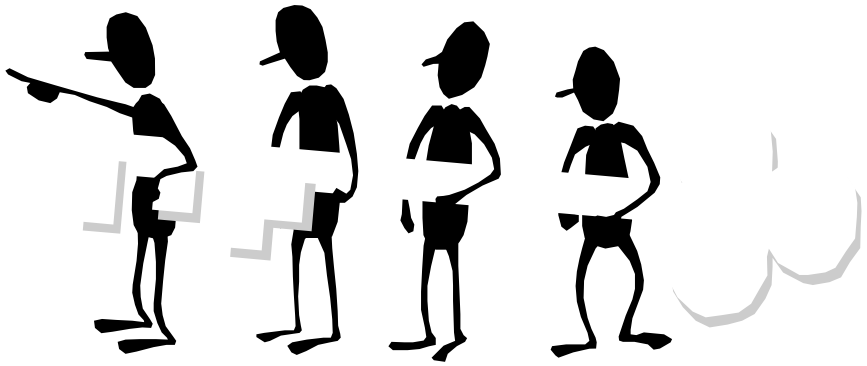
Public Key Cryptography and RSA



Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

Audio/Video recordings of this lecture are available at:

http://www.cse.wustl.edu/~jain/cse571-14/

1. Public Key Encryption
2. Symmetric vs. Public-Key
3. RSA Public Key Encryption
4. RSA Key Construction
5. Optimizing Private Key Operations
6. RSA Security

These slides are based partly on Lawrie Brown's slides supplied with William Stallings's book "Cryptography and Network Security: Principles and Practice," 6th Ed, 2013.

Public Key Encryption

- Invented in 1975 by Diffie and Hellman at Stanford
- Encrypted_Message = Encrypt(Key1, Message)
- Message = Decrypt(Key2, Encrypted_Message)

- Keys are interchangeable

- One key is made public while the other is kept private
- Sender knows only public key of the receiver
- Asymmetric

Ref: http://en.wikipedia.org/wiki/Public-key_cryptography

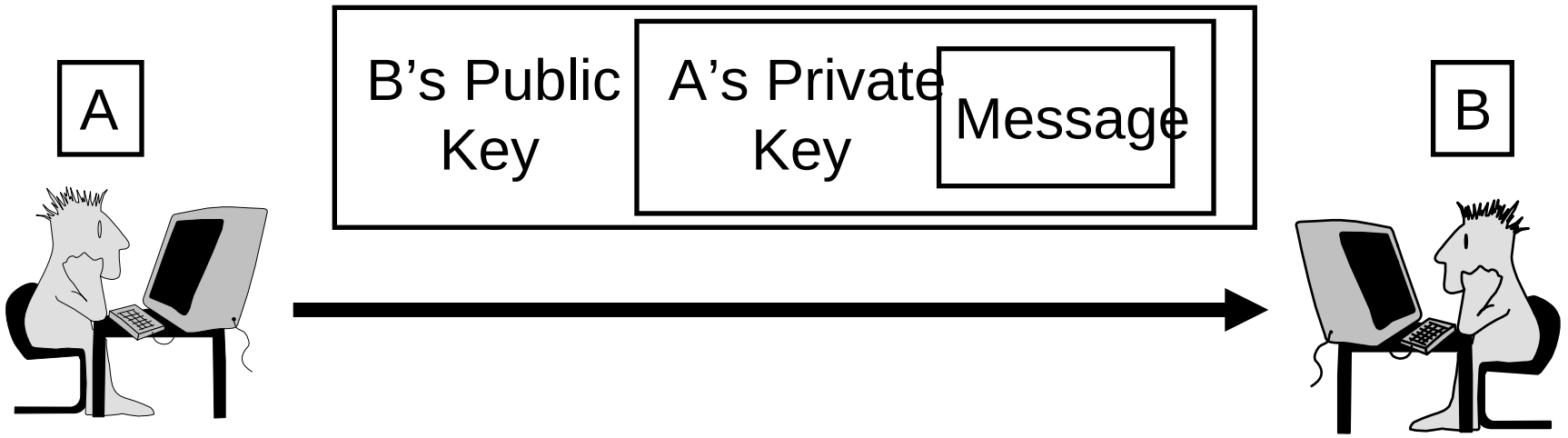
Public Key Encryption Example

- Rivest, Shamir, and Adleman at MIT
- RSA: Encrypted_Message = $m^3 \bmod 187$
- Message = Encrypted_Message$^{107} \bmod 187$
- Key1 = <3,187>, Key2 = <107,187>
- Message = 5
- Encrypted Message = $5^3 = 125$
- Message = $125^{107} \bmod 187 = 5$
  $= 125^{(64+32+8+2+1)} \bmod 187$
  $= \{(125^{64} \bmod 187)(125^{32} \bmod 187)...$
  $(125^{2} \bmod 187)(125 \bmod 187)\} \bmod 187$

Ref: http://en.wikipedia.org/wiki/RSA

Symmetric vs. Public-Key

Public-Key Authentication and Secrecy



- A encrypts the message with its private key and then with B's public key
- B can decrypt it with its private key and A's public key
- No one else can decrypt Ÿ Secrecy
- No one else can send such a message
  - Ÿ B is assured that the message was sent by A
  - Ÿ Authentication

Public-Key Applications

- 3 Categories:
  - Encryption/decryption (provide secrecy)
  - Digital signatures (provide authentication)
  - Key exchange (of session keys)
- Some algorithms are suitable for all uses, others are specific to one

Public-Key Requirements

- Need a trapdoor one-way function
- One-way function has
  - Y = f(X) easy
  - X = f^{-1}(Y) infeasible
- A trap-door one-way function has
  - Y = f_k(X) easy, if k and X are known
  - X = f_k^{-1}(Y) easy, if k and Y are known
  - X = f_k^{-1}(Y) infeasible, if Y known but k not known
- A practical public-key scheme depends on a suitable trap-door one-way function

Security of Public Key Schemes

- Like private key schemes brute force exhaustive search attack is always theoretically possible
- But keys used are too large (>512bits)
- Security relies on large enough difference in difficulty between easy (en/decrypt) and hard (cryptanalyse) problems
- More generally the hard problem is known, but is made hard enough to be impractical to break
- Requires the use of very large numbers
- Hence is slow compared to private key schemes

RSA Public Key Encryption

- Ron Rivest, Adi Shamir, and Len Adleman at MIT 1978
- Exponentiation in a Galois field over integers modulo a prime
  - Exponentiation takes $O((\log n)^3)$ operations (easy)
- Security due to cost of factoring large numbers
  - Factorization takes $O(e^{\log n \log \log n})$ operations (hard)
- Plain text M and ciphertext C are integers between 0 and n-1.
- Key 1 = {e, n},
  Key 2 = {d, n}
- $C = M^e \bmod n$
  $M = C^d \bmod n$
- How to construct keys:
  - Select two large primes: p, q, $p \neq q$
  - $n = p \times q$
  - Calculate Euler's Totient Fn: $\phi(n) = (p-1)(q-1)$
  - Select e relatively prime to $\phi(n)$: $\gcd(\phi(n), e) = 1; 0 < e < \phi(n)$
  - Calculate d = inverse of e mod $\phi(n)$: $de \bmod \phi(n) = 1$
  - Euler's Theorem: $x^{ed} = x^{k\phi(n)+1} = x \bmod n$