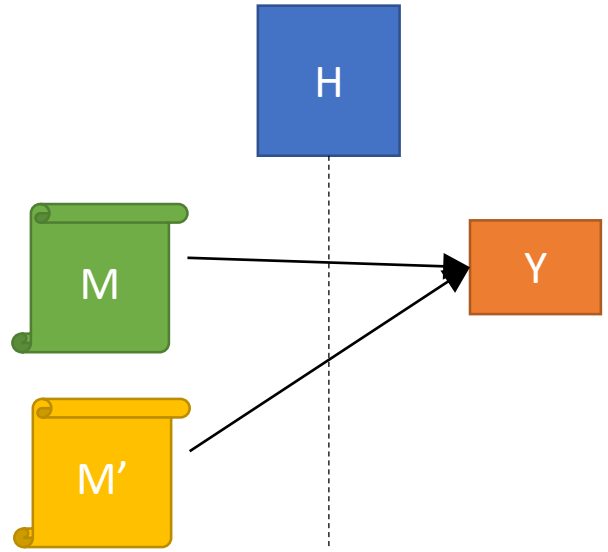


Asymetrická konštrukcia

- Súkromný kľúč – podpisovanie
- Verejný kľúč - overovanie

Schémy digitálnych podpisov – poznámky

- V prednáške sa budeme zaoberať len schémami s apendixom
- Použitie hašovacej funkcie
  - Rýchlejšie podpisovanie – podpisuje sa len krátky odtlačok
  - Zabraňuje určitým druhom útokov, napr. falšovanie náhodnej správy
- Bezpečnosť schémy závisí aj od vlastností použitej h.f.
  - napr. potrebujeme odolnosť voči kolíziám



Schémy digitálny podpisov

Podpisová schéma: (Gen, Sig, Vrf)

Gen efektívny (pravdepodobnostný polynomiálny) algoritmus, generuje verejný a súkromný kľúč (pk, sk)

Sig efektívny algoritmus, ktorý na základe správy a tajného kľúča podpisujúceho vytvorí podpis: $\sigma = Sig_{sk}(\mathrm{m})$

Vrf zvyčajne deterministický polynomiálny algoritmus, ktorý overuje podpis danej správy: $Vrf_{pk}(m, \sigma) \in \{\mathrm{true}, \mathrm{false}\}$

Korektnosť podpisovej schémy:

$$\forall (pk, sk) \leftarrow Gen\left(1^k\right) : \forall m : Vrf_{pk}\left(m, Sig_{sk}(m)\right) = true$$

Bezpečnosť podpisových schém

- Idea podobná bezpečnosti autentizačných kódov
- Útočník má prístup k verejnému kľúču

1. Cieľ – čo chce útočník dosiahnuť
   - čo najjednoduchší
2. Možný scenár útoku – ako prebieha útok
   - čo najsilnejší
3. Zdroje – výpočtová sila útočníka
   - čo najväčšie, ale dosiahnuteľné

Bezpečnosť podpisových schém

1. Cieľ
   a) Odhaliť súkromný kľúč
   b) Možnosť vytvoriť falošný podpis pre akúkoľvek správu
   c) Možnosť vytvoriť falošný podpis danej správy m
   d) Možnosť vytvoriť falošný podpis útočníkom zvolenej správy m
      - aj keď m nedáva zmysel
      - “Existential forgery”

Bezpečnosť podpisových schém

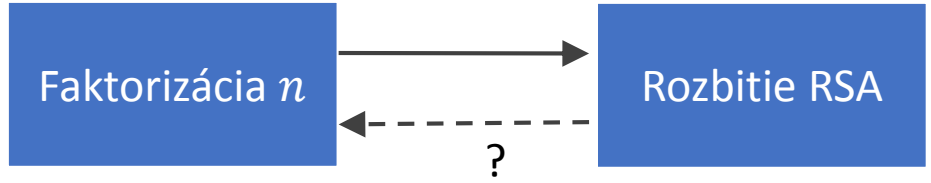
2. Scenár útoku
   a) Len so znalosťou verejného kľúča
   b) Útočník pozná niekoľko dvojíc (m, σ), kde σ je podpis správy m
   c) Útočník si môže zvoliť niekoľko správ, ku ktorým následne dostane ich podpis
   d) Adaptívny útok, kde má útočník prístup k podpisovaciemu orákulu $Sig_{sk}(\cdot)$ – môže si dať podpísať akúkoľvek správu
      - „Adaptive Chosen Message Attack“ (CMA)
3. Zdroje – cca 2^{80} operácií v rozumnom čase

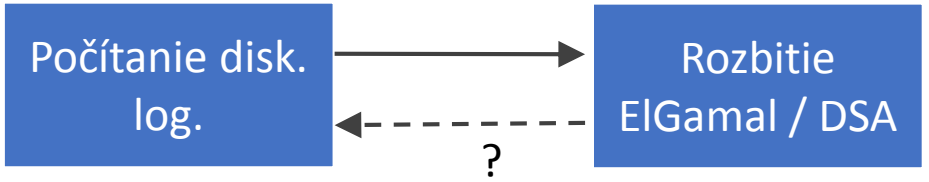
Bezpečnosť podpisových schém

- EUF-CMA (existentially unforgeable under the chosen message attack)
  - CMA – útočník má prístup k orákulu $Sig_{sk}(\cdot)$
  - EUF – úlohou útočníka je vytvoriť správu m (na ktorú sa nepýtal orákula) a jej podpis σ, kde $Vrf_{pk}(m, \sigma) = true$
- Podpisová schéma je EUF-CMA, ak pravdepodobnosť úspechu akéhokoľvek efektívneho útočníka je zanedbateľná
  - Za predpokladu, že nejaký matematický problém je ťažký (faktorizácia, diskrétny logaritmus, DH-problém)
- Efektívny útočník:
  - $< 2^{80}$ operácií, alebo
  - pravdepodobnostný polynomiálny algoritmus

V praxi používané podpisové schémy

- RSA
  - Postavené na RSA probléme, ktorý súvisí s problémom faktorizácie
- ElGamal
  - postavená na probléme diskrétneho logaritmu
- DSA
  - štandardizovaný NISTom v roku 1994
  - postavený na probléme diskrétneho logaritmu
  - obmena ElGamalovej schémy





RSA podpisová schema

1. pokus – „učebnicová“ verzia

- Veľmi podobná šifrovacej schéme RSA – „obrátené“ transformácie

Gen

- $n = p \cdot q$, kde p, q sú veľké prvočísla
- $e \cdot d \equiv 1 \pmod{(p-1)(q-1)}$
- Verejný kľúč: $pk = (n, e)$
- Súkromný kľúč: $sk = d$

Nepoužívať!

Sig: $\sigma = m^d \bmod n$

Vrf: $\sigma^e \bmod n == m$?

- Korektnosť vyplýva z vlastností (bijektívnosti) RSA

RSA podpisová schema

1. pokus – „učebnicová“ verzia – **Problémy**

- Vieme podpisovať iba krátke správy ($m \in \mathbb{Z}_n$)
- Falšovanie náhodnej správy:
  - Útočník vie vygenerovať dvojicu (správa, podpis) – Existential forgery
  - $(\underbrace{\sigma^e \ mod \ n}_{m}, \sigma)$, kde $\sigma \overset{\$}{\leftarrow} \mathbb{Z}_n$
  - Útočník nemá kontrolu nad správou m
- Z dvojíc $(m_1, \sigma_1), (m_2, \sigma_2)$ vie útočník vytvoriť validný podpis $(m_1 m_2 \bmod n, \sigma_1 \sigma_2 \bmod n)$

RSA podpisová schema

2. pokus – „hašovaná“ verzia

Gen: rovnaké ako v predošlom prípade

Sig: $\sigma = H(m)^d \bmod n$

Vrf: $\sigma^e \bmod n \;== H(m)$?

✓Môžeme podpisovať ľubovoľne dlhé správy

✓Odolnosť H voči nájdeniu vzoru ⇒ odolnosť schémy voči falšovaniu náhodnej správy

- H musí byť odolná voči kolíziám
- RSA-FDH (Full Domain Hash) schéma využívajúca h.f. H, ktorej obraz ∈ $\mathbb{Z}_n$
  - Dokázateľná bezpečnosť v modeli s náhodným orákulom
- Obraz H je zvyčajne kratší ako n ⇒ padding

PKCS #1 v 1.5 (EMSA-PKCS1-v1_5)

- Konštrukcia štandardizovaná v roku 1998
  - Bez formálneho dôkazu bezpečnosti
- Často používaná v praxi, napr. v X.509 certifikátoch:
  - „sha1RSA“ alebo „PKCS #1 sha1 with RSA encryption“
- Padding pre odtlačok H(M):

$$0\text{x}00 \parallel 0\text{x}01 \parallel 0\text{xff} \parallel \cdots \parallel 0\text{xff} \parallel 0\text{x}00 \parallel H(m)$$

“Moreover, while no attack is known against the EMSA-PKCS-v1_5 encoding method, a gradual transition to EMSA-PSS is recommended as a precaution against future developments.” (RFC 3447)

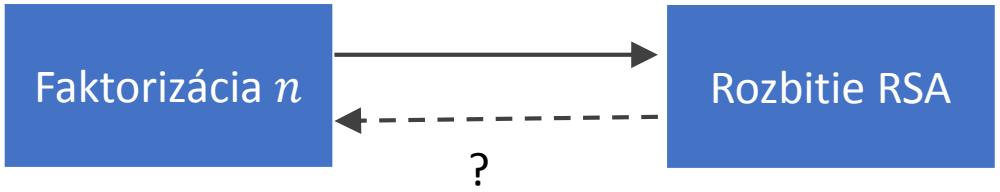
RSA-PSS

- Probabilistic Signature Scheme – štandardizovaná v roku 2002 PKCS #1 v 2.1, RFC 3447
- Dokázateľne bezpečná v modeli s náhodným orákulom

Úspešný EUF-CMA útočník
$\Rightarrow$ vieme riešiť RSA problém (počítať $\sqrt[e]{\cdot} \mod n$)

! Za predpokladu, že H je modelované ako náhodné orákulum

- dôkaz nehovorí o bezpečnosti schémy ak je použitá štandardná h.f.

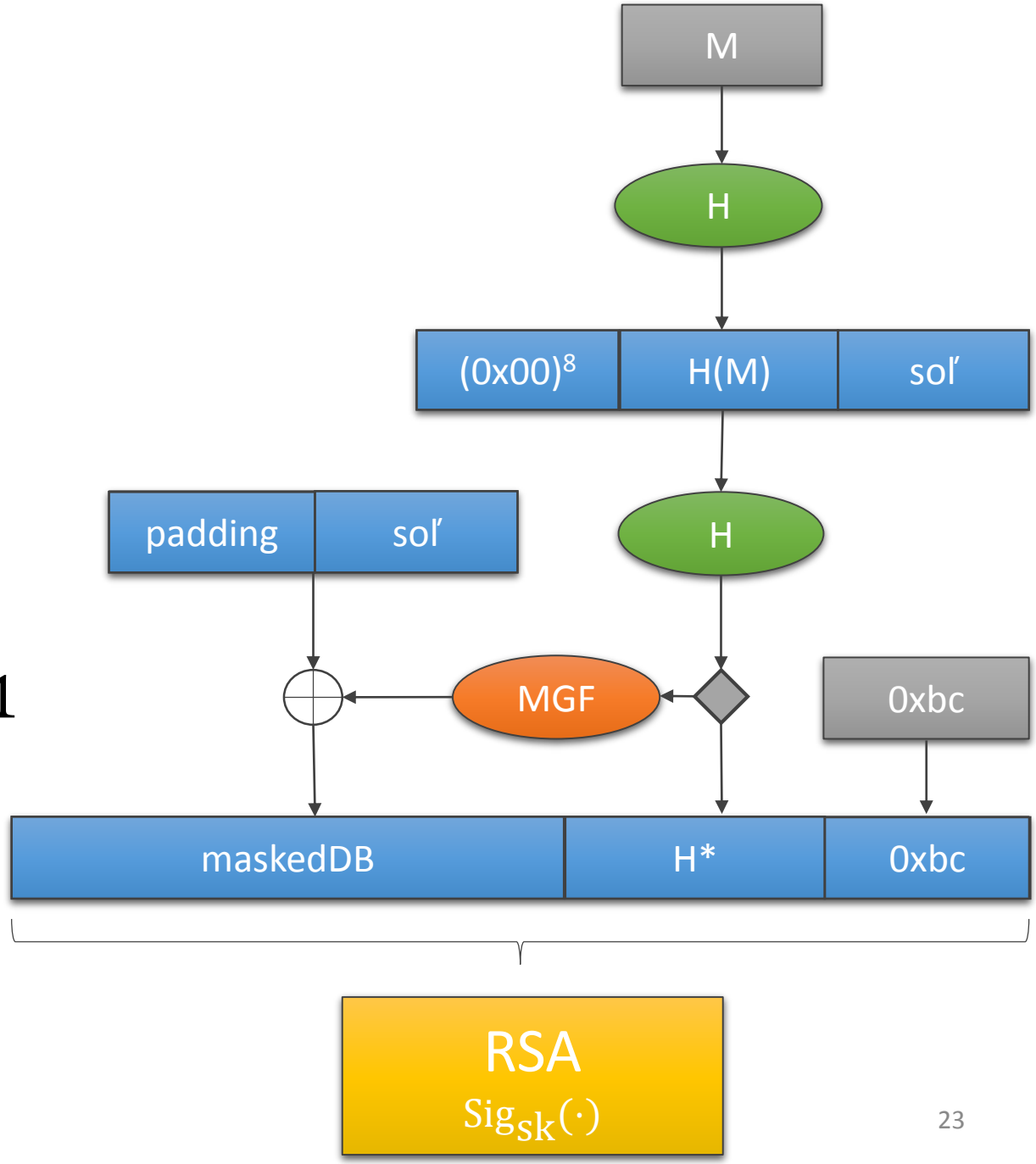


RSA-PSS - podisovanie

MGF - Mask Generation Function
(použité aj v RSA-OAEP)

soľ - sekvencia náhodných bajtov

padding=$0x00 \| \cdots \| 0x00 \| 0x01$



RSA-PSS - overovanie

$\mathrm{Vrf}_{pk}(m, \sigma)$

1. Parsuj a over: $\sigma^e \bmod n \mapsto \mathrm{maskedDB} \parallel H^* \parallel \mathrm{0xbc}$
2. $\mathrm{DB} = \mathrm{maskedDB} \oplus \mathrm{MGF}(H^*)$
3. Parsuj a over: $\mathrm{DB} \rightarrow \mathrm{padding} \parallel \mathrm{soľ}$
4. Over, či $H^* = H\left((\mathrm{0x00})^8 \parallel H(m) \parallel \mathrm{soľ}\right)$

Podpis je validný, ak sedia všetky kontroly

RSA – podpisové schémy

RSA-PKCS #1 v 1.5 – bez dôkazu bezpečnosti, bez známych útokov, útoky na rôzne varianty

- V praxi často používané
- „Deprecated“

RSA-FDH – dôkaz bezpečnosti v modeli s náhodným orákulom

- V praxi málo používané – ako nájsť h.f. s obrazom 2048 bitov?

RSA-PSS – dôkaz bezpečnosti v modeli s náhodným orákulom

- Štandard PKCS #1 v 2.1
- V súčasnosti najlepší spôsob ako podpisovať s RSA >= 1024 bitov

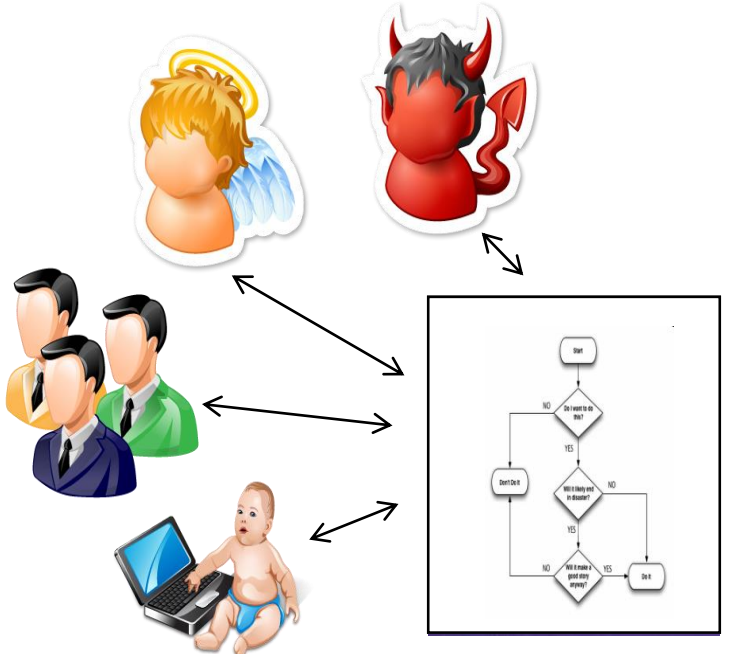
*Náhodné orákulum

- Kompikovanú h.f. nahradíme jednoduchým matematickým objektom – náhodnou funkciou
- T.j. odtlačok Y = H(M) je vybraný uniformne náhodne pre každé M
  - Y je nezávislé na M
  - Pre rôzne M_1,M_2, ..., odtlačky $Y_i = H(M_i)$ sú navzájom úplne nezávislé
  - Je ťažké nájsť kolíziu
  - Je ťažké hľadať predobrazy
  - atď.

*Model s náhodným orákulom

[Bellare-Rogaway, 1993]

1. „Predstieraj “, že H je náhodná
2. Navrhni kryptosystém, ktorý bude využívať takú (náhodnú) H
   - Dokáž bezpečnosť systému vzhľadom na „náhodné orákulum“
3. Nahraď náhodné orákulum skutočnou h.f.
   - Dúfaj, že systém ostane bezpečný



*Model s náhodným orákulom: odôvodnenie

- Nech S je schéma (napr. podpisová) využívajúca h.f. H
- Ak dokážeme, že S je bezpečná za predpokladu, že H je náhodná, potom akýkoľvek útok na S musí využiť nejakú „nonrandom“ vlastnosť hašovacej funkcie H
  - T.j. mali sme si vybrať lepšiu H, bez tejto „nonrandom“ vlastnosti
- Problém: ako vieme, ktoré „nonrandom“ vlastnosti sú dôležité / využiteľné útočníkom?
- Problém: Existujú podpisové schémy, ktoré
  - Sú bezpečné vzhľadom na náhodné orákulum
  - Sú ľahko prelomiteľné pre každú skutočnú h.f.

ElGamal

- Taher ElGamal, 1984
- ElGamalovu šifrovaciu schému nie je možné použiť aj na podpisovanie
  - Len veľmi málo schém je bijekcia ako RSA

Inicializácia – Gen: (rovnaké ako v šifrovacej schéme)

1. Vygeneruj vhodné prvočíslo p, generátor g grupy $\mathbb{Z}_p^*$
2. $x \xleftarrow{\$} \{2, \dots, p-2\}$
3. $y = g^x \bmod p$
4. Verejný kľúč $\mathrm{pk} = (p, g, y)$
5. Súkromný kľúč $\mathrm{sk} = x$

ElGamal

Podpisovanie - $\mathrm{Sig}_{sk}(m)$:

1. $k \xleftarrow{\$} \mathbb{Z}_{p-1}$,
   - pričom $\gcd(k, p-1) = 1$
2. $r = g^k \bmod \mathrm{p}$
3. $s = (H(m) - xr) \cdot k^{-1} \bmod (p-1)$
4. $\sigma = (r, s)$

Overovanie - $\mathrm{Vrf}_{pk}(m, (r, s))$

1. Over, či $1 \leq r < p$
2. Over, či $\mathrm{y}^r \cdot r^s \equiv g^{H(m)} \pmod{p}$

ElGamal

- Korektnosť
  - $1 \leq r < p$ triviálne platí (ak (r, s) je validný podpis správy m)
  - $y^r \cdot r^s \equiv g^{xr} \cdot g^{ks} \equiv g^{xr+ks} \equiv g^{H(m)} \pmod{p}$
- Efektívnosť
  - Sig – jedno modulárne umocnenie, môže sa predvypočítať
  - Vrf – 3 modulárne umocnenia
  - Dĺžka podpisu $\backsim$ dvojica prvkov zo $\mathbb{Z}_p$
- Bezpečnosť
  - Výpočítať súkromný kľúč x z y je zrejme ťažké - problém diskrétneho logaritmu

ElGamal - bezpečnosť

- Predpovedateľné k vedie k odhaleniu súkromného kľúča

$$s = (H(m) - xr) \cdot k^{-1} \bmod (p-1)$$

$$\Rightarrow \qquad x = (H(m) - ks) \cdot r^{-1} \bmod (p-1)$$

- Dvojnásobné použitie rovnakého k vedie k odhaleniu s.k.
  - $m_1, (r_1, s_1) \Rightarrow H(m_1) \equiv xr + ks_1 \pmod{p-1}$
    $m_2, (r_2, s_2) \Rightarrow H(m_2) \equiv xr + ks_2 \pmod{p-1}$
  - Teda $H(m_1) - H(m_2) \equiv k(s_1 - s_2) \pmod{p-1}$
  - Nech $d = \gcd(s_1 - s_2, p-1)$
  - Ak $d = 1$ tak $k = \big(H(m_1) - H(m_2)\big)(s_1 - s_2)^{-1} \pmod{p-1}$
  - V opačnom prípade rovnicu podelíme d, vyriešime a následne odtestujeme d kandidátov na k

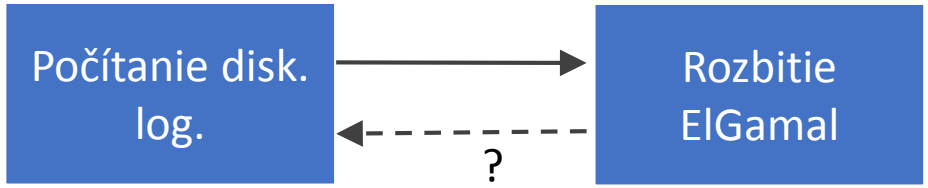
ElGamal - bezpečnosť

- Kontrola $1 \leq r < p$ je nutná
  - Nech (r, s) je podpis m, t.j. $g^{H(m)} \equiv y^r r^s \pmod{p}$
  - V opačnom prípade je možné vypočítať falošný podpis (r', s') pre nejakú inú správu $m' \neq m$
  1. Vypočítajme $u = H(m') \cdot H(m)^{-1} \bmod (p-1)$ (ak je $H(m)$ nesúdeliteľné s $p-1$)
  $$g^{H(m')} \equiv g^{H(m)u} \equiv y^{ur} \cdot r^{us} \pmod{p}$$
  2. Nech $s' = us \bmod (p-1)$ a vypočítajme r' pre ktoré platí
  $$\mathrm{r}' \equiv ru \pmod{p-1}$$
  $$r' \equiv r \pmod{p}$$
  - (r', s') je podpis m'
  - Keďže H je odolná voči kolíziám, s veľkou pravdepodobnosťou je $r' \geq p$
    - V opačnom prípade je $u = 1$ a teda máme kolíziu $H(m') \equiv H(m) \pmod{p-1}$

ElGamal – bezpečnosť

- Bleichenbacherov útok (1996)
  - Falšovanie podpisov ak g má malé prvočíselné faktory a $g|(p-1)$
  - T.j. $g = 2$ je zlá voľba
  - Poznámka: pre problém diskrétneho logaritmu sú všetky voľby generátora g ekvivalentné

- Zhrnutie
  - Vieme počítať disk. log. ⇒ vieme rozbiť ElGamal
  - Neexistuje formálny dôkaz bezpečnosti ElGamal
    - Ani v modeli s náhodným orákulom



DSA

- Súčasť štandardu DSS - FIPS 186-4 (DSA, RSA, ECDSA)
- Bezpečnosť súvisí s problémom diskrétneho logaritmu
- Vychádza z ElGamalovej schémy

Inicializácia - Gen:

1. Vygeneruj prvočísla p, q (napr. $|p| = 2048, |q| = 256$), pričom $q|(p-1)$
2. Vygeneruj $h \xleftarrow{\$} \mathbb{Z}_{p-1}$, vypočítaj $g = h^{p-1/q} \bmod p$
3. Súkromný kľúč: $x \xleftarrow{\$} \mathbb{Z}_q^*$
4. Verejný kľúč: $y = g^x \bmod p$ a parameter p, q, g

DSA

Podpisovanie - $\mathrm{Sig}_{sk}(m)$:

1. $r = g^k \bmod p \bmod q$, kde $k \xleftarrow{\$} \mathbb{Z}_q^*$
2. $s = k^{-1}(H(m) + xr) \bmod q$
3. Ak $r = 0$ alebo $s = 0$ začni znova krokom 1 (veľmi málo pravdepodobné)
4. $\sigma = (r, s)$

Overovanie - $\mathrm{Vrf}_{pk}(m, (r, s))$

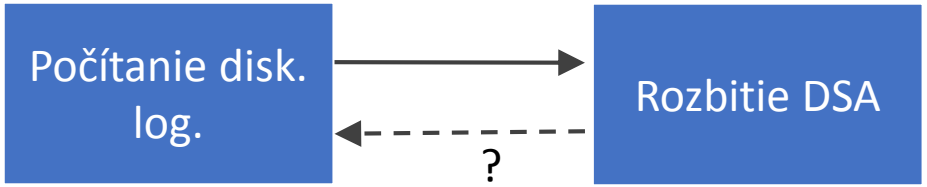
1. Over, či $r, s \in \mathbb{Z}_q^*$
2. $u_1 = H(m) \cdot s^{-1} \bmod q$
   $u_2 = r \cdot s^{-1} \bmod q$
3. Over, či $(g^{u_1} \cdot y^{u_2} \bmod p) \bmod q = r$

DSA

- Korektnosť:
  - $(g^{u_1} \cdot y^{u_2} \bmod p) \bmod q = g^{H(m)s^{-1}+xrs^{-1}} \bmod \text{p} \bmod \text{q}$
    $= g^{s^{-1}(H(m)+xr)} \bmod p \bmod q = g^k \bmod p \bmod q$
    $= r$
- Ak $r = 0$, potom podpis nezávisí na súkromnom kľúči x (podpis možno ľahko falšovať)
- Ak $s = 0$, potom $s^{-1} \bmod q$ neexistuje
- Predpovedateľné k vedie ku kompromitácii súkromného kľúča
  - (2010) Sony PS3 ECDSA s konštantným k
  - Existuje variant deterministickej voľby k odvodenej zo s.k. a $H(m)$ (RFC 6979)

DSA

- Parametre p, q, g môžu byť zdieľané medzi viacerými používateľmi systému
  - Málo používané, potreba zabezpečiť, že neboli vyberané útočníkom
  - Existuje overiteľná procedúra na generovanie týchto parametrov (súčasť štandardu)



Bezpečnosť DSA:

- Súvisí s problémom diskrétneho logaritmu
  - Vieme počítať disk. log. ⇒ vieme rozbiť DSA
- Neexistuje formálny dôkaz bezpečnosti DSA
  - Ani v modeli s náhodným orákulom

DSA, RSA, ElGamal

- DSA je výkonovo porovnateľné s ElGamalom
  - DSA má kratší podpis
- DSA má rýchlejšie podpisovanie ako overovanie
  - Vhodné pre výpočtovo slabé zariadenia – Smart Karty
- DSA ma kratší podpis ako RSA
- RSA má rýchlejšie overovanie ako podpisovanie
  - Podpis robíme raz, overovať môžeme viac krát

- DSA navrhovalo NSA, možno obsahuje zadné vrátka
  - Výber algoritmov DSA nebol verejný

DSA, RSA, ElGamal

n	Dĺžka hašu	Dĺžka podpisu	Bezpečnosť
RSA-FDH			
1024	1024	1024	80
2048	2048	2048	112
3072	3072	3072	128
RSA-PSS			
1024	160-512	1024	80
2048	160-512	2048	112
3072	160-512	3072	128

p	q	Dĺžka hašu	Dĺžka podpisu	Bezpečnosť
DSA				
1024	160	160	320	80
2048	224	224	448	112
3072	256	256	512	128
ElGamal				
1024	-	1024	2048	80
2048	-	2048	4096	112
3072	-	3072	6144	128

ECDSA

- Variant DSA nad eliptickými krivkami
- Problém diskrétneho logaritmu je nad eliptickými krivkami ťažší
  ⇒ môžeme použiť kratšie kľúče
- Dĺžka kľúča 160-256 bitov ekvivalentná 1024-3072 bitovému RSA
  - Ekvivalentné 80-128 bitovej bezpečnosti symetrických schém

Dĺžka kľúča	Dĺžka podpisu	Dĺžka hašu	Bezpečnosť
160	320	160	80
224	448	224	112
256	512	256	128

ECDSA

Inicializácia Gen:

1. Voľba parametrov eliptickej krivky (CURVE, G, n)
   - CURVE – rovnica / popis eliptickej krivky (väčšinou trojica (p, a, b), p je prvočíslo – definuje poľe $\mathbb{F}_p$, a, b sú konštanty popisujúce krivku $y^2 = x^3 + ax + b$
   - G – bod na krivke - generátor podgrupy rádu n
   - Voľba parametrov krivky je netriviálna – možnosť vybrať zo štandardných „pomenovaných“ kriviek – napr. z NIST
2. Súkromný kľúč: $x \xleftarrow{\$} \{1, \dots, n-1\}$
3. Verejný kľúč: $Y = x \times G$

   $\times$ znamená skalárne násobenie bodu na krivke (t.j. $G + G + \cdots + G$)

ECDSA

Podpisovanie - $\mathrm{Sig}_x(m)$:

1. $k \xleftarrow{\$} \{1, \dots, n-1\}$
2. $(x_1, y_1) = k \times G$
3. $r = x_1 \bmod n$
4. $s = k^{-1}(H(m) + rx) \bmod n$
5. Ak $r = 0$ alebo $s = 0$ začni znova krokom 1 (veľmi málo pravdepodobné)
6. $\sigma = (r, s)$

Overovanie - $\mathrm{Vrf}_Y(m, (r, s))$

1. Over, či $r, s \in \{1, \dots, n-1\}$
2. $w = s^{-1} \bmod n$
3. $u_1 = H(m) \cdot w \bmod n$
   $u_2 = rw \bmod n$
4. $(x_1, y_1) = u_1 \times G + u_2 \times Y$
5. Over, či $r \equiv x_1 (\bmod n)$

Výkonové porovnanie

	podpisovanie [operácie/s]	overovanie [operácie/s]
RSA-1024	6 100	93 281
RSA-2048	857	27 496
RSA-4096	118	7 370
ECDSA-224 (nistp224)	15 375	7 349
ECDSA-256 (nistp256)	9 024	3 697
ECDSA-521 (nistp521)	3 252	1 501

i7-2600, 3.40GHz, Ubuntu 12.04 LTS 64-bit, openssl 1.0.1, 8kB bloky

(Ne)determinizmus

- Nederministické schémy
  - Pre jednu správu existuje veľa podpisov
  - ElGamal, DSA, RSA-PSS
  - Generovanie náhodných čísel je ťažké, hlavne na smart kartách
  - Existujú metódy ako ich zmeniť na deterministké schémy bez straty bezpečnosti
    - RFC 6979 pre DSA
- Deterministické schémy
  - Pre každú správu existuje len jeden podpis
  - RSA-FDH, RSA-PKCS #1 v 1.5

Bezpečnosť podpisov v čase

Čas útočníka na rozbitie schémy

- Autentizácia: ~1 hodina – potom je už neskoro
- Digitálny podpis: 20 rokov a viac…

Treba brať do úvahy aj útoky odhalené až v budúcnosti

Dĺžky kľúčov

Ochrana	Symetrický kľúč	Výstup hašovacej funkcie	RSA modul	Eliptická krivka
~ 4 roky	80	160	1248	160
~ 20 rokov	112	224	2432	224
~ 30 rokov	128	256	3248	256
	256	512	15424	512

- Podľa správy ECRYPT II (2012)
- Porovnanie rôznych metód: www.keylength.com
- Bezpečnosť vs. výpočtové nároky

Timestamping

- Časom môže dôjsť ku kompromitácii podpisovej schémy / súkromného kľúča podpisujúceho
- Eva odhalí súkromný kľúč Boba ⇒ môže falšovať Bobov podpis na akejkoľvek správe
- Autentickosť všetkých Bobových podpisov pred kompromitáciou kľúča je teda taktiež otázna
- Problém je, že nevieme určiť kedy bola správa podpísaná
- Timestamping – dôkaz, že správa bola podpísana v určitom čase

Timestamping

Ako na to

Nech pub je nejaká nepredpovedateľna verejne dostupná informácia (napr. hodnota akcí na burze)

Postup: nech m je správa, ktorú chce Bob podpísať

1. Bob vypočíta $z = H(m)$
2. Bob vypočíta $z' = H(z' \| \text{pub})$
3. Bob vypočíta $y = Sig_{sk}(z')$
4. Bob zverejní (z, pub, y) v novinách

Zjavne, podpis nemohol byť vytvorený po tom, ako bolo (z, pub, y) zverejnené

Nepredpovedateľnosť pub znamená, že podpis nemohl byť vytvorený ani predtým

Timestamping

Dôveryhodný poskytovateľ časových pečiatok

- Autorita (TSA), ktorá garantuje časovú pečiatku dokumentu
- Alica chce opečiatkovať dokument m
  1. A vypočíta $z = H(m)$ a pošle ho TSA
  2. TSA vygeneruje pečiatku ts
     - vypočíta $z' = H(z \parallel ts)$
     - podpíše $y = Sig_{sk_{TSA}}(z')$
  3. A udržuje (ts, y) ako dôkaz
- Každý môže overiť podpis a pečiatku
- TSA nevidí samotný dokument

Slepé podpisy

- Interaktívny protokol medzi podpisujúcim S a odosielateľom A
- Podpisujúci nevie čo podpisuje, no odosielateľ získa korektný podpis zvoleného dokumentu
  - Elektronické peniaze, Elektronický notár

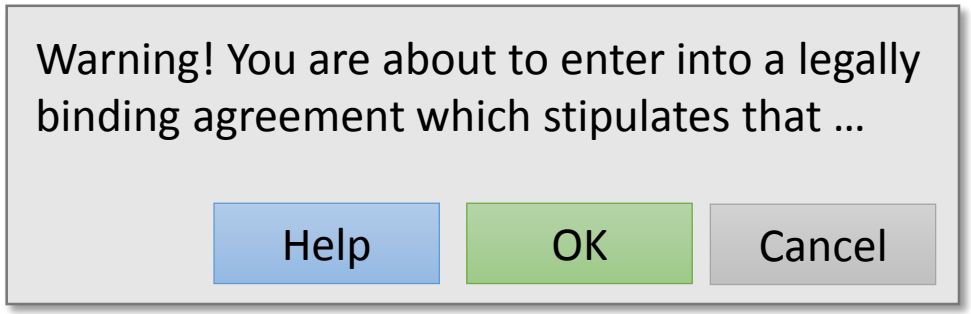
1. $A \to S: r = H(m) \cdot x^e \bmod n$, kde $x \overset{\$}{\leftarrow} \mathbb{Z}_n^*$
2. $S \to A: s = r^d \bmod n$, t.j. S podpíše prijatú správu a pošle ju A
3. A vypočíta podpis m:
4. $$s \cdot x^{-1} \bmod n = \left(r^d \bmod n\right) \cdot x^{-1} \bmod n$$
$$= H(m)^d \cdot x^{ed} \cdot \mathrm{x}^{-1} \bmod n$$
$$= H(m)^d \bmod n$$

Nepopierateľnosť autorstva v praxi

- Teoreticky môžeme hovoriť, že digitálne podpisy poskytujú nepopierateľnosť autorstva
  - Bez znalosti súkromného kľúča nikto nevie vytvoriť validný podpis
- V praxi je to však veľmi ťažké dosiahnuť
  - Existencia manuálneho podpisu znamená, že podpisujúci mal dočinenia a videl podpísaný dokument
  - Existencia dig. podpisu znamená, že niekedy niečo vykonalo matematickú operáciu nad nejakými dátami

Nepopierateľnosť autorstva v praxi

- Validný digitálny podpis môže byť veľmi jednoducho popretý
  - „Software ma nedostatočne upozornil na dôsledky vykonaných akcií“
    - „babička klikla na zlé tlačidlo a prišla o dom“
  - „Urobil to vírus“
    - Univerzálna výhovorka
  - Používateľ zverejnil svoj súkromný kľúč
    - Môžeme ho potrestať za ľahostajnosť, ale nie za obsah podpísaného dokumentu
  - Ak má podpis časovú pečiatku, používateľ bude tvrdiť, že v čase podpisovania ešte nevedel o kompromitácii svojho súkromného kľúča
- Potrebujeme vytvoriť ekvivalent „informovaného súhlasu” (WYSIWYS)

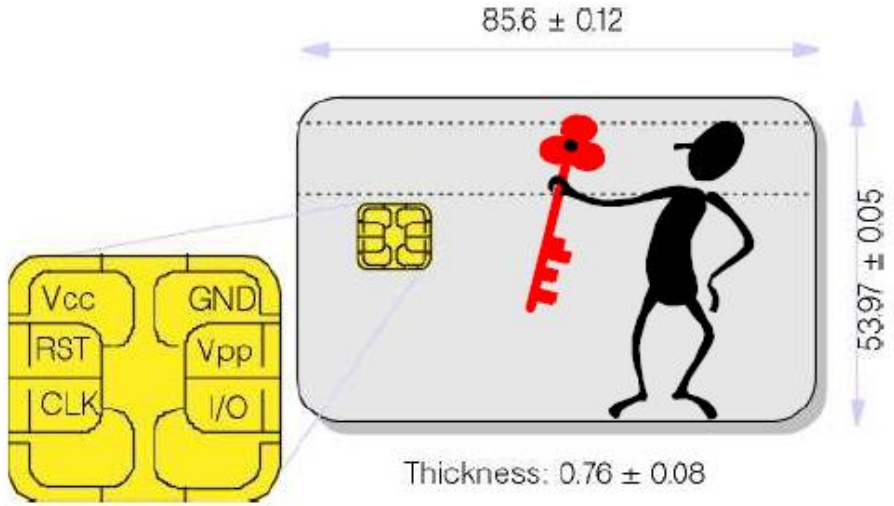


Smart karty, bezpečný hardware

Súkromný kľúč musí zostať utajený!

Musí byť bezpečne

- Generovaný
- Uložený
- Používaný
- Zálohovaný
- Vymazaný



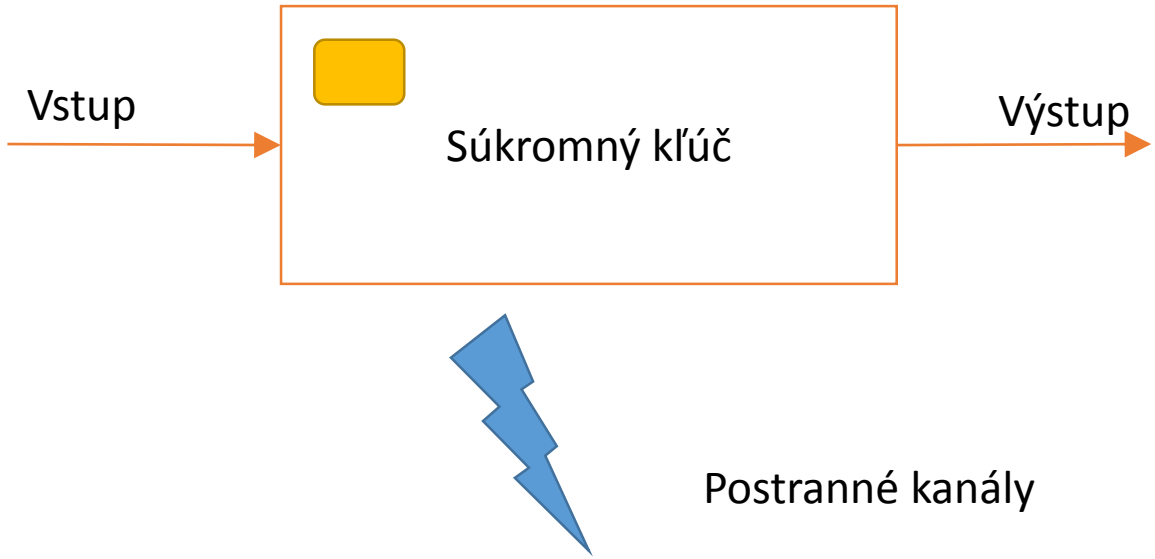
Počítač neposkytuje dostatočnú bezpečnosť

Riešenie: smart karty

- Pre zavedenie nepopierateľnosť autorstva je bezpečný hardware kľúčový

Útoky postrannými kanálmi

- Timing útoky
- Meranie napätia
- EM merania – TEMPEST
- Meranie hlučnosti
- ...

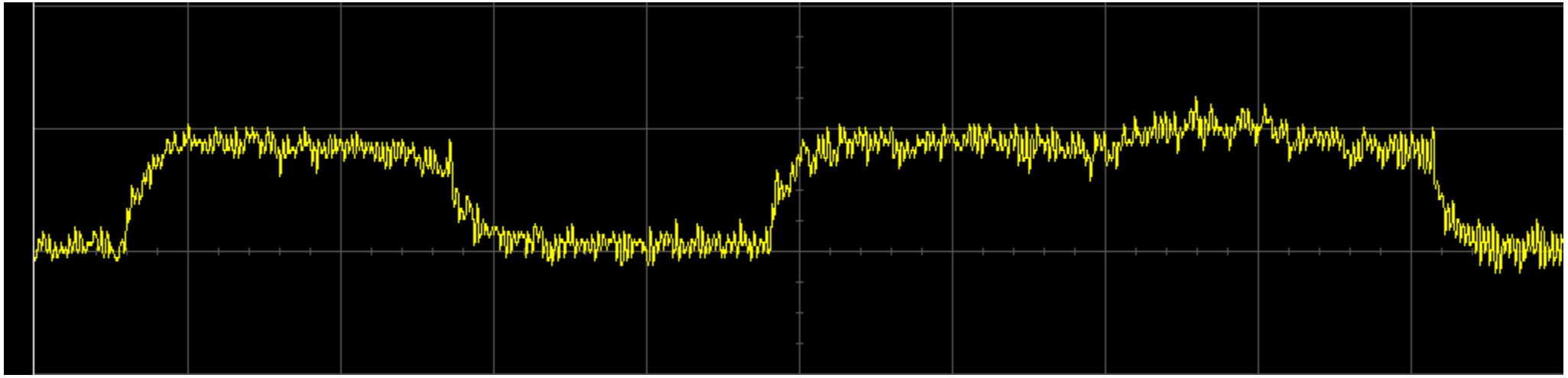


Timing útoky

- Čas behu nejakého kroku algoritmu môže závisieť od dát
- Čas beh algoritmu „square-and-multiply“ na počítanie modulárneho umocňovania závisi od počtu 1 v kľúči
- 2003, Boneh, Brumley – timing útok na SSL využívajúci slabinu v implementácii RSA využívajúcu optimizáciu cez čínsku zvyškovú vetu
  - Na odhalenie súkromného kľúča stačilo niekoľko hodín

- Ochrana: „maskovacie” techniky v implementácii algoritmov

Meranie napätia



- Priebeh napätia počas výpočtu RSA podpisu na smart karte
- Ľavý vrchol – priebeh napätia CPU počas kroku algoritmu bez násobenia
- Pravý vrchol – napätie CPU počas kroku s násobením
- Môžeme tak rozoznať bity súkromného kľúča

Aplikácie digitálnych podpisov

- V praxi je podpisovanie správ pre iných používateľov (t.j. nie pre automatizované procesy) veľmi zriedkavé
  - Používatelia zväčša overia autenticitu správy na základe obsahu, nie digitálneho podpisu
- Príklad: S/MIME a digitálne podpisy
  - Vývojari S/MIME diskutovali či použijú dig. podpisy v ich mailing liste
    - Správy od ľudí, ktorých poznáte môžete overiť na základe ich obsahu
    - Podpis od ľudí, ktorých nepoznáte je irelevantný
  - Používanie digitálnych podpisov je taktiež vcelku obtiažne
  - Výsledok: vývojari S/MIME nepoužívaju dig. podpisy na posielanie správ

Aplikácie digitálnych podpisov

- WEB – SSL/TLS
- Autentizácia software (Microsoft, Apple store, Google play, …)
- Bankové karty – internet banking
- PKI
- Bezpečný email – S/MIME
- ...

Štandardy

- DSS - FIPS 186-4
  - ECDSA + SHA256
- PKCS - Public-Key Cryptography Standards
  - Implementačné štandardy
  - PKCS #1: RSA Cryptography Standard
  - PKCS #5: Password-Based Cryptography Standard
  - PKCS #7: Cryptographic Message Syntax Standard
  - PKCS #10: Certification Request Syntax Standard
  - PKCS #11: Cryptographic Token Interface Standard
  - PKCS #12: Personal Information Exchange Syntax Standard

Záver

Bez ohľadu na podpisovú schému, digitálne podpisy vyžadujú nasledovné

1. **Kvalitné algoritmy**
   - Bezpečnosť niektorých podpisových schém bola kompromitovaná
2. **Kvalitná implementácia**
   - Dobrý algoritmus implementovaný s chybou nefunguje
3. **Súkromný kľúč musí zostať utajený**
   - V prípade jeho odhalenie môže útočník falšovať podpisy
4. **Identita držiteľa verejného kľúča musí byť overiteľná**
   - PKI
5. **Používatelia (a software) musia dodržiavať postup podpisovacieho protokolu**

Záver

- Digitálne podpisy vs. elektronické podpisy vs. ručné podpisy
- Digitálny podpis: autenticita, integrita, nefalšovateľnosť, nepopierateľnosť pôvodu
- Najznámejšie podpisové schémy
  - RSA-FDH, RSA-PKCS #1 v1.5, RSA-PSS
  - ElGamal, DSA, ECDSA
- Použitie rovnakých kľúčov na šifrovanie a podpisovanie sa neodporúča
- Časové pečiatky
- Bezpečný hardware je nevyhnutný
  - Útoky postrannými kanálmi

Ďakujem za pozornosť