

# 4-8 RSA

Hengfeng Wei

hfwei@nju.edu.cn

May 06, 2019





这是一段关于“坚持”与“灵感”的旅程 …

Programming  
Techniques

S.L. Graham, R.L. Rivest\*  
Editors

---

# A Method for Obtaining Digital Signatures and Public- Key Cryptosystems

R. L. Rivest, A. Shamir, and L. Adleman  
MIT Laboratory for Computer Science  
and Department of Mathematics

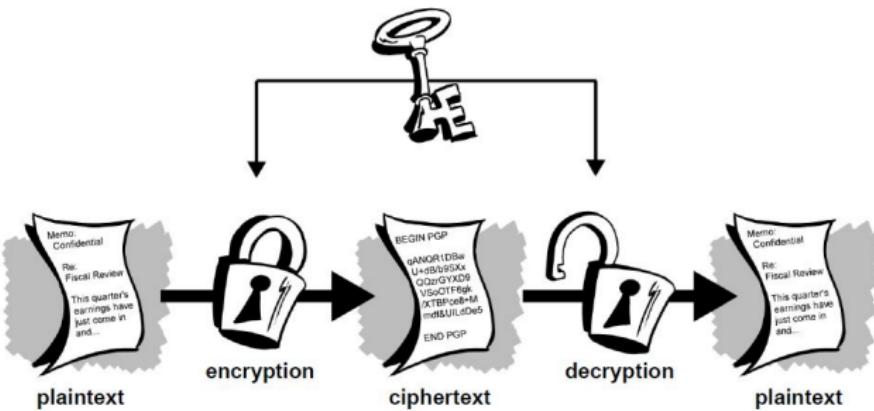
---

CACM'1978

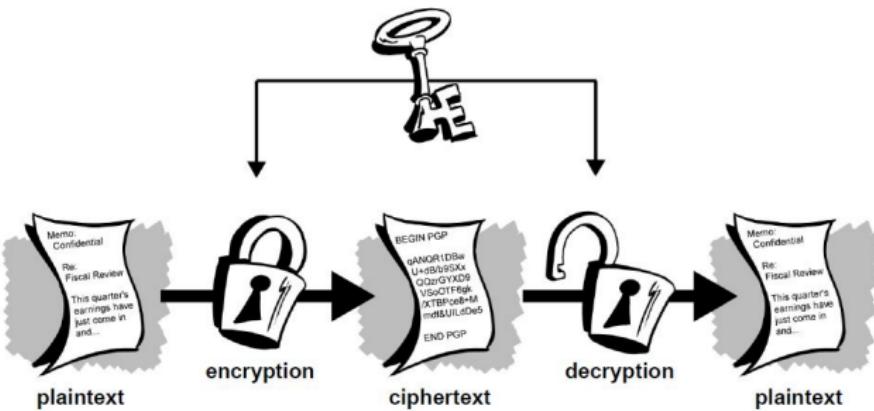


# Symmetric-key Encryption

# Symmetric-key Encryption

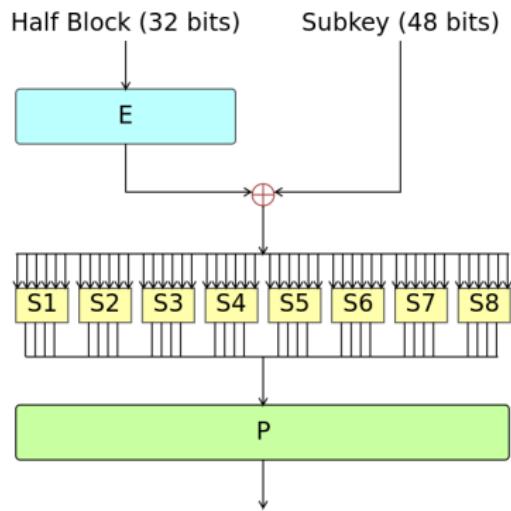


# Symmetric-key Encryption

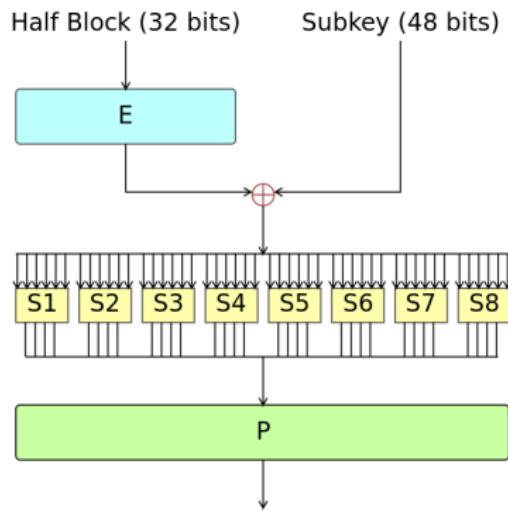


**Q : How to share this KEY?**

# Data Encryption Standard (DES; 1976)



# Data Encryption Standard (DES; 1976)



Keys Sent by People

*Q* : How to share this KEY?

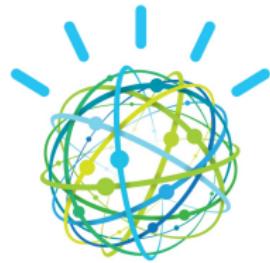
*Q : How to share this KEY?*



*Q : How to share this KEY?*

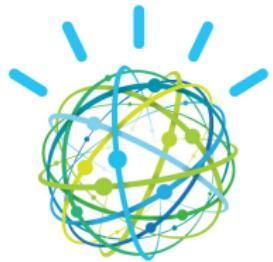


Whitfield Diffie (1944 ~)



**IBM Watson™**

Whitfield Diffie@IBM  
Watson'1974



IBM Watson™

Whitfield Diffie@IBM  
Watson'1974



Martin Hellman (1945 ~)



Martin Hellman, Whitfield Diffie



Martin Hellman, Whitfield Diffie



+ Ralph Merkle

两千多年来的密码学“公理”：

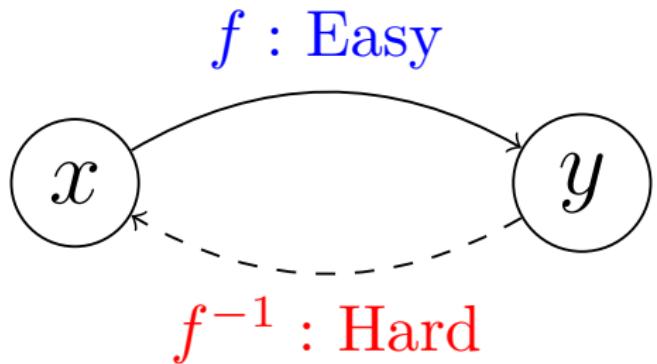
“不管采用什么方法，密钥就是一定得发送，  
这个动作无论如何避免不了”



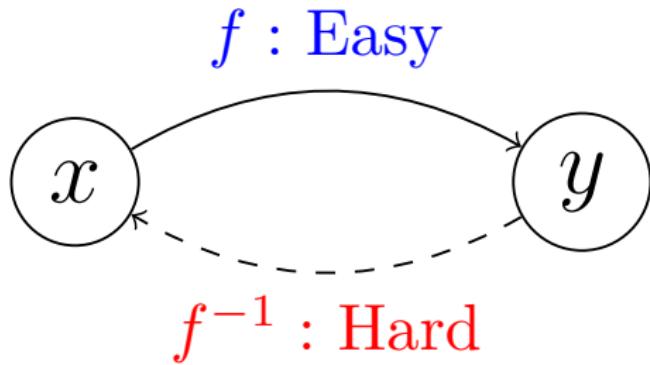
+ Ralph Merkle

Martin Hellman, Whitfield Diffie

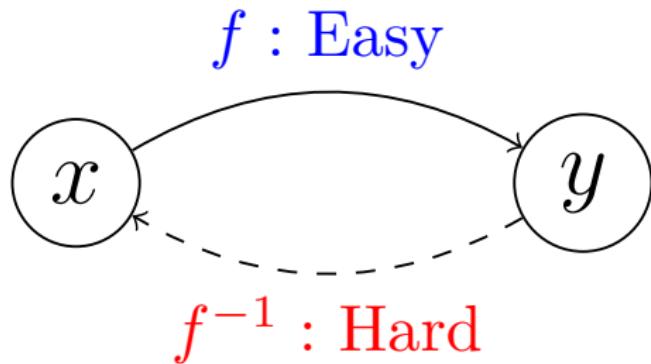
## Definition (One-way Function)



## Definition (One-way Function)

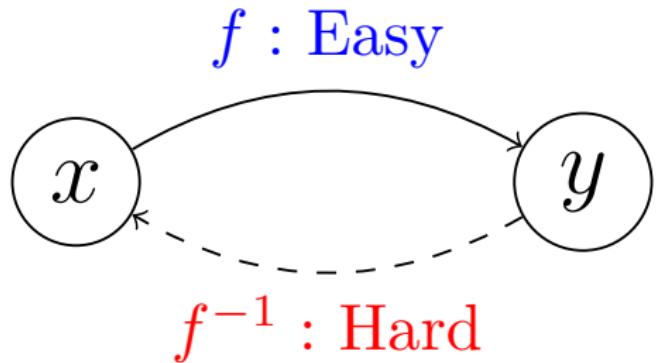


## Definition (One-way Function)



Q : Hard in *worst case* or in *average case*?

## Definition (One-way Function)



Q : Hard in *worst case* or in *average case*?

Q : Do one-way functions exist?

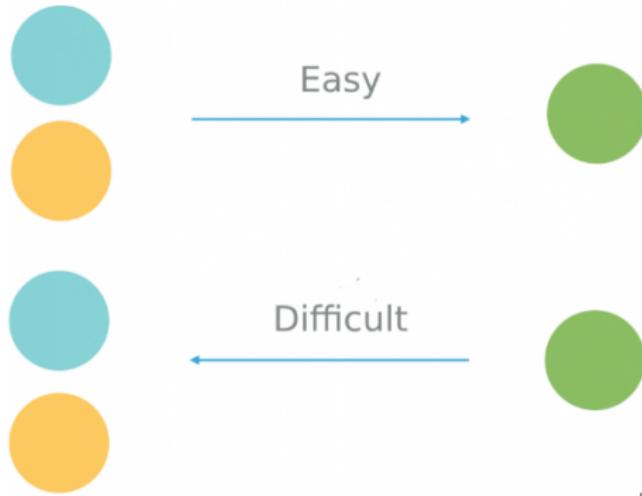


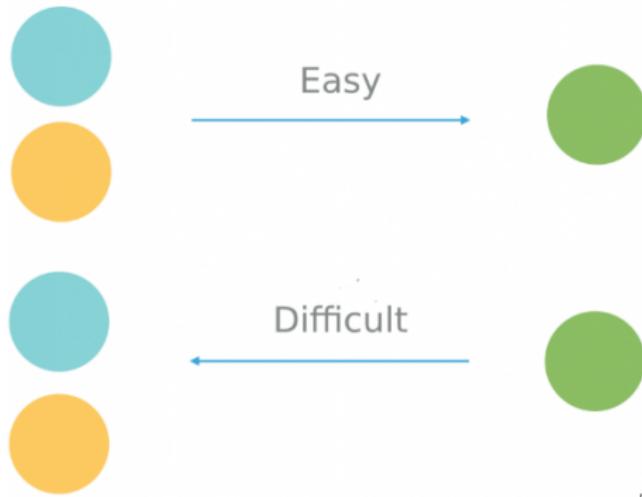
[One-way Function \(wiki\)](#)

**Unsolved problem in computer science:**

**? Do one-way functions exist?**

(more unsolved problems in computer science)





*Q : How to share a COLOR?*

# Diffie-Hellman-Merkle Key Exchange

(Martin Hellman; Spring, 1976)

# Diffie-Hellman-Merkle Key Exchange

(Martin Hellman; Spring, 1976)

Definition (Discrete Logarithm)

$$g^{\textcolor{red}{x}} \equiv a \pmod{p}$$

$p$        $g$  (generator for  $\mathbb{Z}_p$ )

$p$        $g$  (generator for  $\mathbb{Z}_p$ )

Alice

1. Randomly choose  $a$
2. Compute  $A = g^a \bmod p$
3. Send  $A$  to Bob
4. Compute  $K = B^a \bmod p$

Bob

1. Randomly choose  $b$
2. Compute  $B = g^b \bmod p$
3. Send  $B$  to Alice
4. Compute  $K = A^b \bmod p$

$p$        $g$  (generator for  $\mathbb{Z}_p$ )

Alice

1. Randomly choose  $a$
2. Compute  $A = g^a \bmod p$
3. Send  $A$  to Bob
4. Compute  $K = B^a \bmod p$

Bob

1. Randomly choose  $b$
2. Compute  $B = g^b \bmod p$
3. Send  $B$  to Alice
4. Compute  $K = A^b \bmod p$

$$K = g^{ab} \bmod p$$

$$p \quad g \text{ (generator for } \mathbb{Z}_p\text{)}$$

Alice

1. Randomly choose  $a$
2. Compute  $A = g^a \bmod p$
3. Send  $A$  to Bob
4. Compute  $K = B^a \bmod p$

Bob

1. Randomly choose  $b$
2. Compute  $B = g^b \bmod p$
3. Send  $B$  to Alice
4. Compute  $K = A^b \bmod p$

$$K = g^{ab} \bmod p$$

$K$  used for DES



## Man-in-the-Middle Attack



Alice and Bob need to be online simultaneously.

两千多年来的密码学“公理”：

“不管采用什么方法，密钥就是一定得发送，  
这个动作无论如何避免不了”

两千多年来的密码学“公理”：

“不管采用什么方法，密钥就是一定得发送，  
这个动作无论如何避免不了”

Proof.





Whitfield Diffie



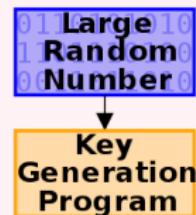
Whitfield Diffie

“我是个没用的科学家，一辈子将一事无成”

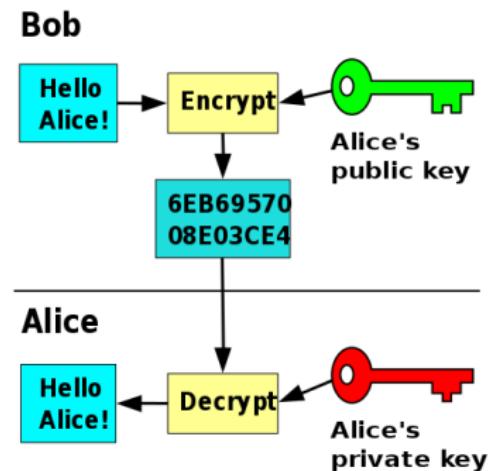
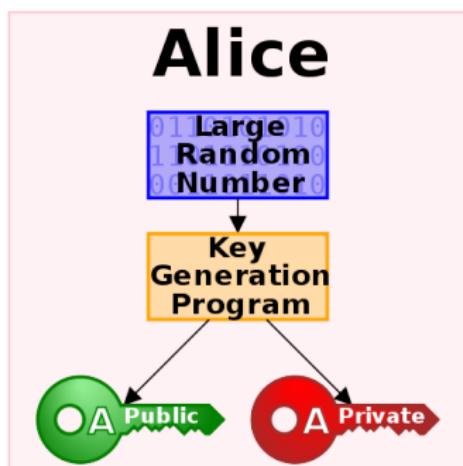
# Asymmetric Cryptography (Public-Key Cryptography) (Whitfield Diffie; 1975)

# Asymmetric Cryptography (Public-Key Cryptography) (Whitfield Diffie; 1975)

## Alice



# Asymmetric Cryptography (Public-Key Cryptography) (Whitfield Diffie; 1975)

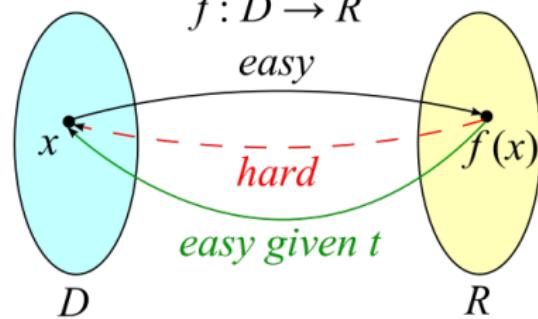


## Definition (Trapdoor (陷门) One-Way Functions)

## Definition (Trapdoor (陷门) One-Way Functions)

$$(f, t) = \mathbf{Gen}(1^n)$$

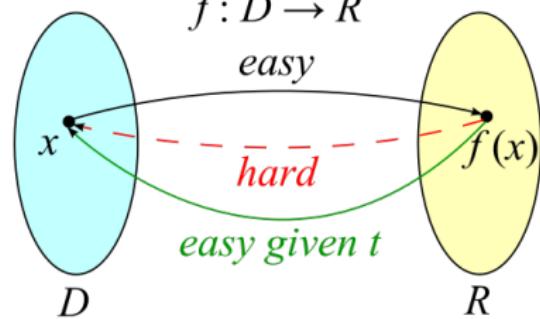
$$f: D \rightarrow R$$



## Definition (Trapdoor (陷门) One-Way Functions)

$$(f, t) = \mathbf{Gen}(1^n)$$

$$f: D \rightarrow R$$



$$g^x \equiv a \pmod{p}$$

*Q* : How to send (encrypt and decrypt) a COLOR?

# New Directions in Cryptography

*Invited Paper*

WHITFIELD DIFFIE AND MARTIN E. HELLMAN, MEMBER, IEEE

**Abstract**—Two kinds of contemporary developments in cryptography are examined. Widening applications of teleprocessing have given rise to a need for new types of cryptographic systems, which minimize the need for secure key distribution channels and supply the equivalent of a written signature. This paper suggests ways to solve these currently open problems. It also discusses how the theories of communication and computation are beginning to provide the tools to solve cryptographic problems of long standing.

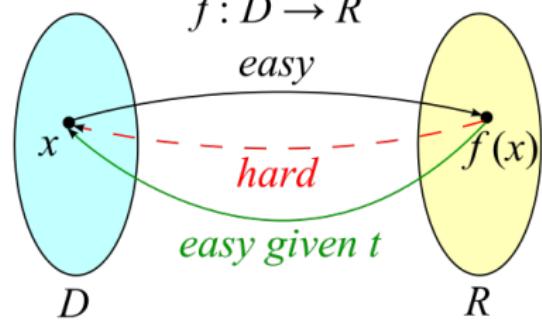
The best known cryptographic privacy: preventing the unauthorized eavesdropping on communications over an open channel. In order to use cryptography to insure privacy, it is necessary for the communicating parties to share a key which is known to no one else. One way to do this is by mailing the key in advance over some secure courier or registered mail. Another way is to use a public key system, where each user has a public key which can be freely distributed, and a private key which is known only to that user. The public key is used for encryption, and the private key is used for decryption. This allows anyone to send a message to the user, but only the user can read it. The user's public key is also used to verify the user's identity, so that the user can be sure that the message came from the intended recipient.

Whitfield Diffie, Martin Hellman (1976)

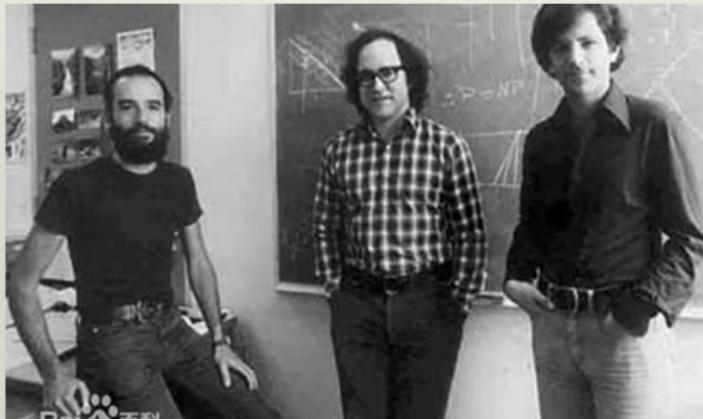
## Task: Search for Trapdoor One-way Functions

$$(f, t) = \mathbf{Gen}(1^n)$$

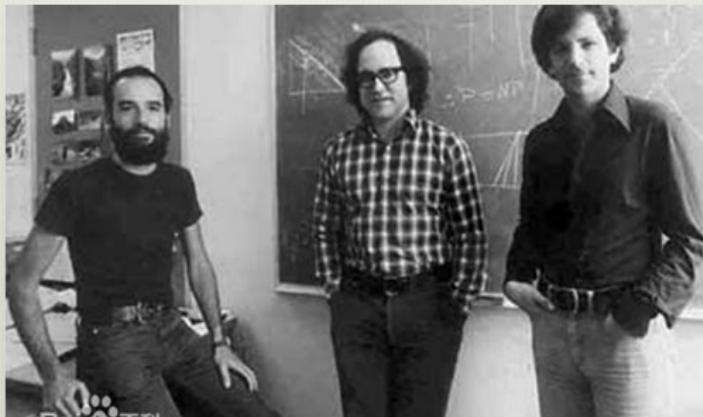
$$f: D \rightarrow R$$



# Ron Rivest, Adi Shamir and Leonard Adleman



# Ron Rivest, Adi Shamir and Leonard Adleman



“也许、大概、或许并不存在这样的  
*Trapdoor One-Way Functions*”

# RSA

(Ron Rivest; April 1977)

# RSA

(Ron Rivest; April 1977)



# RSA

(Ron Rivest; April 1977)



Programming  
Techniques

S.L. Graham, R.L. Rivest\*  
Editors

---

## A Method for Obtaining Digital Signatures and Public- Key Cryptosystems

R. L. Rivest, A. Shamir, and L. Adleman  
MIT Laboratory for Computer Science  
and Department of Mathematics

---

$p$ ,  $q$

$p$ ,  $q$

$n = pq$

$p$ ,  $q$

$\textcolor{blue}{n} = pq$

$$\phi(\textcolor{red}{n}) = (p - 1)(q - 1)$$

$p, q$

$n = pq$

$$\phi(n) = (p-1)(q-1)$$

$$e : (e, \phi(n)) = 1$$

$p, q$

$$\textcolor{blue}{n} = pq$$

$$\phi(\textcolor{red}{n}) = (p-1)(q-1)$$

$$\textcolor{blue}{e} : (e, \phi(n)) = 1$$

$$\textcolor{red}{d} : ed \equiv 1 \pmod{\phi(n)}$$

$p, \quad q$

$\textcolor{blue}{n} = pq$

$$\phi(\textcolor{red}{n}) = (p-1)(q-1)$$

$$\textcolor{blue}{e} : (e, \phi(n)) = 1$$

$$\textcolor{red}{d} : ed \equiv 1 \pmod{\phi(n)}$$

$$P = (e, n)$$

$p, q$

$\textcolor{blue}{n} = pq$

$$\phi(\textcolor{red}{n}) = (p-1)(q-1)$$

$$\textcolor{blue}{e} : (e, \phi(n)) = 1$$

$$\textcolor{red}{d} : ed \equiv 1 \pmod{\phi(n)}$$

$$P = (e, n)$$

$$S = (d, n)$$

$$p, \quad q$$

$$\textcolor{blue}{n} = pq$$

$$\phi(\textcolor{red}{n}) = (p-1)(q-1)$$

$$\textcolor{blue}{e} : (e, \phi(n)) = 1$$

$$\textcolor{red}{d} : ed \equiv 1 \pmod{\phi(n)}$$

$$P = (e, n)$$

$$S = (d, n)$$

$$P(M) = M^e \pmod{n}$$

$$p, \quad q$$

$$\textcolor{blue}{n} = pq$$

$$\phi(\textcolor{red}{n}) = (p-1)(q-1)$$

$$\textcolor{blue}{e} : (e, \phi(n)) = 1$$

$$\textcolor{red}{d} : ed \equiv 1 \pmod{\phi(n)}$$

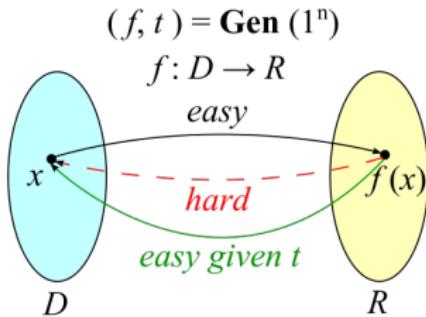
$$P = (e, n)$$

$$S = (d, n)$$

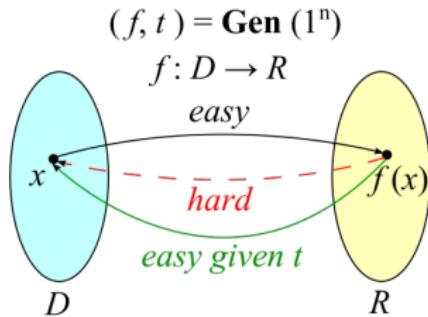
$$P(M) = M^e \pmod{n}$$

$$S(C) = C^d \pmod{n}$$

*Q* : What is the trapdoor one-way function in RSA?

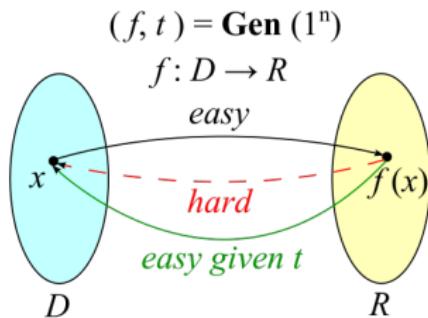


*Q* : What is the trapdoor one-way function in RSA?



$$f(x) = x^e \bmod n$$

*Q* : What is the trapdoor one-way function in RSA?



$$f(x) = x^e \bmod n$$

*d*

## RSA-number : $n = pq$

number	decimal digits	prize	factored (references)
RSA-100	100		Apr. 1991
RSA-110	110		Apr. 1992
RSA-120	120		Jun. 1993
RSA-129	129		Apr. 1994 (Leutwyler 1994, Cipra 1995)
RSA-130	130		Apr. 10, 1996
RSA-140	140		Feb. 2, 1999 (te Riele 1999a)
RSA-150	150		Apr. 6, 2004 (Aoki 2004)
RSA-155	155		Aug. 22, 1999 (te Riele 1999b, Peterson 1999)
RSA-160	160		Apr. 1, 2003 (Bahr et al. 2003)
RSA-200	200		May 9, 2005 (see Weisstein 2005a)
RSA-576	174	\$10000	Dec. 3, 2003 (Franke 2003; see Weisstein 2003)
RSA-640	193	\$20000	Nov. 4, 2005 (see Weisstein 2005b)
RSA-704	212	withdrawn	Jul. 1, 2012 (Bai et al. 2012, Bai 2012)
RSA-768	232	withdrawn	Dec. 12, 2009 (Kleinjung 2010, Kleinjung et al. 2010)
RSA-896	270	withdrawn	
RSA-1024	309	withdrawn	
RSA-1536	463	withdrawn	
RSA-2048	617	withdrawn	

## “Small $e$ Attack” (CLRS 31.7-2)

$$e = 3$$

$$0 < d < \phi(n)$$

## “Small $e$ Attack” (CLRS 31.7-2)

$$e = 3$$

$$0 < d < \phi(n)$$

$$n = pq$$

## “Small $e$ Attack” (CLRS 31.7-2)

$$e = 3$$

$$0 < d < \phi(n)$$

$$n = pq$$

Why do I factor  $n$  if I have obtained  $d$ ?

## “Small $e$ Attack” (CLRS 31.7-2)

$$e = 3$$

$$0 < d < \phi(n)$$

$$n = pq$$

Why do I factor  $n$  if I have obtained  $d$ ?



Common  $n$ ; Different  $e$ 's and  $d$ 's

$$ed \equiv 1 \pmod{\phi(n)}$$

$$ed \equiv 1 \pmod{\phi(n)}$$

$$ed = 1 + k\phi(n) \quad (k \in \mathbb{Z})$$

$$ed \equiv 1 \pmod{\phi(n)}$$

$$ed = 1 + k\phi(n) \quad (k \in \mathbb{Z})$$

$$ed = 1 + k\phi(n) \quad (\textcolor{red}{k} \in \mathbb{N}, k < \min \{e, d\})$$

$$ed \equiv 1 \pmod{\phi(n)}$$

$$ed = 1 + k\phi(n) \quad (k \in \mathbb{Z})$$

$$ed = 1 + k\phi(n) \quad (\textcolor{red}{k} \in \mathbb{N}, k < \min \{e, d\})$$

$$3d = 1 + k\phi(n) \quad (\textcolor{red}{k} \in \{1, 2\})$$

$$ed \equiv 1 \pmod{\phi(n)}$$

$$ed = 1 + k\phi(n) \quad (k \in \mathbb{Z})$$

$$\boxed{ed = 1 + k\phi(n) \quad (\textcolor{red}{k} \in \mathbb{N}, k < \min \{e, d\})}$$

$$3d = 1 + k\phi(n) \quad (\textcolor{red}{k} \in \{1, 2\})$$

$$\phi(n) = (p - 1)(q - 1) = n - (p + q) + 1$$

$$n = pq$$

# Common Modulus Attack



Common  $n$ ; Different  $e$ 's and  $d$ 's

# Common Modulus Attack



Common  $n$ ; Different  $e$ 's and  $d$ 's

$$(e_i, e_j) = 1$$

# Common Modulus Attack



Common  $n$ ; Different  $e$ 's and  $d$ 's

$$(e_i, e_j) = 1$$

$$E = M^{e_1} \pmod{n} \quad F = M^{e_2} \pmod{n}$$

## Common Modulus Attack



Common  $n$ ; Different  $e$ 's and  $d$ 's

$$(e_i, e_j) = 1$$

$$E = M^{e_1} \pmod{n} \quad F = M^{e_2} \pmod{n}$$

$$(e_1, e_2) = 1$$

## Common Modulus Attack



Common  $n$ ; Different  $e$ 's and  $d$ 's

$$(e_i, e_j) = 1$$

$$E = M^{e_1} \pmod{n} \quad F = M^{e_2} \pmod{n}$$

$$(e_1, e_2) = 1 \implies e_1 x + e_2 y = 1$$

## Common Modulus Attack



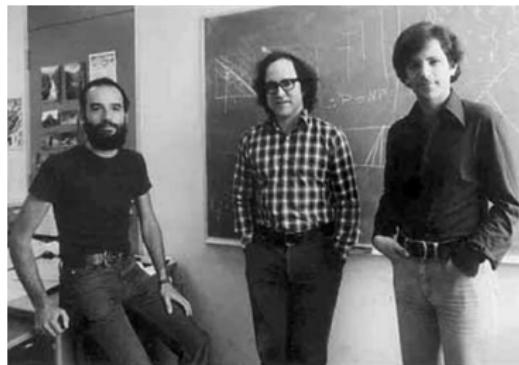
Common  $n$ ; Different  $e$ 's and  $d$ 's

$$(e_i, e_j) = 1$$

$$E = M^{e_1} \pmod{n} \quad F = M^{e_2} \pmod{n}$$

$$(e_1, e_2) = 1 \implies e_1 x + e_2 y = 1$$

$$M = E^x \cdot F^y \pmod{n}$$



*“For their ingenious contribution for  
making public-key cryptography useful in practice.”*

— ACM Turing Award, 2002



*“For fundamental contributions to modern cryptography.”*

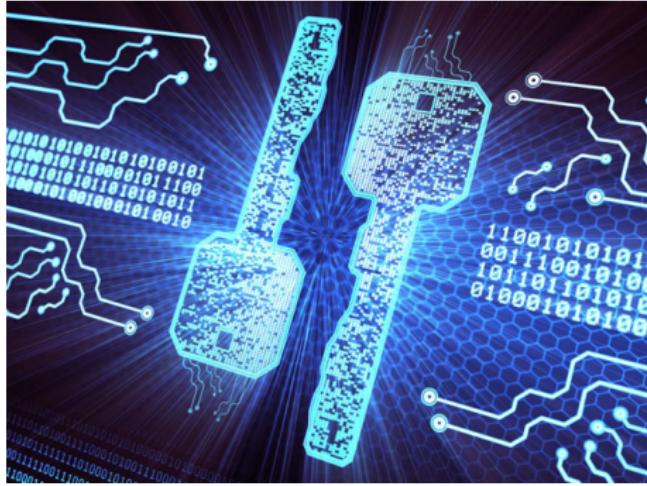
*— ACM Turing Award, 2015*



I don't know' has become 'I  
don't know yet'.

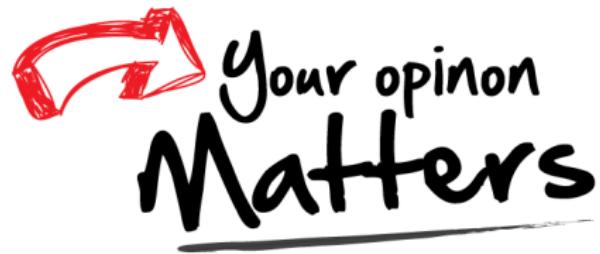
*Bill Gates*

[www.thequotes.in](http://www.thequotes.in)



## Will Quantum Computers Break Encryption?





Office 302

Mailbox: H016

hfwei@nju.edu.cn