Információs rendszerek biztonságtechnikája

Kriptográfia
Nyilvános kulcsú algoritmusok

Vakulya Gergely

A nyilvános kulcsú titkosítás elve

- A szimmetrikus kulcsú titkosítással szemben külön titkosító és kititkosító kulcs.
- A két kulcs együtt kerül létrehozásra.
- A két kulcs közül az egyiket nyilvánosságra hozom (nyilvános kulcs, public key).
- A másikat titkosan kezelem (titkos kulcs, private key).
- Ha valaki nekem szeretne üzenetet küldeni, az üzenetet az **én nyilvános kulcsommal titkosítja**.
- Ezt az üzenetet **csak én tudom megfejteni** az én **titkos kulcsommal**.
- A gyors működés jellemzően nem elsődleges követelmény.

Az RSA titkosítás

- Ron Rivest, Adi Shamir és Len Adleman, 1976
- Az algoritmus arra épül, hogy a prímszámok szorzatára való felbontásra (faktorizációra) nem ismert gyors algoritmus.
- Az algoritmus vázlata:
  - Választunk két különböző, nagy (többszáz jegyű), **véletlenszerű** prímszámot (p, q).
  - Kiszámítjuk a prímek szorzatát ($N = p \cdot q$). Fontos: N-ből p-t és q-t nagyon nehéz megkapni.
  - Kiszámítjuk az N Euler-függvényét: $\varphi(N) = (p-1) \cdot (q-1)$.
  - Választunk egy $\varphi(N)$-hez relatív prím e számot. (Például egy prímszám megfelel, szokásos értékek: 3, 65537.)
  - Kiszámítjuk d-t, e modulo $\varphi(N)$ inverzét, tehát egy olyan számot, amire: $e \cdot d \equiv 1 \mod \varphi(N)$.
  - Nyilvános kulcs: (N, e), titkos kulcs: d
  - Kódolás: $c = p^e \mod N$
  - Dekódolás: $p = c^d \mod N$

A prímszámok előállítása

- Az RSA algoritmus alapja a nagy méretű, véletlenszerű prímszámok előállítása.
- A kiindulási alap: véletlenszámok előállítása. Kiválasztás: Prímtesztek.
- A legtöbb prímteszt nem garantál prímszámokat, de ennek valószínűsége tetszőlegesen nagy lehet.
- Fontos, hogy a prímszámokat ne használjuk fel újra.
- Fontos a véletlenszámok minősége.
- CVE-2008-0166
- CVE-2008-0166 összefoglaló

Két matematikai kérdés az RSA-val kapcsolatban

Prímfaktorizáció

- Az RSA biztonsága a prímfaktorizációban rejlik. Ha a kulcsgeneráláskor kapott szorzatot faktorizálni tudjuk, kiszámíthatjuk a titkos kulcsot.
- Ez a faktorizáció általános esetben NP nehézségű, de valószínűleg nem NP-teljes.
- NP = Non-Polinomial? Nem!
- NP = Nondeterministic Polinomial? Igen!
- A kvantumszámítógépek komoly fejtörést okozhatnak.

Modulo-n hatványozás

- A kódolás és dekódolás (modulo n) hatványozást használ.
- Ez ismételt szorzással valósítható meg.
- A végrehajtási időkből következtetni lehet a kulcsra.

Nyilvános kulcsok továbbítása

- Az RSA nagy bizonságot garantál.
- Viszont számításigényes.
- A blokk kódolók bitekkel dolgoznak, rajtuk logikai műveleteket hajtanak végre.
- Az RSA számokon hajt végre matematikai műveleteket (modulo n hatványozást).
- Minden üzenetet egyetlen számmá kell transzformálni az RSA-val való titkosításhoz.
- Így az RSA célszerűtlen hosszabb üzenetek továbbítására.
- Megoldás: RSA-val csak egy blokk kódolóhoz használatos kulcsot küldjük át, maga a kommunikáció már a (sokkal gyorsabb) blokk kódolóval történik.

A Diffie-Hellman kulcscsere

- Szimmetrikus kulcsú titkosító algoritmus kulcsai más módszerrel is megoszthatóak a partnerek között.
- A Diffie-Hellman kulcscsere lényege, hogy nem egy, az egyik partnernek már birtokában levő kulcsot küld el a másik partnernek
- Ehelyett **közösen hoznak létre** egy **megosztott titkot** (shared secret).
- Harmadik fél az összes üzenet lehallgatásával sem tudja a megosztott titkot létrehozni.
- A konkrét megvalósításokban nem maga a megosztott titok a kulcs, hanem abból valamilyen hash algoritmussal van származtatva.

D-H alalógia

- Analógia:
  - Alice elküld egy aktatáskát Bobnak, amit egy lakattal zár le. Ehhez a lakathoz csak Alice-nak van kulcsa.
  - Bob nem tudja kinyitni a táskát, mert azon lakat van, viszont ő is le tudja zárni. Bob tehát szintén tesz a táskára egy lakatot, amihez csak neki van kulcsa, majd visszaküldi Alice-nak a duplán lelakatolt táskát.
  - Alice kinyitja és leveszi a saját lakatját, majd visszaküldi a táskát Bobnak.
  - Bob kinyitja és leveszi a lakatot, így ki tudja nyitni a táskát.
- A Diffie-Helmann kucscsere alapja az RSA-hoz hasonlóan egy olyan két olyan művelet, amik kioltják egymást.

A digitális aláírás

- Az RSA titkosításnál a kódolás és a dekódolás művelete gyakorlatilag azonos, csak a kulcs más.
- A titkos kulccsal is lehet kódolni és a nyilvános kulccsal is lehet dekódolni.
- Mi értelme úgy kódolni valamit, hogy bárki visszafejtheti? Ezzel tudja bizonyítani valaki, hogy ő küldte az üzenetet.
- A digitális aláírás lényege:
  - Generálunk egy kulcspárt. A publikus kulcsot mindenki rendelkezésére bocsájtjuk.
  - Az aláírandó üzenetből hash-t képezünk.
  - A hash-t a **titkos** kulcsunkkal titkosítjuk. Ennek eredménye lesz a digitális aláírás.
  - A címzett a hash-t kititkosítja a mi **nyilvános** kulcsunkkal, majd összeveti a vett üzenet hash-ével.

Digitális tanúsítványok

- Az üzeneteket digitális aláírással láthatjuk el. De mi van, ha valaki más is készít egy kulcspárt a mi nevünkben és ezt aláírásra használja?
- Erre jelent megoldást a digitális tanúsítvány (certificate).
- A tanúsítványt a tanúsító hatóság (certificate authority, CA) állítja ki.
- A tanúsítvány részei:
  - A tanúsítvány tulajdonosának adatai
  - A tulajdonos publikus kulcsa
  - Érvényességi idő
  - A tanúsítvány kiállítója
  - A kiállító digitális aláírása
- A CA-ban mindkét résztvevő megbízik.
- A tanúsítvány (ami a nyilvános kulcs hitelességét bizonyítja) a CA nyilvános kulcsával ellenőrizhető.

A bizalmi lánc (chain of trust)

- Egyetlen CA csak korlátozott számú tanúsítványt képes kiadni.
- Megoldás: Bizalmi hierarchia (fa struktúrával).
- Az egyes CA-k saját tanúsítványai a felettük levő szinten levő CA-ra hivatkozhatnak.
- A hierarchia tetején a root CA áll. Root CA-ből több is van, többféle bizalmi lánc is létezhet párhuzamosan.
- A nyilvános kulcsú tanúsítványokat az X.509 szabvány definiálja.

Nyílt kulcsú titkosítási szabványok (Public-Key Cryptography Standards, PKCS)

Bizonyos célra ajánlott titkosítási szabványok gyűjteménye

- PKCS #1: RSA
- PKCS #3: Diffie-Helman kulcscsere
- PKCS #5: Jelszó alapú titkosítás (PBKDF2)
- PKCS #7: Digitális aláírások és tanúsítványok
- PKCS #8: Titkos kulcsok tárolása
- PKCS #10: Tanúsítványkérelmek
- PKCS #12: Kriptográfiai fájlok tárolása