บทที่ 4 : Cryptography & Steganography Part2

สธ412 ความมั่นคงของระบบสารสนเทศ

อาจารย์อภิพงศ์ ปิงยศ

apipong.ping@gmail.com

Agenda

- Public Key Cryptography
- มาตรฐานการเข้ารหัสข้อมูล
  - RSA
  - Diffie-Hellman
  - Digital Signature Algorithm (DSA)
  - Digital Certificate

Public Key Cryptography

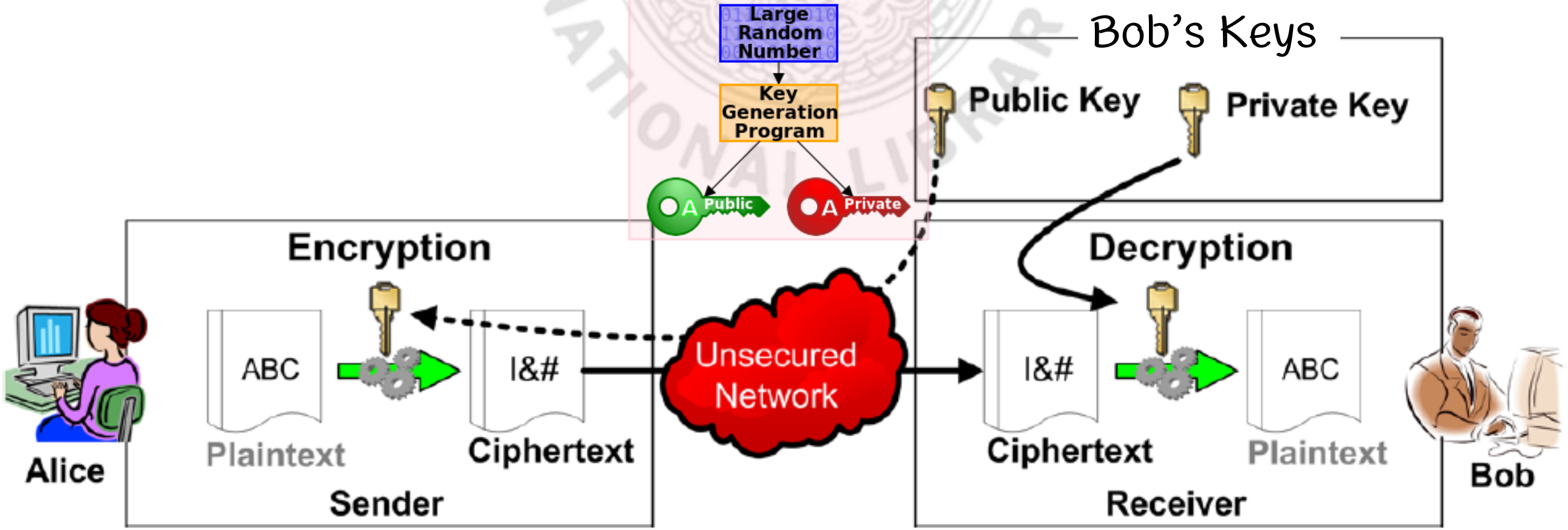
- ปัญหาของการเข้ารหัสแบบซีเคร็ทคีย์คือ การแลกเปลี่ยนคีย์ ที่ใช้ในเชิงปฏิบัติได้ยาก
- จึงมีการพัฒนาเทคนิคการแจกจ่ายคีย์อย่างปลอดภัย โดยใช้**คีย์ที่เข้ารหัสและถอดรหัสคนละคีย์**
- **เรียกว่า (Public/Private Key Cryptography)**

Public Key Cryptography

- Private Key เป็นคีย์ที่รู้เฉพาะเจ้าของ
- Public Key เป็นคีย์ที่ประกาศให้สาธารณะทราบ ใครจะนำไปใช้ก็ได้
- คีย์หนึ่งใช้เข้ารหัส อีกคีย์หนึ่งใช้ถอดรหัส
- ทั้งสองคีย์จะมีความสัมพันธ์กันทางคณิตศาสตร์
- ถูกอธิบายครั้งแรกในปี 1976 โดย มาร์ติน เฮลล์แมน (Martin Hellman)และไวท์ฟิลด์ ดิฟฟี (Whitefield Diffie) แห่งมหาวิทยาลัยสแตนฟอร์ด

Whitefield Diffie & Martin Hellman

Public Key Cryptosystem



Public Key Cryptography [3]

- Public Key สร้างกุญแจคู่โดยอาศัยทฤษฎี One-way Function ซึ่งเป็นฟังก์ชันทางคณิตศาสตร์ที่คำนวณค่าได้ง่าย แต่คำนวณในทางตรงกันข้ามได้ยากมาก

One-way function :

Multiplication vs. Factorization

What X, Y ???

144 * 1 => 144

72 * 2 => 144

48 * 3 => 144

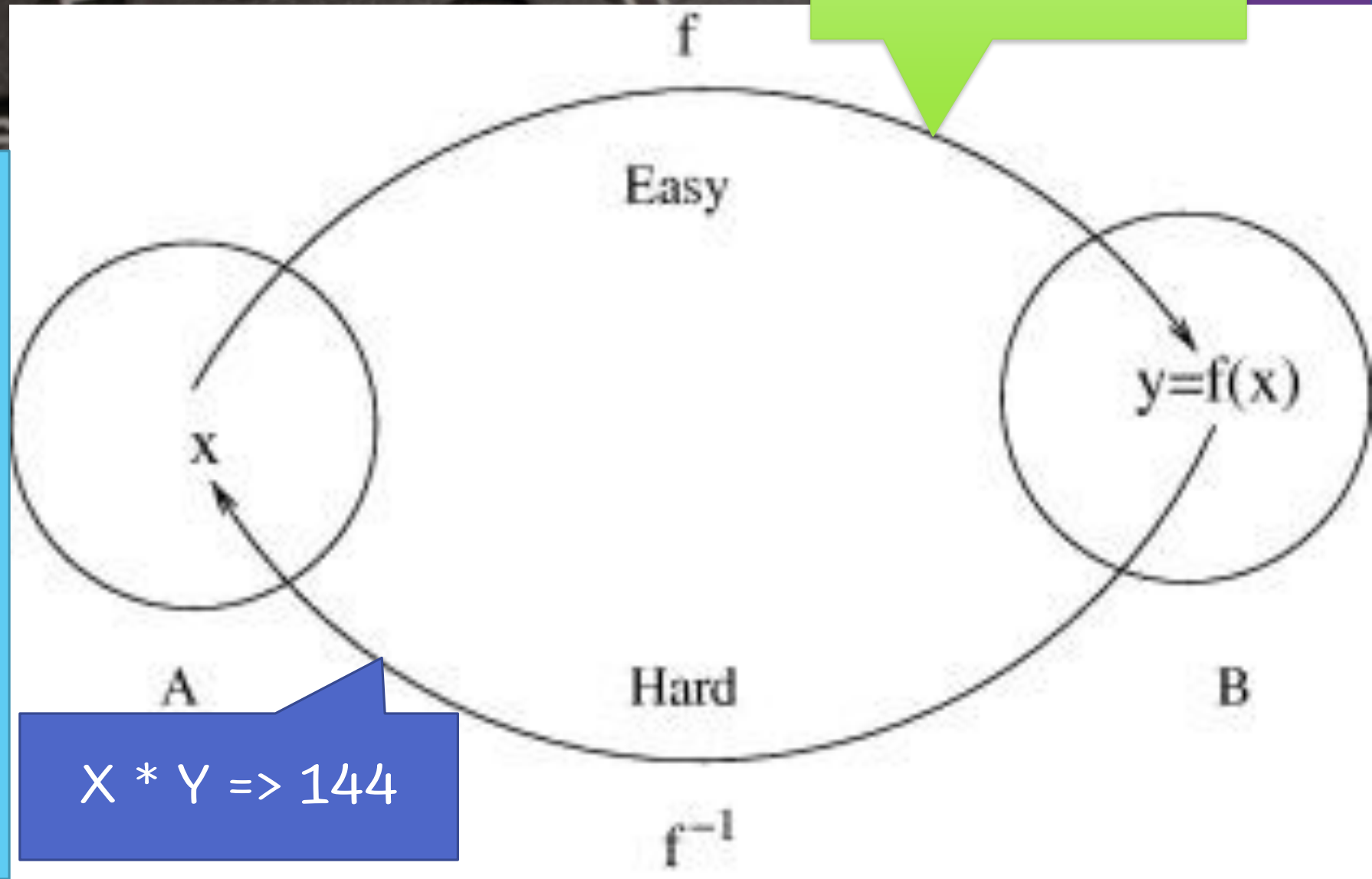
36 * 4 => 144

24 * 6 => 144

18 * 8 => 144

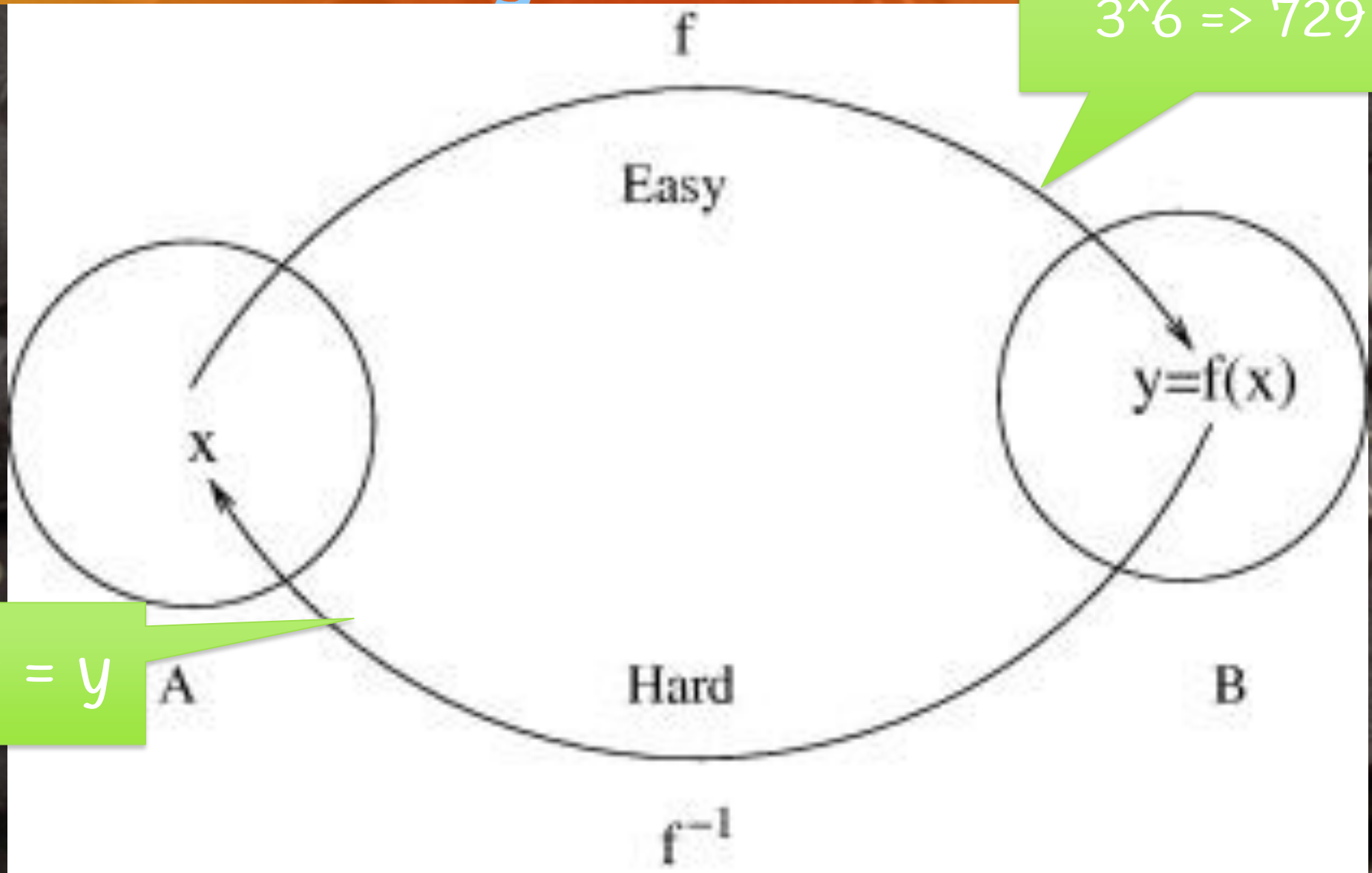
16 * 9 => 144

12 * 12 => 144



One-way function :

Exponentiation vs. Logarithms



3^6 => 729

$\log_x 729 = y$

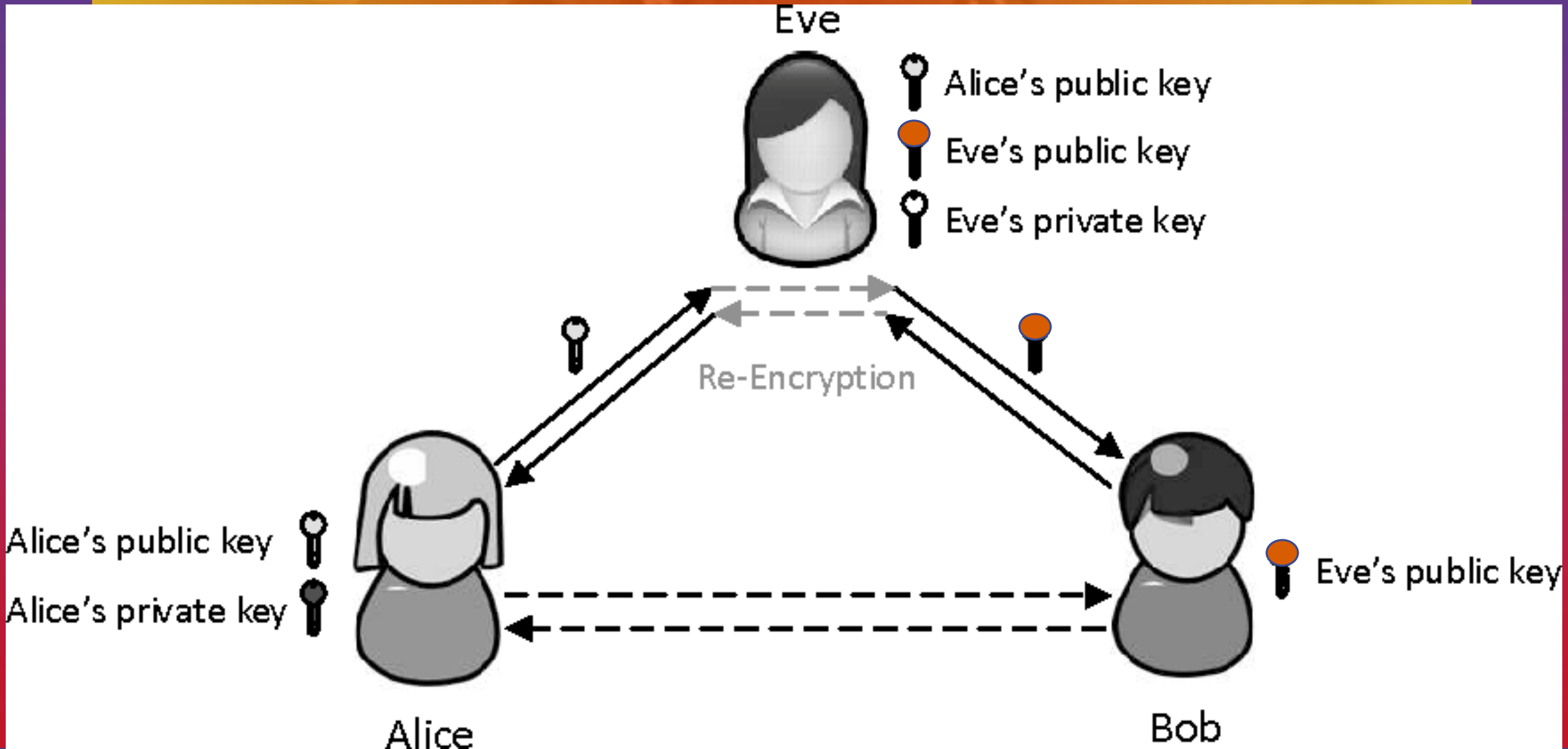
Asymmetric Key Cryptography

- การรู้คีย์ใดคีย์หนึ่งเป็นการยากที่จะคำนวณหาอีกคีย์หนึ่ง
- คีย์หนึ่งใช้สำหรับเข้ารหัสข้อมูล อีกคีย์หนึ่งใช้ถอดรหัสข้อมูล สามารถใช้สลับกันได้
- *ไม่สามารถใช้คีย์ใดคีย์หนึ่งเพียงคีย์เดียวในการเข้าและถอดรหัสได้*
- เรียกการเข้ารหัสแบบนี้ว่า **Asymmetric Key Cryptography (การเข้ารหัสแบบกุญแจอสมมาตร)**

ปัญหาของ Public Key Cryptography

- ถูกโจมตีได้ด้วยวิธี Man-in-the-Middle โดยสมบูรณ์ ซึ่ง Attacker สามารถดักจับข้อมูลที่รับส่งได้ทั้งหมด
- Attacker สามารถส่ง Public Key ของตัวเองไปให้กับเหยื่อทั้งสองฝั่งได้ แล้วใช้ Private Key ของตัวเองในการ Decryption
- แก้ไขโดยการใช้ Digital Signature และ Public Key Infrastructure (PKI)

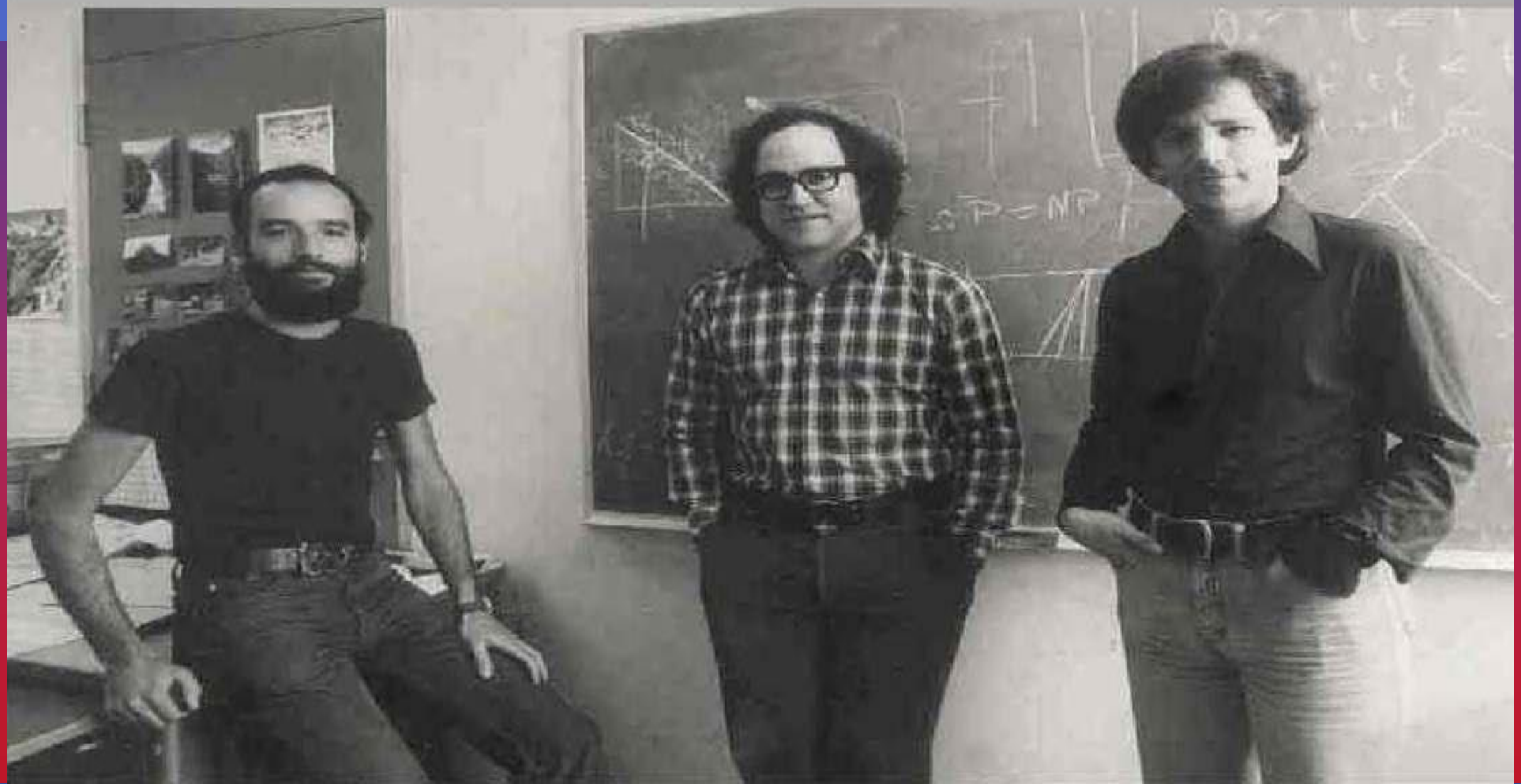
การโจมตีโดยวิธี MITM



มาตรฐานการเข้ารหัสข้อมูล : RSA

- เป็นการเข้ารหัสแบบ Public Key ที่ได้รับความนิยมอย่างสูง และถูกนำไปประยุกต์ใช้หลายแบบ
- RSA มาจากชื่อของผู้คิดอัลกอริทึมนี้ คือ Revest, Shamir และ Adlemen จาก MIT
- ข้อมูลที่เข้ารหัสด้วย Public Key จะถูกถอดรหัสได้โดยใช้ Private Key ที่เป็นคู่กันเท่านั้น

IN RSA WE TRUST



มาตรฐานการเข้ารหัสข้อมูล : RSA

- ขั้นตอนการเลือกพับลิกคีย์และไพรเวทคีย์
- 1) เลือกจำนวนเฉพาะ p และ q ซึ่งหากเลือกเลขจำนวนมากเท่าไรยิ่งถอดรหัสยากเท่านั้น แต่จะทำให้กระบวนการเข้าและถอดรหัสช้าลง
- 2) คำนวณ n = pq และ z = (p-1)(q-1)
- 3) เลือกจำนวน e ซึ่งมีค่าน้อยกว่า n และ e ต้องไม่มีตัวหารร่วมกับ z
- 4) คำนวณหา d โดยเมื่อ d คูณกับ e แล้วหารด้วย z เหลือเศษ 1 (ed mod z = 1)

มาตรฐานการเข้ารหัสข้อมูล: RSA

- 5) พับลิกคีย์คือ จำนวน (n, e) ส่วนไพรเวทคีย์คือ จำนวน (n, d)
- ขั้นตอนในการเข้ารหัสคือ สมมติเราต้องการส่งข้อมูล m โดย m<n สมมติเขียนรหัสด้วยพับลิกคีย์ (n, e) สูตรในการเข้ารหัสคือ $\mathbf{c = m^e \bmod n}$ โดย c คือ Ciphertext
- การถอดรหัสจะใช้ไพรเวทคีย์ (n, d) สูตรในการคำนวณคือ $\mathbf{m = c^d \bmod n}$ โดย m คือ plaintext ที่ถอดได้

ตัวอย่างการเข้ารหัสแบบ RSA

โดยเลือก p=5, q=7, n=35, z = 24, e = 5, d = 29

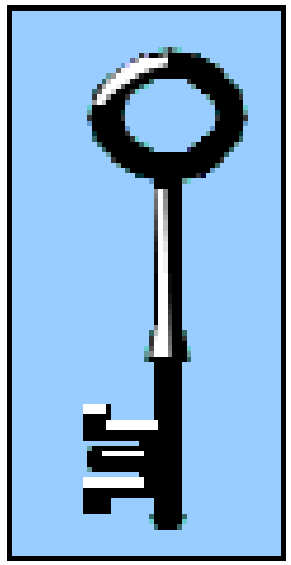
Plaintext	m	m^e	Ciphertext (m^e mod n)
l	12	248832	17
o	15	759375	15
v	22	5153632	22
e	5	3125	10

Ciphertext	c^d	$m = c^d \bmod n$	Plaintext
17	ค่าเยอะมาก	12	l
15	ค่าเยอะมาก	15	o
22	ค่าเยอะมาก	22	v
10	ค่าเยอะมาก	5	e

Session Key

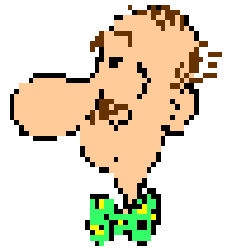
- โดยทั่วไป คีย์ที่ใช้บนอินเทอร์เน็ตจะมีขนาดอย่างน้อย 1,024 บิต ทำให้การเข้าและถอดรหัสช้ามากเมื่อเทียบกับการเข้ารหัสแบบ Symmetric Key
- ในทางปฏิบัติมักใช้ RSA คู่กับ AES โดยใช้ RSA สำหรับแจกจ่าย Secret Key ของ AES เพื่อเข้ารหัส
- Secret Key ที่ถูกเข้ารหัสด้วย Public Key จะเรียกว่า Session Key

Session Key Exchange



Randomly generated symmetric key

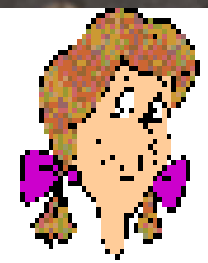
Alice's public key



Encrypted symmetric key

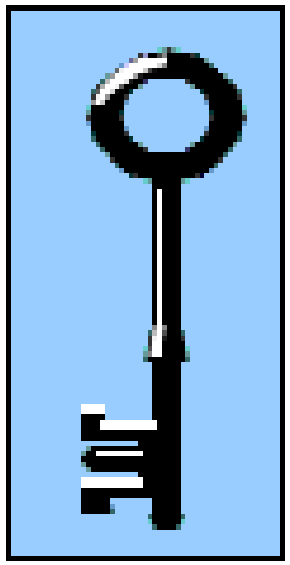
Session Key

Bob sends Alice the encrypted key



Encrypted symmetric key

Alice's private key



Both parties now possess the symmetric key

มาตรฐานการเข้ารหัสข้อมูล : Diffie-Hellman

- เป็น Public Key Cryptography แบบแรกที่ได้รับการตีพิมพ์
- ใช้หลักการ One-Way Fuction ที่ว่า การคำนวณค่าเอ็กซ์โพเนนต์ (ยกกำลัง) ง่ายกว่าการคำนวณล็อกการิทึม
- อัลกอริทึมนี้จะอนุญาตให้คนสองคนสามารถสร้าง ซีเคร็ทคีย์เพื่อใช้ในการเข้ารหัสข้อมูลที่รับส่งกันได้
- ในปี 2015 มีผลการวิจัยว่าอัลกอริทึมนี้ไม่แข็งแกร่งมากพอ

Diffie-Hellman Key Exchange

Alice

Bob

Bob and Alice know and have the following :
p = 23 (a prime number) g = 11 (a generator)

Alice	Bob
Alice chooses a secret random number a = 6	Bob chooses a secret random number b = 5
Alice computes : $A = g^a \bmod p$ $A = 11^6 \bmod 23 = 9$	Bob computes : $B = g^b \bmod p$ $B = 11^5 \bmod 23 = 5$
Alice receives B = 5 from Bob	**Bob receives A = 9 from Alice**
Secret Key = $K = B^a \bmod p$	Secret Key = $K = A^b \bmod p$
$K = 5^6 \bmod 23 = 8$	$K = 9^5 \bmod 23 = 8$

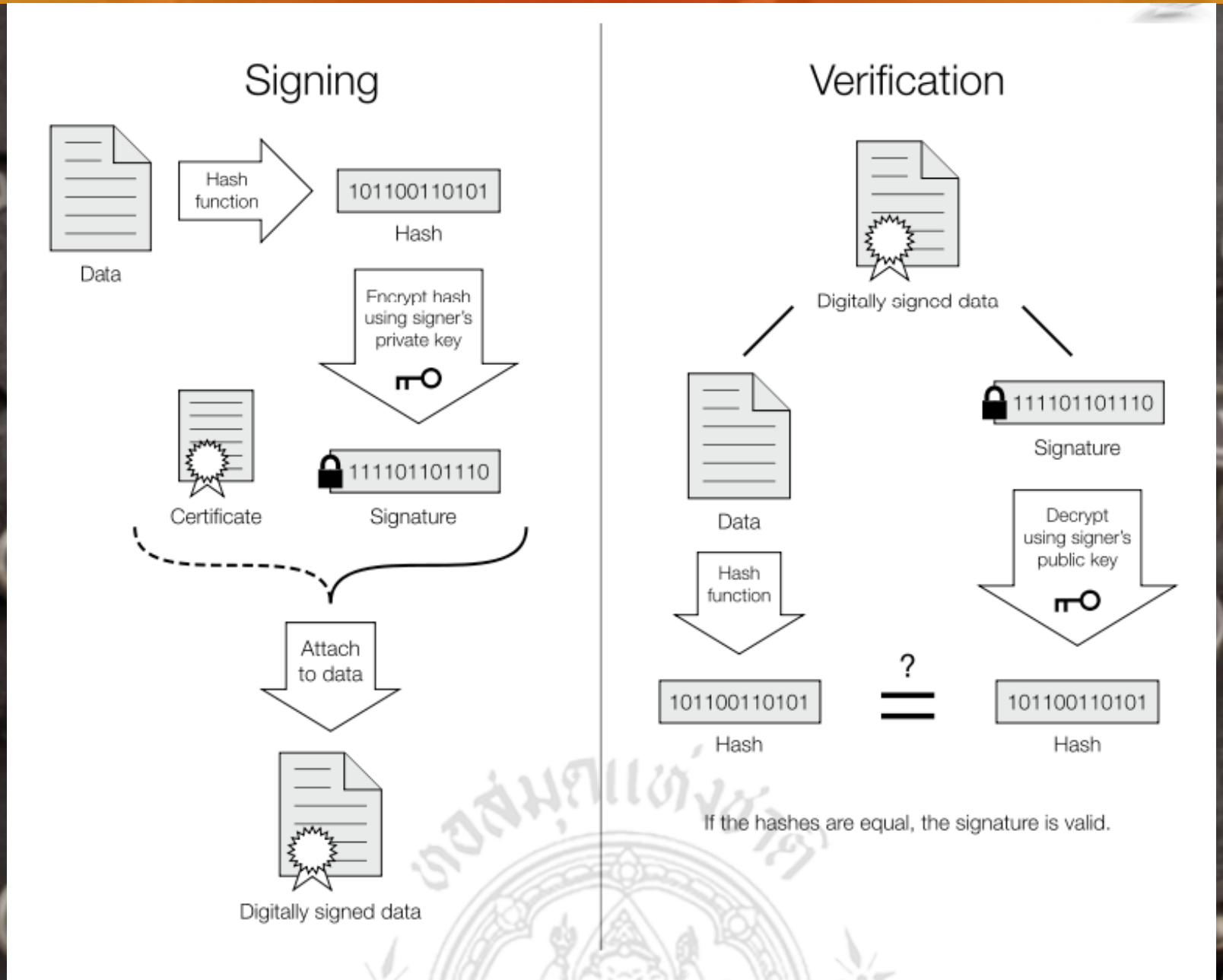
The common secret key is : 8

N.B. We could also have written : $K = g^{ab} \bmod$

มาตรฐานการเข้ารหัสข้อมูล
: Digital Signature Algorithm

- เป็นอัลกอริทึมในการ**ตรวจสอบ**ว่าข้อความนั้น**ไม่ได้ถูกแก้ไขเปลี่ยนแปลง**ระหว่างการรับส่ง รวมไปถึงเป็นการ**พิสูจน์ทราบตัวตน**ของผู้ส่งข้อความ
- ผู้ที่ลงลายเซ็นจะใช้ไพรเวทคีย์ของตัวเองลงลายเซ็น ส่วนผู้รับจะใช้พับลิกคีย์ของผู้ส่งในการตรวจสอบ

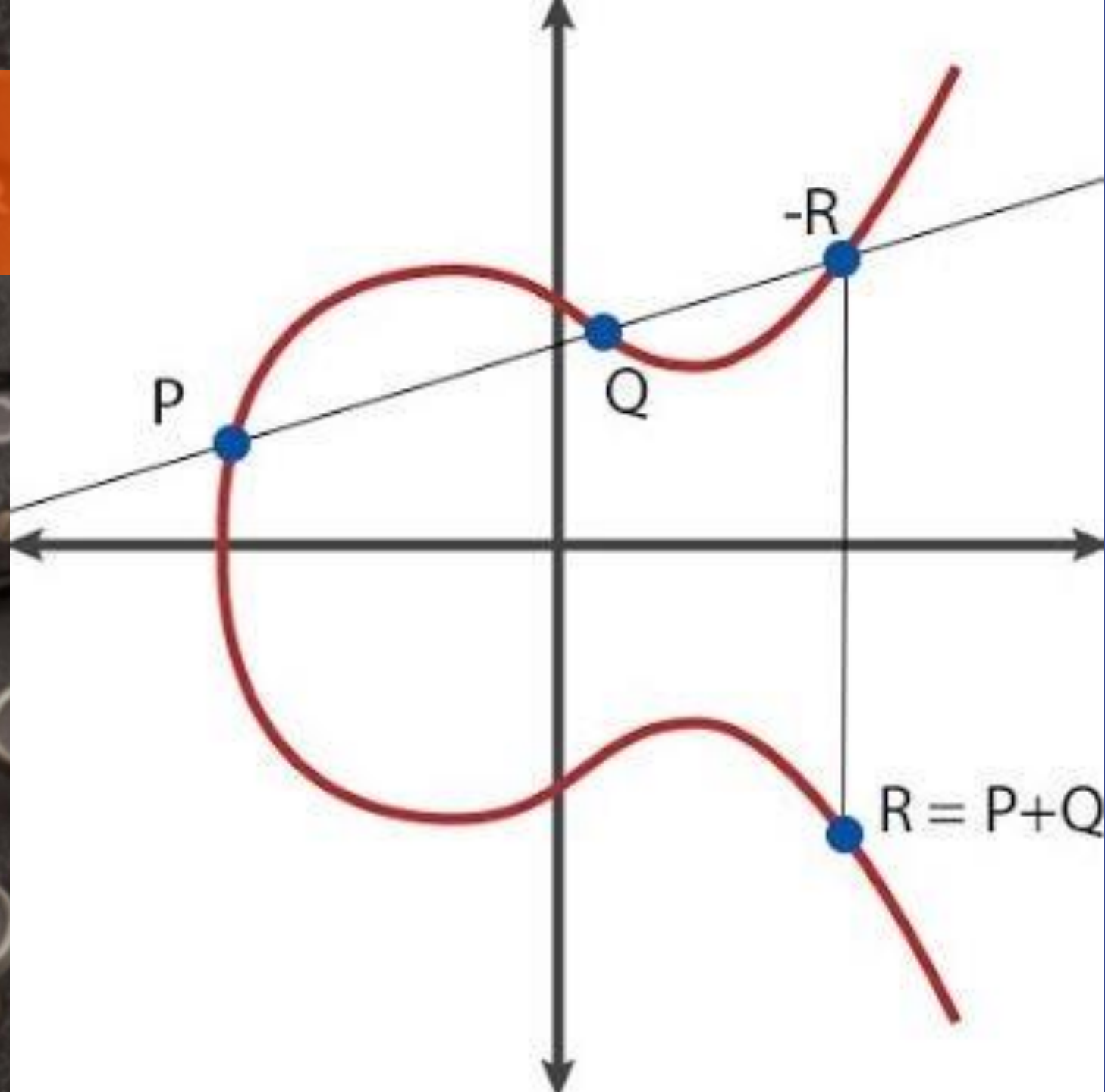
กระบวนการทำงานของ Digital Signature



มาตรฐานการเข้ารหัสข้อมูล
: Elliptic Curve Cryptography (ECC)

- ECC เป็นอัลกอริทึมที่ใช้สมการเส้นโค้งรี (Elliptic Curve) คิดค้นในปี 1985 โดย Victor Miller และ Neal Koblitz
- ข้อดีคือคีย์ที่ใช้ไม่ต้องมีความยาวมาก

Elliptic Curve



ตารางการเปรียบเทียบ RSA และ ECC

RSA Key size	Time to Break Key (MIPS Years)	ECC KeySize	RSA:ECC Key-size Ratio
512	10^4	106	5:1
768	10^8	132	6:1
1,024	10^{11}	160	7:1
2,048	10^{20}	210	10:1
21,000	10^{78}	600	35:1