České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra telekomunikační techniky

Asymetrické kryptosystémy I

Ing. Tomáš Vaněk, Ph.D. tomas.vanek@fel.cvut.cz

Osnova

- obecné informace
- IFP
  - RSA
- DLP
  - Diffie-Hellman
  - DSA
- ECDLP
- Jiné

Šifrovací algoritmy - rozdělení

symetrické

- blokové (DES, 3DES, IDEA, RC6, Blowfish, CAST, AES - Rijndael….)
- proudové (Enigma, SEAL, RC4, A5,…)

asymetrické

- IFP (RSA, Rabin-Williams, Lucas,…)
- DLP (DSA, Diffie-Hellman, El-Gamal, …)
- ECDLP (ECDSA, PSEC,…)

Šifrovací algoritmy - srovnání

Symetrické algoritmy

- dnes velmi rychlé algoritmy hodící se k šifrování velkých objemů dat
- malé (relativně) délky klíčů (128-bit, 256-bit)
- obě dvě strany musí znát stejný tajný klíč
- problém distribuce/změny klíčů v případě velkého množství komunikujících stran
- nelze prokázat totožnost autora zašifrovaných dat (data mohla zašifrovat kterákoliv strana vlastnící klíč)

Šifrovací algoritmy - srovnání

Asymetrické algoritmy

- pomalé (100-1000x)
- velké délky klíčů (1024b, 2048b,…)
- různé klíče pro šifrování a dešifrování
- **klíče spolu vzájemně souvisejí**

Hybridní systémy

- využívají kladné vlastnosti obou předchozích skupin.
- zpráva je zašifrována symetrickým algoritmem s náhodně vygenerovaným klíčem a tento klíč je zašifrován asymetrickým algoritmem
- příklad - PGP

Použití kryptosystémů veřejného klíče

Tři možné oblasti použití:

- **šifrování /dešifrování**
- **podepisování (digitální podpis)**
- **výměna klíčů**

Ne všechny kryptosystémy veřejného klíče umožňují realizovat všechny činnosti!

Šifrování s asymetrickým algoritmem

1) Vygeneruji pár klíčů – veřejný (šifrovací) soukromý (dešifrovací).
   - Veřejný klíč může být volně dostupný.
   - Lze ho zveřejnit např. na Internetu.
   - Není žádný důvod ho tajit.
   - Soukromý (dešifrovací) klíč je potřeba uchovávat v tajnosti.
2) Pokud mi někdo chce poslat tajnou zprávu, zašifruje ji mým veřejným klíčem.
3) Pouze já (jakožto držitel tajného klíče) mohu dešifrovat zprávu svým soukromým klíčem.

EN terminologie: Veřejný = Public Soukromý/Tajný = Private

Podepisování s asymetrickým algoritmem

1) Vygeneruji dva klíče – soukromý (pro vytváření podpisu) a veřejný (pro ověřování podpisu).

- Veřejný klíč může být volně dostupný.
- Není žádný důvod ho tajit.
- Soukromý klíč je potřeba uchovávat v tajnosti.

2) Pokud chci podepsat zprávu, podepíši (tato transformace je identická s šifrováním) ji s využitím svého soukromého klíčem.

3) Pokud chce někdo ověřit autorství zprávy, ověřím platnost podpisu (= transformace dešifrování) s využitím mého veřejného klíče. Protože pouze držitel tajného klíče mohl zprávu podpesat…

Nezaměňovat podepisování a šifrování !

Autentizace pomocí asymetrické kryptografie

- Alice pošle „výzvu“ někomu, kdo se představil jako Bob
  - Náhodné číslo # (nonce) zašifrované Bobovým veřejným klíčem
  - Pokud je neznámý skutečně Bob, dokáže zprávu dešifrovat

{Udělej „HAF!“}$_{PK(Bob)}$ →

← HAF!

Bob

Alice

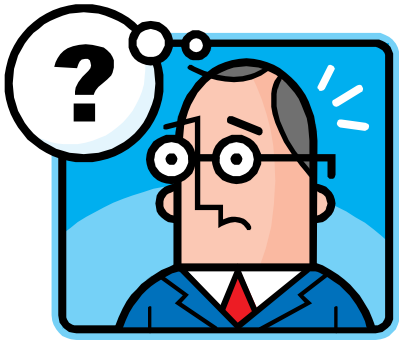
{Řekni „sýr“}$_{PK(Bob)}$ →

← ?????

Eva

Co neřeší asymetrická kryptografie

Stačí takto definované principy asymetrické kryptografie (soukromý a veřejný klíč) k bezpečné prokázání toho, že konkrétní klíč patří konkrétní entitě (osoba, systém)?



Certifikát, Certifikační autorita a PKI

- Důvěryhodná třetí strana (Trusted Third Party) - CA ověřuje vlastníků klíčových párů
- CA autentizuje Boba & vytváří certifikát obsahující jeho veřejný klíč (spojuje Bobovu identitu s jeho veřejným klíčem)

- Public Key Infrastructure (PKI): CA + všechno další potřebné pro šifrování veřejným klíčem

Detailní informace viz přednáška o elektronickém podpisu.

Využití asymetrických kryptosystémů

- PGP – Pretty Good Privacy
- GPG – implementace OpenPGP
- SSH – Secure Shell
- SSL/TLS – Secure Socket Layer/Transport Layer Security
- IKE (Internet Key Exchange) – součást IPsec
- ESP – Encapsulated Secure Payload – součást IPsec
- …

Historie

- 1970 James Ellis (UK CESG) popsal myšlenku asymetrické kryptografie – „Secure Non-secret Encryption“
  - odtajněna až v roce 1999
  - v CESG nejprve objevili RSA a až potom DH – veřejně to bylo obráceně.
- 1970, W.Diffie & M.E.Hellman představili myšlenku kryptosystému veřejného klíče – zatím bez konkrétní možné realizace
- 1973 Clifford Cocks (GCHQ - Government Communications Headquarters) možnost konstrukce kryptosystémů veřejného klíče na stejném principu
  - veřejně publikováno v roce 1997

Historie

- 1976: W.Diffie & M.E. Hellman veřejně publikovali první asymetrický kryptografický algoritmus - bezpečná výměna tajného klíče
- 1978: L.M. Adleman, R.L. Rivest and A. Shamir – publikovali šifrování pomocí RSA
- 2011 - C. Cocks objevil první bezpečné schéma pro IBE (Identity Based Encryption) založené na kvadratických reziduích ve složených grupách
  - nepoužívá se v praxi kvůli velkému nárůstu velikosti ŠT (řádově 100-1000x)
  - jedno z mála IBE schémat, která nepoužívají bilinéární párování, ale závisí na lépe prozkoumaných matematických problémech

- Od 70. let 20.století bylo představeno mnoho kryptosystémů veřejného klíče
- Řada z nich není bezpečných, nebo jsou velmi nepraktické (např.: vyžadují příliš velké klíče, nebo ŠT je výrazně větší než OT).
- Pouze několik málo algoritmů je současně jak bezpečných tak i praktických.
- Některé algoritmy lze použít jen šifrování, některé jen k podepisování a některé k obojímu.
- Pouze tři algoritmy je možné použít jak pro šifrování, tak pro podepisování: RSA, El-Gamal, a Rabinův.

Faktorizace

- obecně jde o problém rozložení čísla na součin menších čísel
  - nejčastěji se myslí rozklad celého čísla na součin prvočísel
  - lze rozkládat i jiné objekty (např. polynom x^4-4 lze vyjádřit jako součin dvou polynomů $(x^2-2)\cdot(x^2+2)$)
  - libovolné celé číslo lze jednoznačně rozložit na součin prvočísel
- rozklad složeného čísla na prvočísla je považován za obtížný – dnes neznáme **efektivní** algoritmus
- není známa klasifikace tohoto problému do třídy složitosti
  - předpokládá se, že neleží v P, NP-complete, co-NP-complete

RSA

- 1978
- pojmenován po svých třech objevitelích:

  Ron Rivest

  Adi Shamir

  Leonard Adleman
- první asymetrický **šifrovací** algoritmus
- nejznámější kryptosystém veřejného klíče
- používán v řadě protokolů (např.: SSL/TLS, IPsec,…)
- umožňuje jak šifrování, tak i podepisování

RSA

- založen na umocňování celých čísel modulo prvočíslo
- používá velká prvočísla (více než 100 dekadických míst ~1024 bitů)
- dvě prvočísla *p* a *q*, které mají *x* a *y* dekadických čísel je možné podle vynásobit s obtížností $O(log_2 x \cdot log_2 y)$ bitových operací
- opačná úloha tj. pro dané $n = p \cdot q$ najít prvočíselné dělitele je **výrazně** těžší.
- toto tvrzení však dodnes nebylo dokázáno (tj. je možné, že k vyřešení IFP není nutné provádět faktorizaci)
- složitost při šifrování je $O((\log n)^3)$
- faktorizace pokusným dělením zabírá $O(e^{\log n \cdot \log \cdot \log n})$ operací
- nejefektivnější faktorizační metoda (Brent-Pollardova ρ-metoda) vyžaduje $O(\frac{e^{\sqrt{2 \ln p \ln \ln p}}}{\ln p})$

RSA – generování klíčových párů

- Nechť p a q jsou velká a náhodně zvolená prvočísla
- Číslo $N = pq$ nazveme modul
- Zvolíme e, které je relativním prvočíslem k $\varphi(N)=(p-1)(q-1)$
- Nalezneme d takové, že $ed = 1 \bmod \varphi(N)$

- Veřejný klíč je dvojice (N,e)

- Soukromý klíč je dvojice (N,d)

RSA – šifrování a dešifrování

Šifrování zprávy M: $C = M^e \bmod N$

Dešifrování zprávy M: $M = C^d \bmod N$

N, e jsou veřejně známé.

Pokud útočník je schopný najít rozklad N, pak je jednoduché najít *d*, protože ed = 1 mod (p-1)(q-1)

Rozklad modulu je cesta k prolomení RSA.
Není jisté, zda-li je faktorizace jediná možnost jak prolomit RSA.

RSA

- před šifrováním je potřeba upravit text – formátovací pravidla - PKCS#1 (http://www.rsasecurity.com/rsalabs/node.asp?id=2124)
- jinak hrozí, že pro vybrané vstupní texty M bude zašifrovaný text lehce prolomitelný
- bezpečnost RSA závisí na obtížnosti rozkladu modulu N

Na stejném zařízení je

- při HW implementaci :
  - RSA cca 1000x pomalejší nežli DES.
- při SW implementaci :
  - RSA cca 100x pomalejší nežli DES.

Efektivita RSA

RSA s velkými moduly je neefektivní

- Se vzrůstající délkou modulu roste bezpečnost pomalu, ale výpočetní nároky rychle.
  - složitost RSA-1024 je srovnatelná se symetrickou šifrou s klíčem délky 80bitů
  - složitost RSA-2048 přibližně odpovídá klíči délky 112 bitů (3-DES)
  - složitost RSA-3072 přibližně odpovídá klíči délky 128 bitů (např. AES-128)
  - složitost RSA-7680 přibližně odpovídá klíči délky 192 bitů (např. AES-192)
  - složitost RSA-15,380 přibližně odpovídá klíči délky 256 bitů (např. AES-256)

Efektivita RSA

- Výkon RSA s velkými moduly je velmi nízký
- Čas nutný k podepisování roste s třetí mocninou délky klíče
  - K výpočtu podpisu pomocí RSA-2048 je potřebuje přibližně 8x delší čas než s RSA-1024
    - Např. – 60ms pro RSA-1024 se změní na 480 ms pro RSA-2048
    - RSA-15,360 by zabrala 3375-ta násobek trvání RSA-1024 (~ 200 sekund)
- Existuje řešení pro bezpečné délky klíčů a rozumné výpočetní nároky – algoritmy ze skupiny “Suite B”

Proč je RSA bezpečný?

- Dosud nebyl objeven rychlý algoritmus pro rozklad velkých složených čísel.
- Nebylo dokázáno, že takový algoritmus existuje (a nebo, že neexistuje).
  - To znamená, že není pravda, že k prolomení RSA musíme faktorizovat modul, pouze dnes takový algoritmus neznáme!
- Peter Shor objevil velmi rychlý algoritmus pro kvantový počítač (až se někdy postaví…)

Metody faktorizace prvočíselných modulů

- **Pokusné dělení** (Trial Division)
- Metoda řetězových zlomků (Continued Fraction Method)
- Metoda p-1
- Metoda p+1
- **Pollardova ρ metoda**
- Kvadratické síto (Quadratic Sieve)
- Síto číselného pole (Number Field Sieve)
- **Zobecněné síto číselného pole (General NFS)**
- Metoda eliptických křivek

Největší faktorizovaný RSA modul

RSA-768 = 1230186684530117755130494958384962720772853569595334792197322452151726400507263657518745202199786469389956474942774063845925192557326303453731548268507917026122142913461670429214311602221240479274737794080665351419597459856902143413

RSA-768 = 3347807169895689878604416984821269081770479498371376856891243138898288379387800228761471165253174308773781446799948 9 × 3674604366679959042824463379962795263227915816434308764267603228381573966651127923337341714339681027009279873630891 7

Číslo	Počet číslic	Odměna	Faktorizováno
RSA-100	100		04/1991
RSA-110	110		04/1992
RSA-120	120		06/1993
RSA-129	129	$100	04/1994
RSA-130	130		10.4.1996
RSA-140	140		2.2. 1999
RSA-150	150		16. 4. 2004
RSA-155	155		22.8. 1999
RSA-160	160		1.4. 2003
RSA-200	200		9.5. 2005
RSA-576	174	$10,000	3. 12. 2003
RSA-640	193	$20,000	4.11. 2005
RSA-704	212	$30,000	Doposud se
RSA-768	232	$50,000	12.12.2009
RSA-896	270	$75,000	nepodařilo
RSA-1024	309	$100,000	faktorizovat
RSA-1536	463	$150,000	Čeká možná
RSA-2048	617	$200,000	na Vás!

DLP – Discrete Logartithm Problem

Diskrétní logaritmus

Obecná definice problému

- mějme konečnou cyklickou grupu ***G*** řádu *r*
  - prvek grupy α takový, že α je generátor grupy
  - prvky grupy α^a mod *n*, α^b mod *n* pro neznámé hodnoty *a*, *b*
- cílem je najít prvek β takový, že $\beta = \alpha^{ab}$ mod *n*
- číslo ab je diskrétní logaritmus o základu α z β
  - pro srovnání zápis týž hodnot u klasického logaritmu

$$ab = \log_{\alpha} \beta$$

- diskrétní logaritmus je zobecnění normálního logaritmu pro konečné cyklické grupy
- bezpečnost kryptostémů pak závisí na obtížnosti výpočtu diskrétního logaritmu v konečném poli GF (Galois Field)

Diffie-Helmannův protokol

- 1974 - M.J. Williamson (GCHQ)
  - Non-Secret Encryption Using a Finite Field
  - předchůdce D-H
- 1976 *Whitfield Diffie* a *Martin Hellman a* nezávisle na nich i *Ralph Merkle*
- zcela první veřejně publikovaný kryptosystém veřejného klíče
- nelze použít k šifrování
- nelze použít k podepisování
- lze ho použít „pouze“ pro výměnu klíčů
  - používá se k ustanovení sdíleného (symetrického) klíče
- bezpečnost závisí na obtížnosti řešení diskrétního logaritmu

Diffie-Helmannův protokol

Nechť p je prvočíslo a g je generátor

- pro každé $x \in \{1,2,\ldots,p-1\}$ existuje n takové, že $x = g^n \bmod p$

1. Alice vygeneruje tajné náhodné číslo a
2. Bob vygeneruje tajné náhodné číslo b
3. Alice odešle Bobovi $g^a \bmod p$
4. Bob pošle Alici $g^b \bmod p$
5. oba si spočítají sdílenou tajnou hodnotu

   $(g^b \bmod p)^a \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p$
6. toto číslo $g^{ab} \bmod p$ lze použít jako sdílený tajný klíč

Diffie-Helmannův protokol

- Bob a Alice použijí $g^{ab} \bmod p$ jako symetrický klíč
- Útočník může zjistit $g^{a} \bmod p$ a $g^{b} \bmod p$

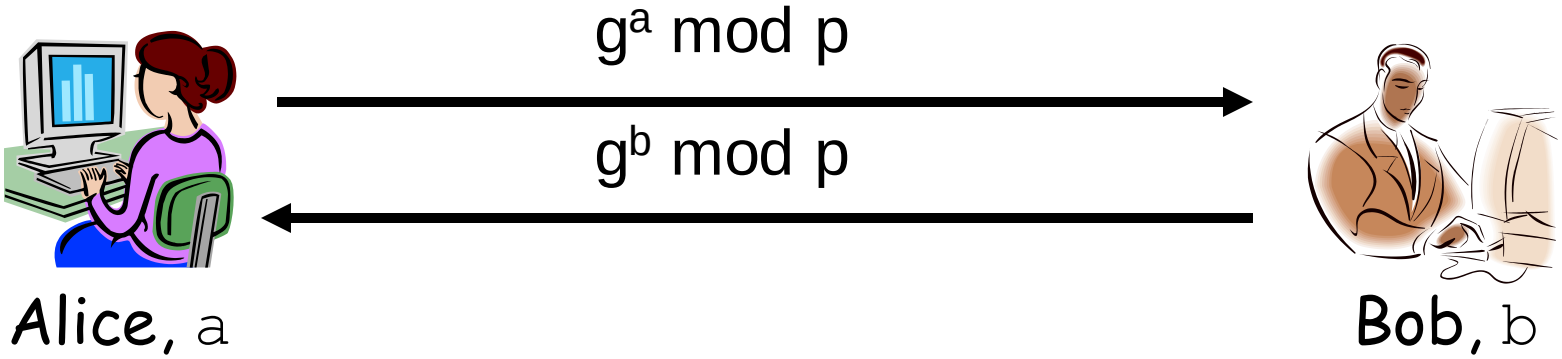
Poznámka:

$$g^{a}g^{b} \bmod p = g^{a+b} \bmod p \neq g^{ab} \bmod p$$

- Systém je prolomen, pokud útočník zjistí hodnotu a nebo b
- Systém je také možné prolomit vyřešením diskrétního logaritmu

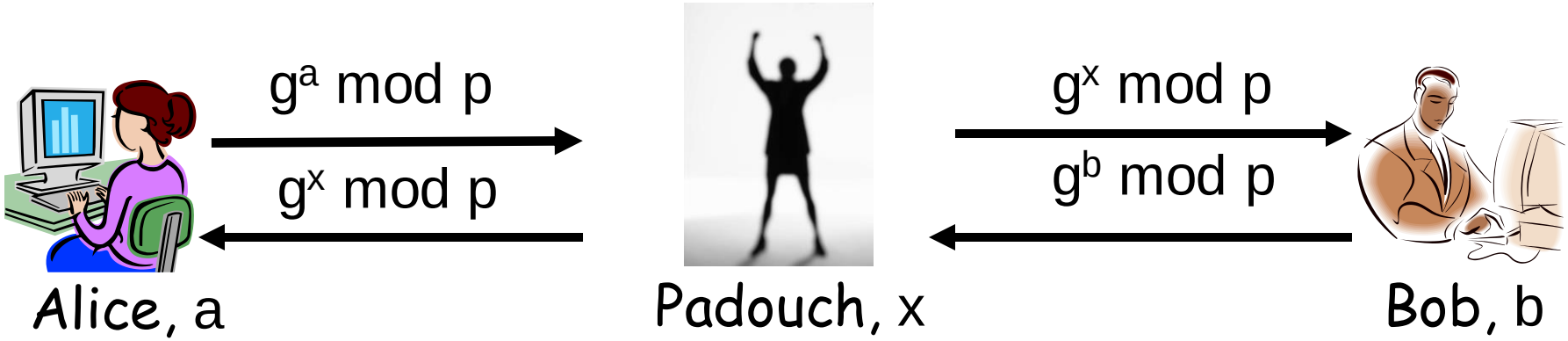
Diffie-Helmannův protokol

- **Veřejný klíč:** g, p
- **Tajný klíč:** pro Alici a, pro Boba b



- Alice spočítá $(g^b)^a = g^{ba}$ mod p = g^{ab} mod p
- Bob spočítá $(g^a)^b = g^{ab}$ mod p
- K = g^{ab} mod p je možné použít jako klíč pro symetrickou šifru
- potencionální útočník může zachytit pouze g,p, g^a g^b

DH protokol a útok Man-in-the-middle



- Padouch sdílí tajemství „g^{ax} mod p“ s Alicí
- Padouch sdílí tajemství „g^{bx} mod p“ s Bobem
- Alice ani Bob nemají ani potuchy, že nekomunikují přímo ale prostřednictvím Padoucha

DH protokol a útok Man-in-the-middle

Jak zabránit útoku MiM ?

- Šifrovat DH výměnu pomocí sdíleného tajemství
- Šifrovat DH výměnu pomocí veřejného klíče
- Podepsat DH výměnu pomocí soukromého klíče
- Jinak ?

V případě použití protokolu DH je nutné počítat s možností MitM útok.

Dotazy