Public Key Cryptosystems

Guevara Noubir
http://www.cs.nec.edu/home/noubir/Courses/CIS223/FP
Reading: Chapter 5 upto section 5.7

Outline

- Concepts behind public key crypto
  - Some number theory
  - RSA cryptosystem
  - Primality testing
  - Factoring numbers and other attacks

Encryption Models

Message source: Plaintext → Encryption Algorithm → Ciphertext → Decryption Algorithm → Plaintext → Message Destination

Symmetric encryption:

Asymmetric encryption:
- Early 70’s
- Published in 76
- Cannot provide unconditional securities

CIS223 Classical Cryptography 3
Applications

- Symmetric algorithms vs. asymmetric algorithms (public-key crypto systems)
  - About 1000 times faster!
  - However, require a shared key!
- Practice:
  - Use public key crypto to establish a shared key
  - Examples
    - Choose a key for the symmetric algorithm $K$ encrypt it with the public key of the destination
    - Use the key $K$ to encrypt the message and integrity protect it
  - IPSec (IKE): establish a session key (using either public-key cryptosystem or shared secrets)
  - IPSec uses the session key to provide confidentiality and integrity

Number Theory

- $\mathbb{Z}_n^*$: abelian group of numbers $< n$, relatively prime to $n$
- Euclidean Algorithm $(a, b)$:
  - Computes the $\text{gcd}(a, b)$
- Extended Euclidean Algorithm $(a, b)$:
  - Computes $r, s, t$, s.t. $sa + tb = r = \text{gcd}(a, b)$
  - If $r = 1 \Rightarrow s = a \mod b$
  - If $r = 1 \Rightarrow t$?
- Time complexity less than $O(\ell^2)$ if $a$ and $b$ are encoded in less than $\ell$ bits.

Chinese Remainder Theorem

- Assume that $m_1, \ldots, m_n$ are pairwise relatively prime positive integers
- Chinese Remainder Theorem (CRT):
  - Suppose $a_1, \ldots, a_n$ are integers s.t.
    - $x = a_1 \mod m_1$
    - $x = a_2 \mod m_2$
    - $\ldots$
    - $x = a_n \mod m_n$
  - There exists a unique $x \mod m_1 m_2 \ldots m_n$ that satisfies all previous equations
    - $x = \sum a_i M_i y_i \mod M$  \hspace{1cm} $M_i = M / m_i$, $y_i = M_i^{-1}$
Other Known Results

- If $G$ is a multiplicative group of order $n$ then the order of any element of $G$ divides $n$
  - Order of $\mathbb{Z}_n^*$ = $\phi(n)$
  
- If $b \in \mathbb{Z}_n^*$, then $b^{\phi(n)} = 1 \pmod{n}$
  - How about when $n$ is prime?
  
- If $p$ is prime then $\mathbb{Z}_p^*$ is a cyclic group

RSA Cryptosystem

- Due to Rivest-Shamir-Adleman in 1977
  - Let $n = pq$, where $p$ and $q$ are primes
  - $G = \mathbb{Z}_n^*$
  - $K = \{(k, p, q, a, b) : ab = 1 \pmod{\phi(n)}\}$
  
  Encryption:
  - $c_2(x) = x^e \pmod{n}$
  - Description:
    - $e_2(y) = y^e \pmod{n}$
  
  Public key: $n$ and $b$
  - Private key: $p$, $q$, $a$

Example

- $p = 101; q = 113 \Rightarrow n = 11413$
  - $\phi(n) = 11200 = 2^5 \cdot 5^2$
  
- Let $b = 3533 \Rightarrow b^d = 6597$
  - How is $b$ chosen?
  
  Encrypt plaintext: 9726
  - Ciphertext: $9726^{11413} \pmod{11413} = 5761$
  - Decryption ciphertext: 5761
    - Plaintext: $5761^{11413} \pmod{11413} = 9726$
Use of RSA

- Encryption (A wants to send a message M to B):
  - A uses the public key of B and encrypts M (i.e., \( e_B(M) \))
  - Since only B has the private key, only B can decrypt M (i.e., \( M = d_B(e_B(M)) \))

- Digital signature (A wants to send a signed message to B):
  - Based on the fact that \( e_B(d_B(M)) = d_B(e_B(M)) \)
  - A encrypts M using its private key (i.e., \( d_B(M) \)) and sends it to B
  - B can check that \( e_B(d_B(M)) = M \)
  - Since only A has the decryption key, only him can generate this message.

Security of RSA

- Security of RSA is based on the belief that:
  - \( x^e \mod n \) is a one-way function

- The trapdoor is the knowledge of the factorization of \( n \) into \( pq \)

- Conjecture:
  - RSA is as difficult as factoring numbers

RSA Implementation

- RSA Parameters Generation
  - Generate two large primes: \( p \) and \( q \)
  - \( n = pq \) and \( \phi(n) = (p-1)(q-1) \)
  - Choose a random \( e \in \{1, 2, \ldots, \phi(n) - 1\} \) s.t. \( \gcd(\phi(n), e) = 1 \)
  - \( d \equiv e^{-1} \mod \phi(n) \)
  - Public key is \((n, e)\) and private key is \((n, d)\)

- \( p \) and \( q \) should be at least 512 bits long each
  - \( n \) is at least 1024 bits long

- Computation Complexity:
  - Exponentiation cost: SQUARE-AND-Multiply
  - Modular inverse: Extended Euclidean Alg.
  - Modular Multiplication:
Prime Numbers Generation

- Density of primes (prime number theorem):
  - $\pi(x) \sim x/\ln(x)$
  - E.g., a random number of 512 bits has probability: $1/\ln(512) = 1/355$ to be prime
- Sieve of Eratosthenes
  - Try if any number less than $\sqrt{n}$ divides $n$
- Fermat’s Little Theorem does not detect Carmichael numbers
  - $b^{n-1} \equiv 1 \mod n$
  - E.g., 561 is the smallest Carmichael number
- Solovay-Strassen primality test
  - If $n$ is not prime at least 50% of $b$ fail to satisfy the following:
    - $\text{Jacobi symbol can be computed in less than } O(\log n^2)$
    - Jacobi symbol is a non-degenerate elliptic Lenstra symbol.

Computing Jacobi Symbol

- Definition: $\left(\frac{a}{n}\right) = \left(\frac{a}{p_1^{e_1}}\right) \cdot \left(\frac{a}{p_2^{e_2}}\right) \cdots \left(\frac{a}{p_k^{e_k}}\right)$
- No need to factor $n$ to compute the Jacobi symbol
- Use the following rules [$n$ is positive odd]:
  - $a \equiv b \mod n$ $\Rightarrow \left(\frac{a}{n}\right) = \left(\frac{b}{n}\right)$
  - $\left(\frac{2}{n}\right) = 1$ if $n \equiv \pm 1 \mod 8$
  - $\left(\frac{2}{n}\right) = -1$ if $n \equiv \pm 3 \mod 8$
  - $\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right) \cdot \left(\frac{a}{p_2}\right) \cdots \left(\frac{a}{p_k}\right)$
  - $\left(\frac{a}{n}\right) = \left(\frac{a}{p}\right)^{\frac{\phi(p)}{2}}$ if $a \equiv \pm 1 \mod p$ and $p = \pm 4 \mod 8$

Rabin-Miller primality test

- If $n$ is not prime then it is not pseudoprime to at least 75% of random $a < n$
  - $a \equiv 1 \mod n$
  - $a \equiv n - 1 \mod n$
  - If $a^k \equiv 1 \mod n$ then return(n prime)
  - For $i = 0$ to $d - 1$
    - If $a^k \equiv -1 \mod n$ then return(n prime)
    - Else $b = a^k$
    - return(n composite)
- Probabilistic test, deterministic if the Generalized Riemann Hypothesis is true
- Deterministic polynomial time primality test [Agrawal, Kayal, Saxena 2002]
Attacks on RSA

- Factoring
  - Many factoring algorithms were proposed: quadratic sieve, elliptic curve factoring, number field sieve, Pollard’s rho-method
  - Capable of factoring a 512 bits modulus ~ 155 digits in 1999 using 8400 MIPS-years

- Other attacks:
  - Computing $\phi(n)$
  - Decryption exponent: if $e$ is known
    - Las Vegas algorithm (3.10) that will factor $n$ with probability $\frac{1}{2}$

- Semantic Security

Rabin Cryptosystem

- Motivation:
  - The difficulty of factoring does not necessarily prove RSA security
  - Hardness of factoring leads to security proof of Rabin’s cryptosystem against chosen-plaintext attack

- Scheme:
  - $n = pq$ ($p$ and $q$ are two primes and $p \equiv q \equiv 3 \mod 4$)
  - $P = C \in \mathbb{Z}_n^*$; $K = ((n, p, q), e)$
  - $e(x) = x^e \mod n$
  - $d(x) = y^e \mod n$

- Note:
  - Conditions: $p \equiv q \equiv 3 \mod 4$ and $\mathbb{Z}_n^*$ is for simplification of decryption and security proof purpose

Rabin Cryptosystem

- Observation:
  - Is the encryption function injective?
    - Solution?
  - How can we decrypt?
    - Solution: CRT
      - Consider $x \equiv y^{(p-1)/4} \mod p$
        - $x \equiv y^{(p-1)/4} \mod q$
      - When can we use this technique of decoding?
    - Example:
      - $n = 7 \times 11$
      - Decrypt $y = 23$
Security of Rabin Cryptosystem

- If Rabin cryptosystem can be broken then we can build a Las Vegas probabilistic algorithm with success probability 1/2
  - Rabin Oracle Factoring(n)
    - External RabinDecryp
      - Choose a random r;
      - Let y← r^e mod n
      - x← RabinDecryp(y);
      - If x = ar mod n return(failure)
      - Else return(p= gcd(x+r, n) ; q=n/p);
    - Conclusion:
      - Rabin cryptosystem is secure against a chosen plaintext attack
      - Additional security results:
        - Rabin cryptosystem is insecure against a chosen ciphertext attack