THE UNIVERSITY OF BRITISH COLUMBIA

Public Key Cryptography

EECE 412

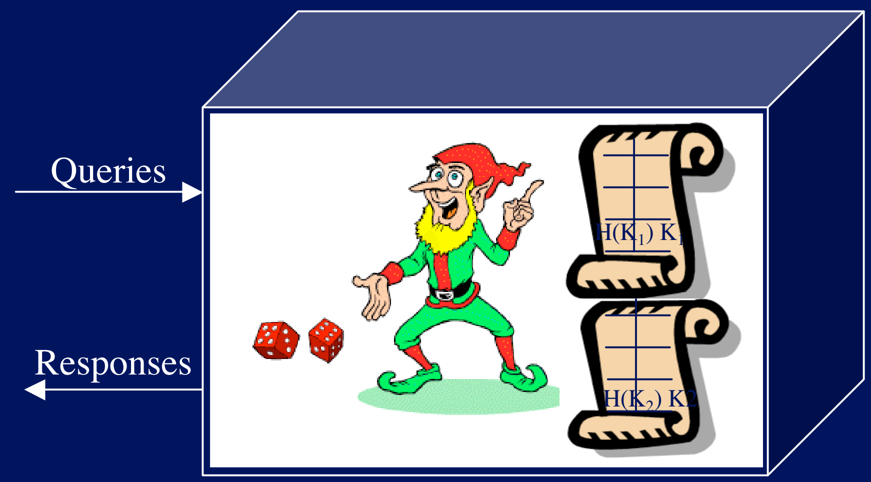
Session Outline

- The Random Oracle model for Public Key Cryptosystems
  - Public key encryption and trapdoor one-way permutations
  - Digital signatures
- RSA
- Uses of Public Crypto
- The order of sign and encrypt

Public Key Encryption and Trap-door One-Way Permutation

as Random Oracle

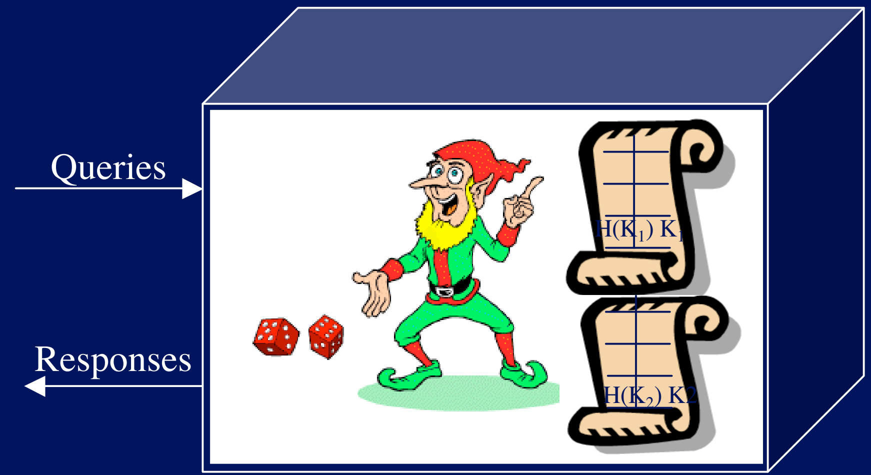
- Public Key Encryption Scheme:
  - Key pair (KR, KR^{-1}) generation function from random string R
    - KR → KR^{-1} is infeasible
  - $C = \{M\}_{KR}$
  - $M = \{C\}_{KR^{-1}}$



- In:
  - fixed size short string (plaintext) M,
  - Key KR
- Out: fixed size short string (ciphertext) C

Digital Signature as Random Oracle

- Public Key Signature Scheme:
  - Key pair (σR, VR) generation function
    - VR → σR is infeasible
  - S = Sig $_{\sigma R}$(M)
  - {True, False} = Ver_{VR}(S)



	Signing	Verifying
Input	Any string M + σR	S + VR
Output	S = hash(M) | cipher block	“True” or “False”

Looking Under the Hood

RSA

RSA

- Invented by Cocks (GCHQ), independently, by Rivest, Shamir and Adleman (MIT)
- Let p and q be two large prime numbers
- Let N = pq be the **modulus**
- Choose e relatively prime to (p-1)(q-1)
- Find d s.t. ed = 1 mod (p-1)(q-1)
- **Public key** is (N,e)
- **Private key** is d

RSA

- To encrypt message M compute
  - $C = M^e \bmod N$
- To decrypt C compute
  - $M = C^d \bmod N$
- Recall that e and N are public
- If attacker can factor N, he can use e to easily find d since $ed = 1 \bmod (p-1)(q-1)$
- Factoring the modulus breaks RSA
- It is not known whether factoring is the only way to break RSA

Simple RSA Example

- Example of RSA
  - Select "large" primes p = 11, q = 3
  - Then N = pq = 33 and (p−1)(q−1) = 20
  - Choose e = 3 (relatively prime to 20)
  - Find d such that ed = 1 mod 20, we find that d = 7 works
- **Public key:** (N, e) = (33, 3)
- **Private key:** d = 7

Simple RSA Example

- **Public key:** $(N, e) = (33, 3)$
- **Private key:** $d = 7$
- Suppose message $M = 8$
- Ciphertext C is computed as

  $C = M^e \bmod N = 8^3 = 512 = 17 \bmod 33$
- Decrypt C to recover the message M by

  $M = C^d \bmod N = 17^7 = 410{,}338{,}673$

  $= 12{,}434{,}505 * 33 + 8 = 8 \bmod 33$

Uses for Public Key Crypto

Uses for Public Key Crypto

- Confidentiality
  - Transmitting data over insecure channel
  - Secure storage on insecure media
- Authentication
- Digital signature provides integrity and **non-repudiation**
  - No non-repudiation with symmetric keys

Non-non-repudiation

- Alice orders 100 shares of stock from Bob
- Alice computes **MAC** using symmetric key
- Stock drops, Alice claims she did not order
- Can Bob prove that Alice placed the order?
- **No!** Since Bob also knows symmetric key, he could have forged message
- **Problem:** Bob knows Alice placed the order, but he can't prove it

Non-repudiation

- Alice orders 100 shares of stock from Bob
- Alice **signs** order with her private key
- Stock drops, Alice claims she did not order
- Can Bob prove that Alice placed the order?
- **Yes!** Only someone with Alice’s private key could have signed the order
- This assumes Alice’s private key is not stolen (revocation problem)

Sign and Encrypt
vs
Encrypt and Sign

Public Key Notation

- **Sign** message M with Alice's **private key:** $[M]_{Alice}$
- **Encrypt** message M with Alice's **public key:** $\{M\}_{Alice}$
- Then

$$\{[M]_{Alice}\}_{Alice} = M$$

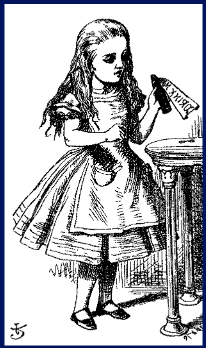
$$[\{M\}_{Alice}]_{Alice} = M$$

Confidentiality and Non-repudiation

- Suppose that we want confidentiality and non-repudiation
- Can public key crypto achieve both?
- Alice sends message to Bob
  - **Sign and encrypt** $\{[M]_{Alice}\}_{Bob}$
  - **Encrypt and sign** $[\{M\}_{Bob}]_{Alice}$
- Can the order possibly matter?

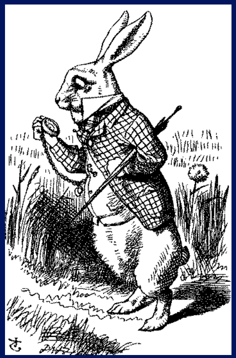
Sign and Encrypt

- M = “I love you”



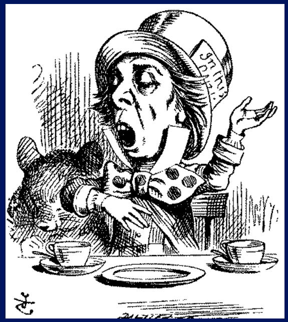
Alice

$\{[M]_{Alice}\}_{Bob}$ →



Bob

$\{[M]_{Alice}\}_{Charlie}$ →



Charlie

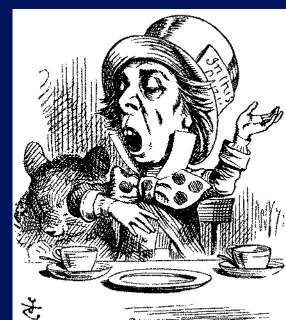
- **Q:** What is the problem?
- **A:** Charlie misunderstands crypto!

Encrypt and Sign

- M = "My theory, which is mine, is this: …."

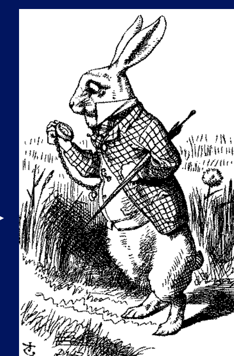
Alice

$[\{M\}_{Bob}]_{Alice}$ →



Charlie

$[\{M\}_{Bob}]_{Charlie}$ →



Bob

- **Note** that Charlie cannot decrypt M
- **Q:** What is the problem?
- **A:** Bob misunderstands crypto!