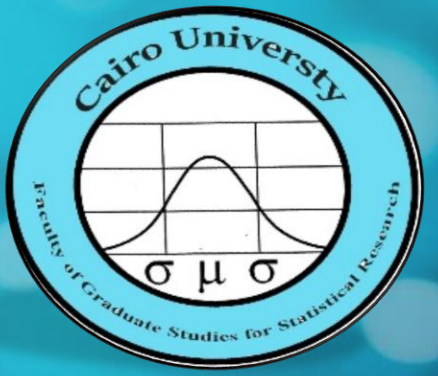
Cairo University



Information Security

(SE205)

Dr. Hany Mohamed

Information Security

Unit 3: Asymmetric Cryptography

Asymmetric Cryptography

What is an Asymmetric cryptography?

- There are two sides in an encrypted communication: the sender, who encrypts the data, and the recipient, who decrypts it.
- As the name implies, asymmetric encryption is different on each side; the sender and the recipient use two different keys.
- **Asymmetric encryption**, also known as **public key encryption**, uses a public key-private key pairing: data encrypted with the public key can only be decrypted with the private key.
- In symmetric encryption, the same key both encrypts and decrypts data. For symmetric encryption to work, the two or more communicating parties must know what the key is; for it to remain secure, no third party should be able to guess or steal the key.

Asymmetric Cryptography

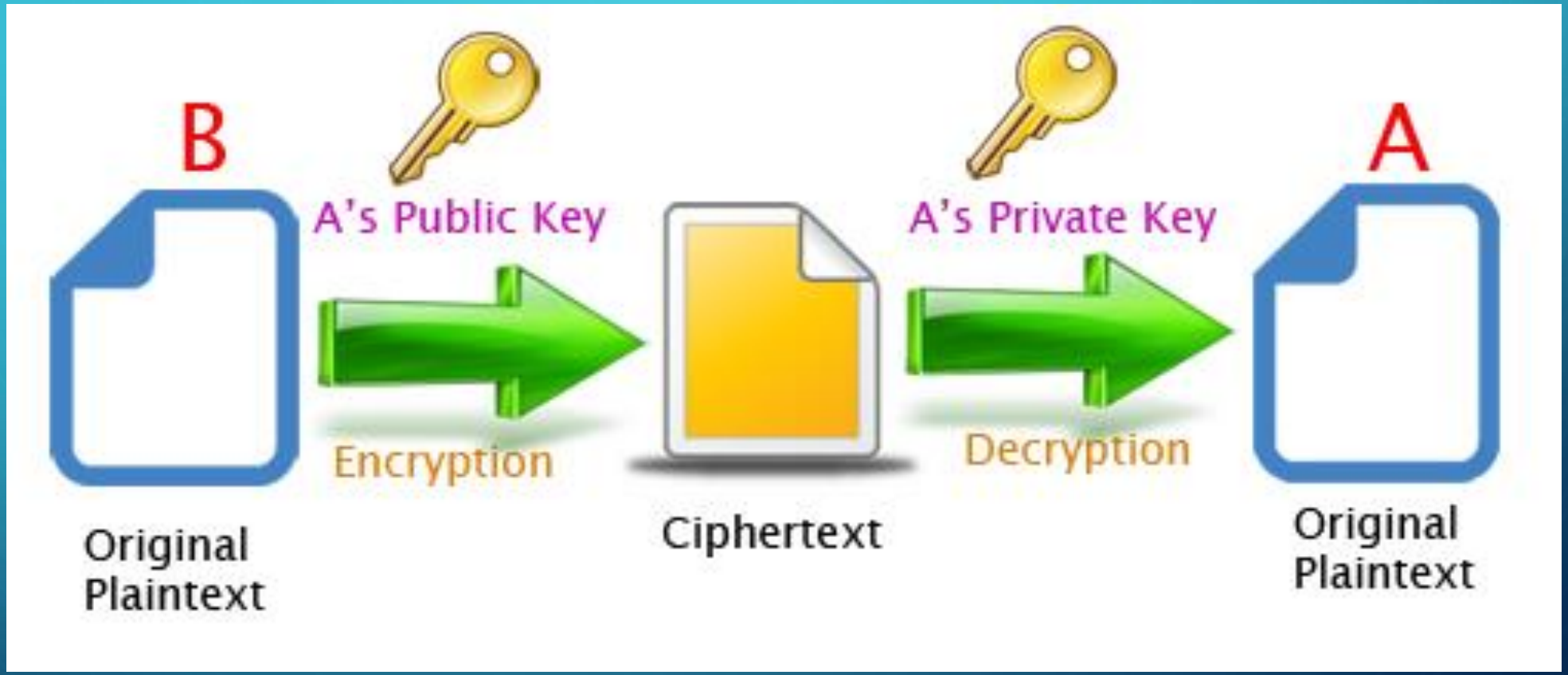
What is an Asymmetric cryptography?

- Symmetric encryption algorithms have been based on the elementary tools of substitutions and permutations.
- Public key algorithms are based on mathematical functions rather than substitutions and permutations.
- As mentioned before, public-key cryptography uses two mathematically related keys:

    1. **Public Key**: Shared openly; used for encryption.
    2. **Private Key**: Kept secret; used for decryption.

- Key principle: Data encrypted with the public key can only be decrypted with the private key, and vice versa.
- Example: Secure email communication (e.g., PGP).

Asymmetric Cryptography



Asymmetric Cryptography

What is an Asymmetric cryptography?

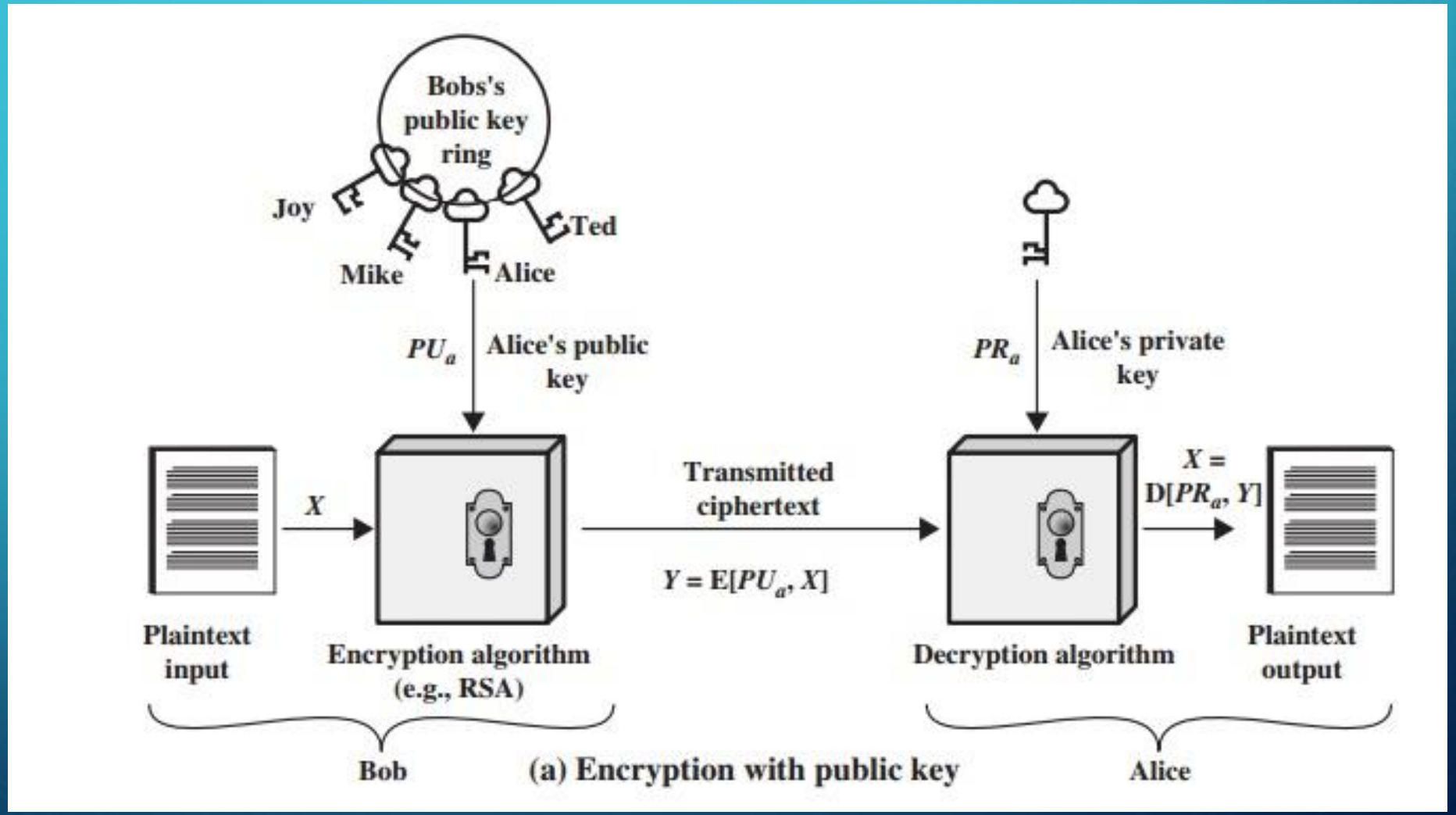
- Assume that there are two entities, PersonB and PersonA, and that PersonB wishes to send a very, very secret message to PersonA.
- PersonB would obviously want to fully encrypt, for obvious security reasons and privacy concerns, the content of the very, very secret message that he wishes to send to PersonA, before PersonB sends out the message to PersonA.
- In this situation, PersonA creates a Public and Private Key pair.
- PersonA sends the public key to PersonB.
- Anyone can be aware of what the public key is and so, it really doesn't matter how PersonA sends it to PersonB – the encryption scheme wouldn't be compromised in any manner whatsoever, even if a rogue person gets hold of the public key.
- PersonB writes the very, very secret message, encrypts the message using the public key received from PersonA, sends the encrypted message to PersonA who can then decrypt the message using the private key.

Asymmetric Cryptography

What is an Asymmetric cryptography?

- The public and private keys are generated together.
- While the public key is used to encrypt the data, the private key is used to decrypt the encrypted data.
- So, how does it become possible that you can make the public key public, without revealing any information at all about the private key? The answer to this question forms the basis of the whole RSA asymmetric encryption scheme.

Asymmetric Cryptography



Asymmetric Cryptography

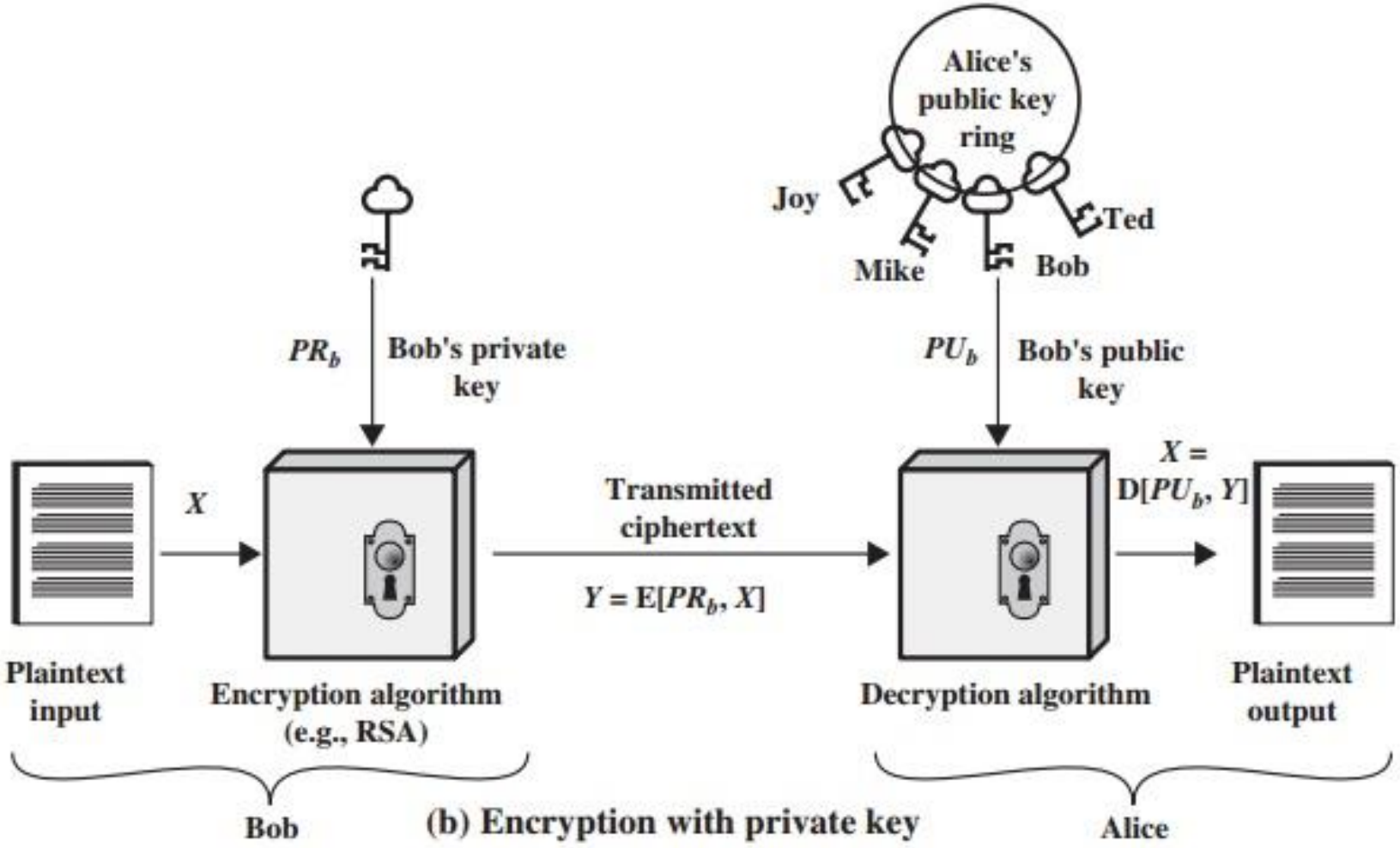


Figure 9.1 Public-Key Cryptography

Asymmetric Cryptography

The essential steps:

Step 1: Key generation

Each person (or their computer) must generate a pair of keys that identifies them: a **private key** and a **public key**.

Step 2: Key exchange

The sending and receiving computers exchange *public keys* with each other via a reliable channel, like TCP/IP. The private keys are *never* exchanged.

Step 3: Encryption

The sending computer encrypts the secret data using the receiving computer's public key and a mathematical operation.

The power of public key encryption is in that mathematical operation. It's a "one-way function", which means it's incredibly difficult for a computer to reverse the operation and discover the original data. Even the public key cannot be used to decrypt the data.

Asymmetric Cryptography

The essential steps:

Step 4: Sending encrypted data

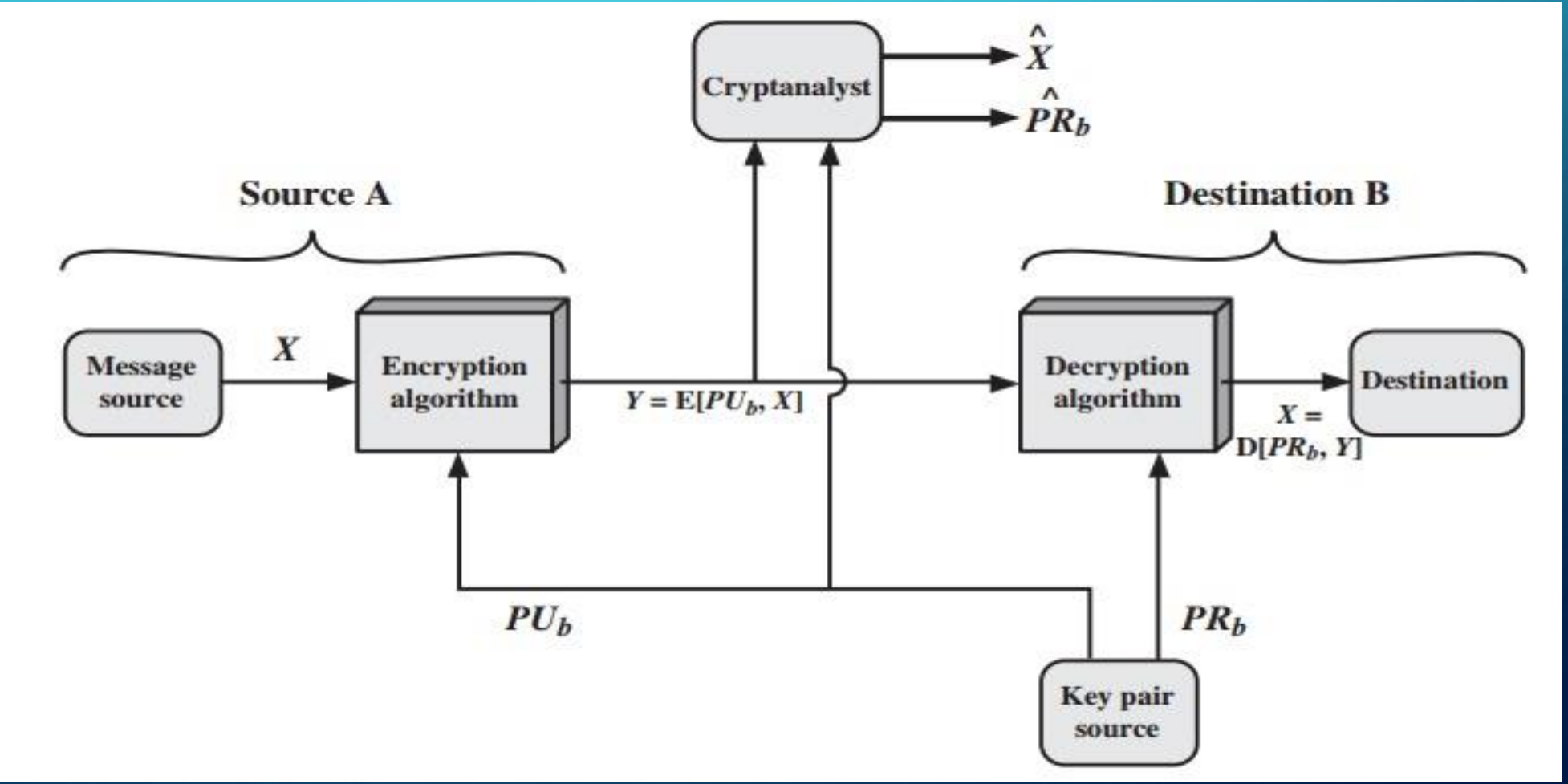
The sender can now safely transmit the encrypted data over the Internet without worry of onlookers.

Step 5: Decryption

Now the receiver can decrypt the message, using their private key. That's the only key that can be used to decrypt the message (in the world!).

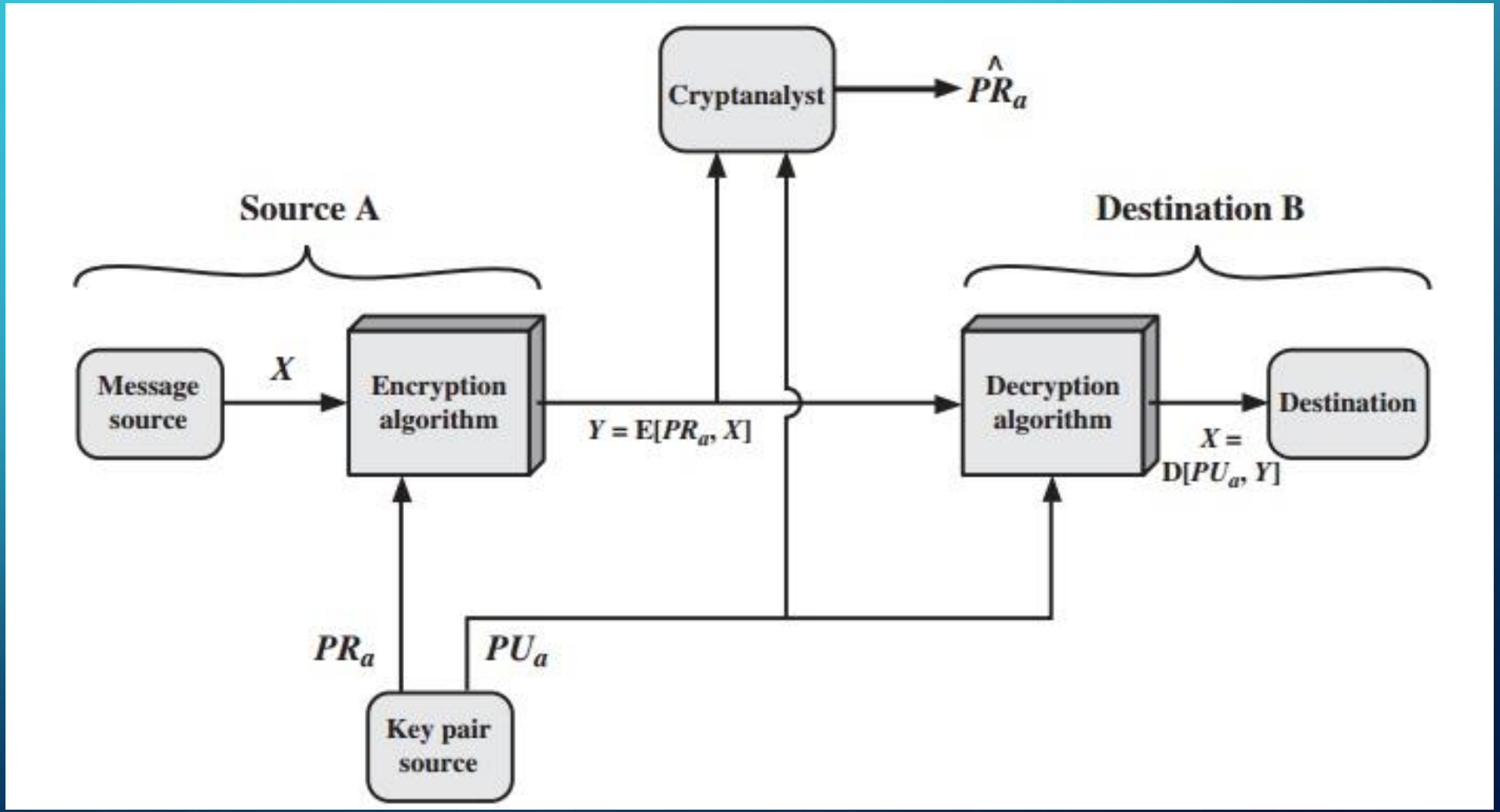
Asymmetric Cryptography

Public-Key cryptosystem : Secrecy



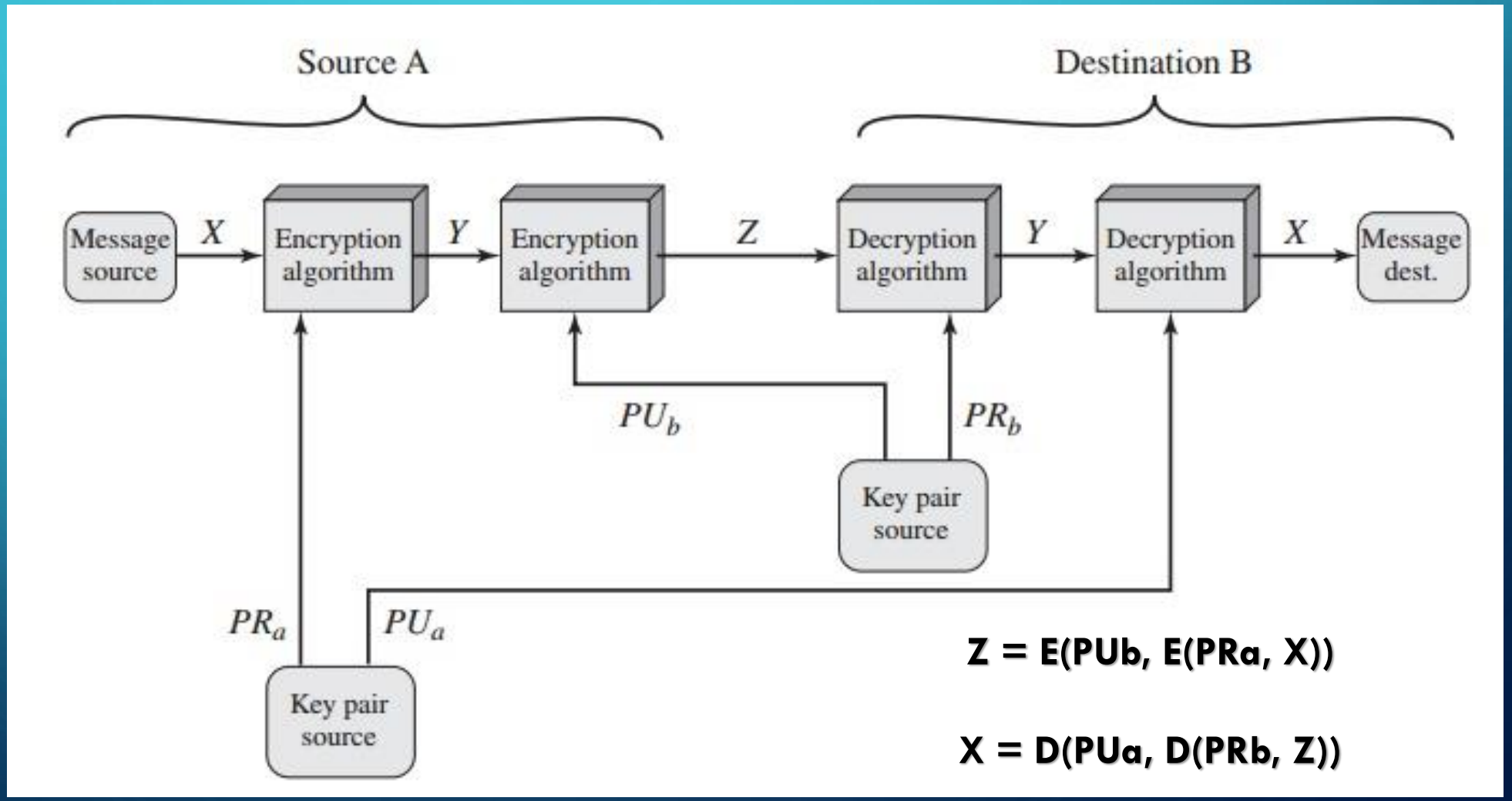
Asymmetric Cryptography

Public-Key cryptosystem : Authentication



Asymmetric Cryptography

Public-Key cryptosystem : Authentication & Secrecy



Asymmetric Cryptography

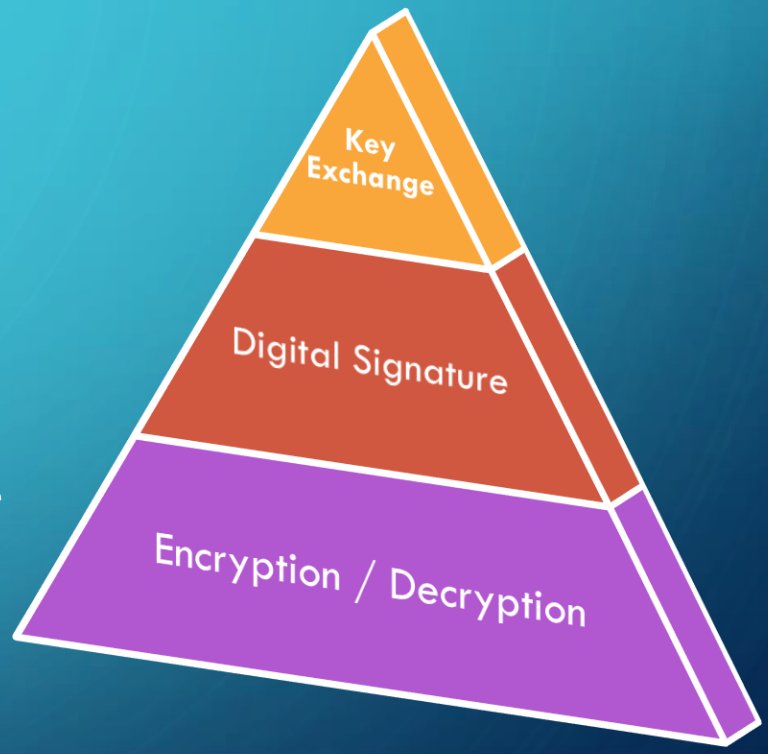
Symmetric v/s Asymmetric

Characteristic	Symmetric Key Cryptography	Asymmetric Key Cryptography
Key used for encryption / decryption	Same key is used for encryption and decryption	One key used for encryption and another, different key is used for decryption
Speed of encryption / decryption	Very fast	Slower
Size of resulting encrypted text	Usually same as or less than the original clear text size	More than the original clear text size
Key agreement / exchange	A big problem	No problem at all
Number of keys required as compared to the number of participants in the message exchange	Equals about the square of the number of participants, so scalability is an issue	Same as the number of participants, so scales up quite well
Usage	Mainly for encryption and decryption (confidentiality), cannot be used for digital signatures (integrity and non-repudiation checks)	Can be used for encryption and decryption (confidentiality) as well as for digital signatures (integrity and non-repudiation checks)

Asymmetric Cryptography

Applications of Public-key cryptosystem:

- **Encryption/decryption:** The sender encrypts a message with the recipient's public key.
- **Digital signature:** The sender "signs" a message with its private key. Signing is achieved by a cryptographic algorithm applied to the message or to a small block of data that is a function of the message.



- **Key exchange:** Two sides cooperate to exchange a session key.

Several different algorithms are suitable for the three applications, whereas others can be used only for one or two.

Asymmetric Cryptography

Algorithm	Encryption/Decryption	Digital Signature	Key Exchange
RSA	Yes	Yes	Yes
Elliptic Curve	Yes	Yes	Yes
Diffie-Hellman	No	No	Yes
DSS	No	Yes	No

Asymmetric Cryptography

Public Key Cryptography Requirements:

- The **computation** of the **pair** of **keys** i.e. private and the public key must be **easy**.
- Knowing the **encryption algorithm** and **public key** of the intended **receiver**, **computation** of **cipher** text **must** be **easy**.
- For a **receiver** of the **message**, it should be **computationally easy** to **decrypt** the obtained cipher text using his **private key**.
- It is also **required** that any **opponent** in the network **knowing** the **public key should** be **unable** to **determine** its **corresponding private key**.
- Having the **cipher text** and **public key** an **opponent should** be **unable** to **determine** the **original message**.

Asymmetric Cryptography

Public Key Cryptanalysis:

- **Brute-force attack**
  - Use large key to avoid brute force attacks
  - Public key algorithms less efficient with larger keys (Slowing)
  - Public key cryptography mainly used for key management and signatures

- **Compute private key from public key**
  - No known feasible methods using standard computing

Asymmetric Cryptography

Is Asymmetric more secure than Symmetric key encryption?

1. Security Strength

- Symmetric Encryption (e.g., AES, ChaCha20)
  - Uses the same key for encryption and decryption.
  - Generally faster and more efficient than public-key encryption.
  - More resistant to brute-force attacks at the same key length.
  - A 256-bit symmetric key is roughly as secure as a 3072-bit RSA key.
- Public-Key Encryption (e.g., RSA, ECC, DH)
  - Uses a key pair: a public key for encryption and a private key for decryption.
  - More computationally expensive and slower than symmetric encryption.
  - Key exchange security: Unlike symmetric encryption, public keys can be shared openly, reducing key distribution risks.

Asymmetric Cryptography

2. Key Management and Distribution

Symmetric Encryption

Requires both parties to **securely share and store** the same secret key.

Risky for large-scale communication (e.g., if the key is intercepted, all encrypted data is compromised).

Public-Key Encryption

Public keys can be freely distributed, eliminating the need for a secure channel for key exchange.

More secure for exchanging keys (e.g., TLS/SSL uses public-key cryptography for key exchange, then switches to symmetric encryption for speed).

3. Use Cases

Symmetric Encryption is best for encrypting large amounts of data efficiently (e.g., encrypting files, databases, VPNs).

Public-Key Encryption is best for secure key exchange, digital signatures, and authentication (e.g., SSL/TLS, PGP, Bitcoin wallets).

Asymmetric Cryptography

Is Asymmetric more secure than Symmetric key encryption?

Conclusion: Which is More Secure?

If key exchange is a concern, **public-key** encryption is **more secure** because it avoids the need to share a secret key.

If **performance and bulk data encryption** are the main goals, **symmetric encryption is stronger and more efficient**.

A common best practice is to use both together:

Use public-key encryption to securely exchange a **symmetric key**.

Use symmetric encryption for **fast, secure** data **encryption**. (This is how SSL/TLS and PGP work.)

Asymmetric Cryptography (RSA)

Ron **R**ivest, Adi **S**hamir and Len **A**dleman

Created in 1978; RSA Security sells related products

Most widely used public-key algorithm

Block cipher: plaintext and cipher text are **integers**

Asymmetric Cryptography (RSA)

The main steps of RSA algorithm are the following:

Key generation

Select p and q	p and q both prime
Calculate $n = pq$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$d \equiv e^{-1} \bmod \phi(n)$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

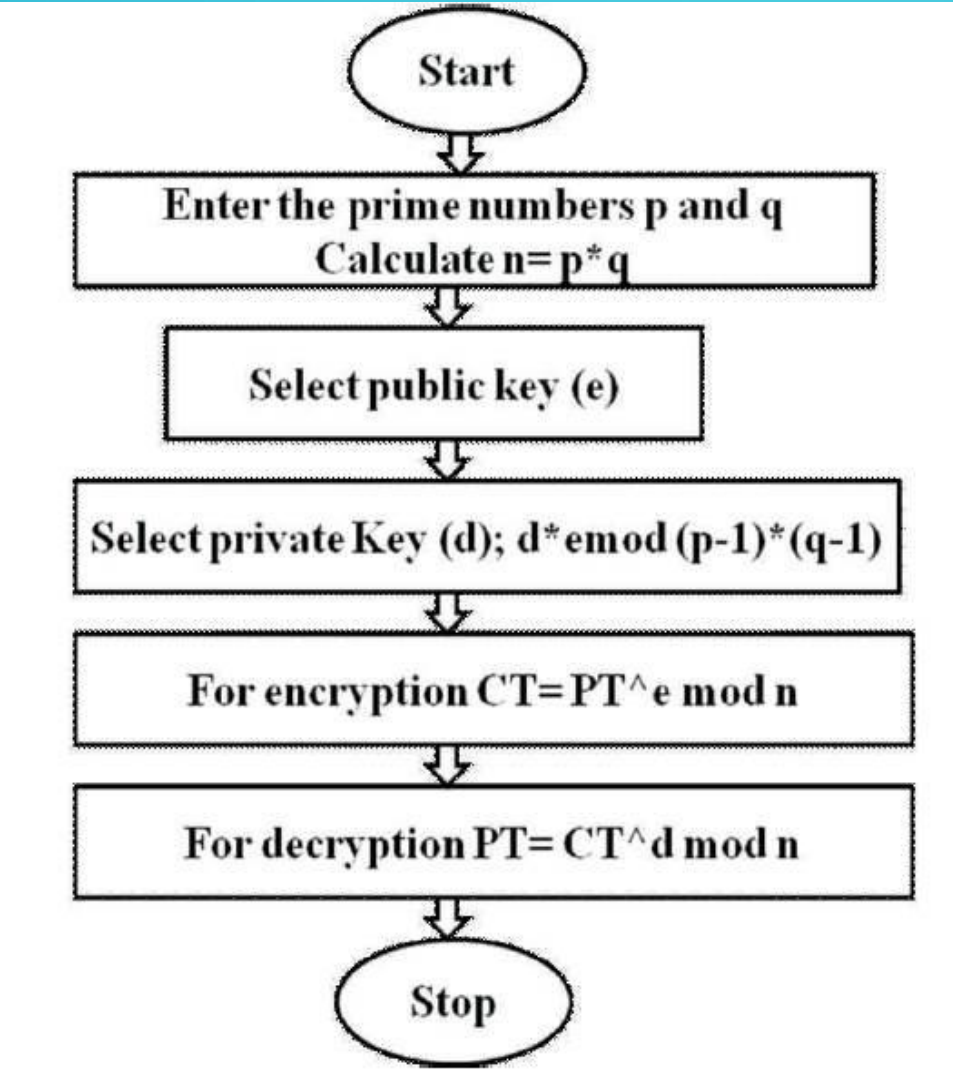
Encryption

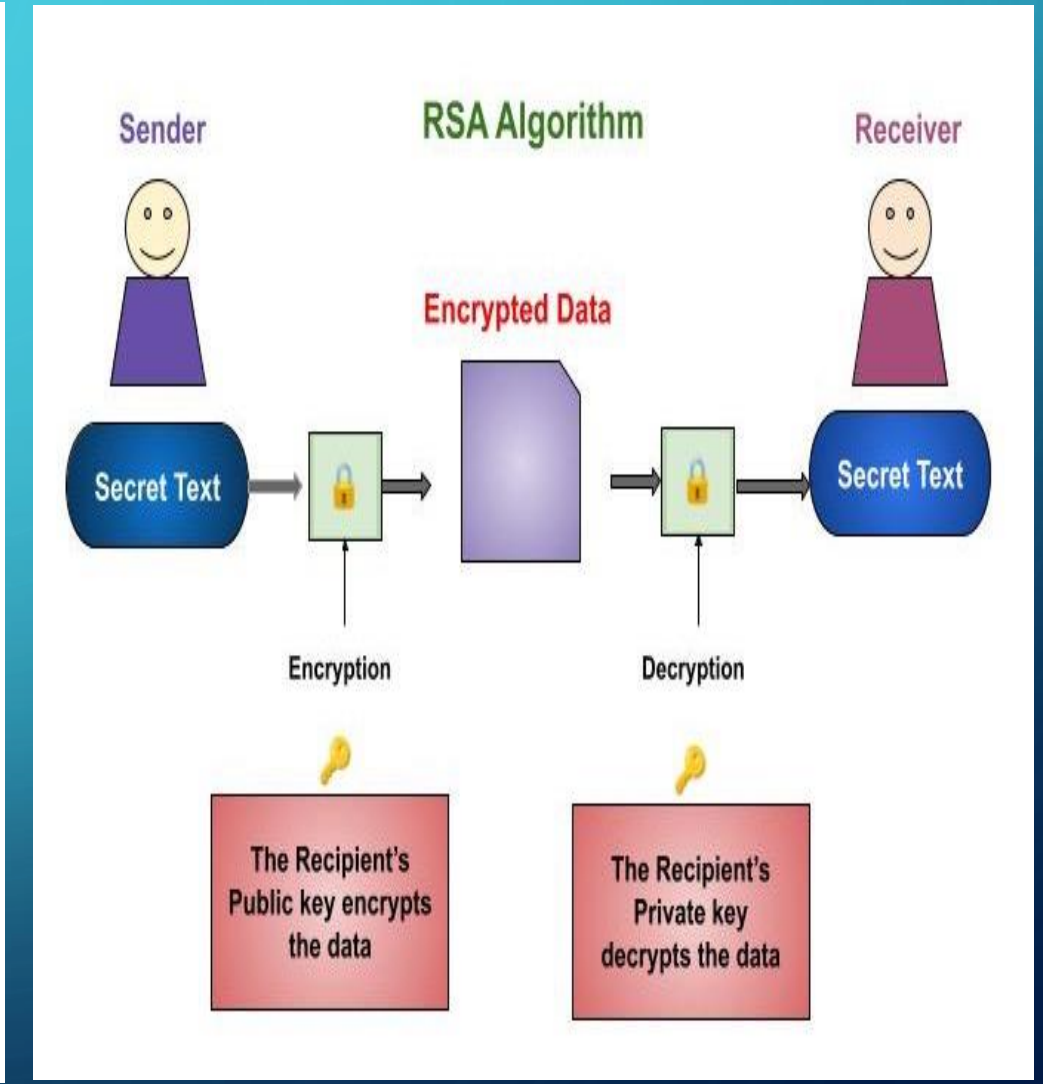
Plaintext :	$M < n$
Ciphertext :	$C = M^e \bmod n$

Decryption

Plaintext:	C
Ciphertext:	$M = C^d \pmod n$

Asymmetric Cryptography (RSA)





Asymmetric Cryptography (RSA)

RSA example:

Step 1: Select two **prime numbers p and q** (where p ≠ q & only be divided on itself and 1)

Example, Two prime numbers **p** = 13, **q** = 11.

Step 2: Calculate **n** = p * q. (must be large number, ex. for 1024 key length, its used 309 digits).

Example, n = p * q = 13 * 11 = 143.

Step 3: Calculate Euler Totient Function Φ(n) = (p-1) * (q-1).

Example, **Φ(n)** = (13 – 1) * (11 – 1) = 12 * 10 = 120.

Step 4: Select encryption exponent (**e**) such that, **e** is relatively prime to Φ(n), i.e. (e, Φ(n)) = 1 and 1 < e < Φ(n). (e must be grater then 1, smaller than euler, coprime with euler)

Example, Select e = 13, gcd (13, 120) = 1.

Step 5: Calculate decryption exponent d = e -1 mod Φ(n) or e * d = 1 mod Φ(n)

Example, Finding d: e * d mod Φ(n) = 1

Asymmetric Cryptography (RSA)

Example, Finding d: $e * d \bmod \Phi(n) = 1$

$13 * d \bmod 120 = 1$

(How to find: $d * e = 1 \bmod \Phi(n)$ → $d = e^{-1} \bmod \Phi(n)$

$d = ((\Phi(n) * i) + 1) / e$

$d = (120 + 1) / 13 = 9.30$ (∵ i = 1)

$d = (240 + 1) / 13 = 18.53$ (∵ i = 2)

$d = (360 + 1) / 13 = 27.76$ (∵ i = 3)

$d = (480 + 1) / 13 = 37$ (∵ i = 4))

Step 6: Public key = {e, n}, private key = {d, n}.

Example, Public key = {13, 143} and private key = {37, 143}.

Step 7: Find out cipher text using the formula, $C = M^e \bmod n$ where, $M < n$.

Example, Plain text M = 13. (Where, $M < n$)

$C = M^e \bmod n = 13^{13} \bmod 143 = 52$.

Step 8: $M = C^d \bmod n$. Plain text M can be obtain using the given formula.

Example, Cipher text C = 52

$M = C^d \bmod n = 52^{37} \bmod 143 = 13$

Asymmetric Cryptography (RSA)

Exercise - 1

Question: P and Q are two prime numbers. P=7, and Q=17. Take public key E=5. If plain text value is 6, then what will be cipher text value according to RSA algorithm? Again calculate plain text value from cipher text.

Solution:

1. Two prime numbers P=7, Q=17
2. $n = P * Q = 17 * 7 = 119$ **n = 119**
3. $\Phi(n) = (P-1) * (Q-1) = (17 - 1) * (7 - 1) = 16 * 6 = 96$ **Φ(n) = 96**
4. Public key E = 5. **E= 5**
5. Calculate d = 77. $d = ((\Phi(n) * i) + 1) / e$ **d= 77**

   $d = ((96*1)+1) / 5 = 19.4$

   $d = ((96*2)+1) / 5 = 38.6$

   $d = ((96*3)+1) / 5 = 57.8$

   $d = ((96*4)+1) / 5 = 77$ (Stop finding d because getting integer value)
6. Public key = {e, n} = {5, 119}, private key = {d, n} = {77, 119}.
7. Plain text PT = 6, $CT = PT^E \bmod n = 6^5 \bmod 119 = 41$. **Cipher Text = 41**
8. Cipher text CT = 41, $PT = CT^d \bmod n = 41^{77} \bmod 119 = 6$. **Plain Text = 6**

Asymmetric Cryptography (RSA)

Exercise - 2

Question: In a public key cryptosystem using RSA algorithm, user uses two prime numbers 5 and 7. He chooses 11 as Encryption key, find out decryption key. What will be the cipher text, if the plaintext is 2? Decrypt the cipher text, what will be the value of plain text?

Solution:

1. Two prime numbers $p = 5$, $q = 7$
2. $n = p * q = 5 * 7 = 35$ **n = 35**
3. $\Phi(n) = (p-1) * (q-1) = (5 - 1) * (7 - 1) = 4 * 6 = 24$ **Φ(n) = 24**
4. Public key $e = 11$. **e= 11**
5. Calculate $d = 11$. $d = ((\Phi(n) * i) + 1) / e$ **d= 11**
6. Public key = {e, n} = {11, 35}, private key = {d, n} = {11, 35}.
7. Plain text $P = 2$, $C = P^e \bmod n = 2^{11} \bmod 35 = 18$. **Cipher Text = 18**
8. Cipher text $C = 18$, $P = C^d \bmod n = 18^{11} \bmod 35 = 2$. **Plain Text = 2**

Asymmetric Cryptography (RSA)

Exercise - 3

Question: P and Q are two prime numbers. P=17, and Q=11. Take public key E=7. If plain text value is 5, then what will be cipher text value & private key value according to RSA algorithm? Again calculate plain text value from cipher text.

Solution:

1. Two prime numbers p = 17, q = 11
2. $n = p * q = 17 * 11 = 187$ **n = 187**
3. $\Phi(n) = (p-1) * (q-1) = (17 - 1) * (11 - 1) = 16 * 10 = 160$ **Φ(n) = 160**
4. Public key e = 7. **e= 7**
5. Calculate d = 23. $d = ((\Phi(n) * i) + 1) / e$ **d= 23**
6. Public key = {e, n} = {7, 187}, private key = {d, n} = {23, 187}.
7. Plain text P = 5, $C = P^e \bmod n = 5^7 \bmod 187 = 146$. **Cipher Text = 146**
8. Cipher text C = 146, $P = C^d \bmod n = 146^{23} \bmod 187 = 5$. **Plain Text = 5**

Asymmetric Cryptography (RSA)

- **Advantages**
- **Security**: RSA algorithm is considered to be very secure and is widely used for secure data transmission.
- **Public-key cryptography**: RSA algorithm is a public-key cryptography algorithm, which means that it uses two different keys for encryption and decryption. The public key is used to encrypt the data, while the private key is used to decrypt the data.
- **Key exchange**: RSA algorithm can be used for secure key exchange, which means that two parties can exchange a secret key without actually sending the key over the network.
- **Digital signatures**: RSA algorithm can be used for digital signatures, which means that a sender can sign a message using their private key, and the receiver can verify the signature using the sender's public key.
- **Widely used**: Online banking, e-commerce, and secure communications are just a few fields and applications where the RSA algorithm is extensively developed.

Asymmetric Cryptography (RSA)

- **Disadvantages**
- **Slow processing speed**: RSA algorithm is slower than other encryption algorithms, especially when dealing with large amounts of data.
- **Large key size**: RSA algorithm requires large key sizes to be secure, which means that it requires more computational resources and storage space.
- **Vulnerability to side-channel attacks**: RSA algorithm is vulnerable to side-channel attacks, which means an attacker can use information leaked through side channels such as power consumption, electromagnetic radiation, and timing analysis to extract the private key.
- **Limited use in some applications**: RSA algorithm is not suitable for some applications, such as those that require constant encryption and decryption of large amounts of data, due to its slow processing speed.
- **Complexity:** The RSA algorithm is a sophisticated mathematical technique that some individuals may find challenging to comprehend and use.
- **Key Management**: The secure administration of the private key is necessary for the RSA algorithm, although in some cases this can be difficult.
- **Vulnerability to Quantum Computing**: Quantum computers have the ability to attack the RSA algorithm, potentially decrypting the data.

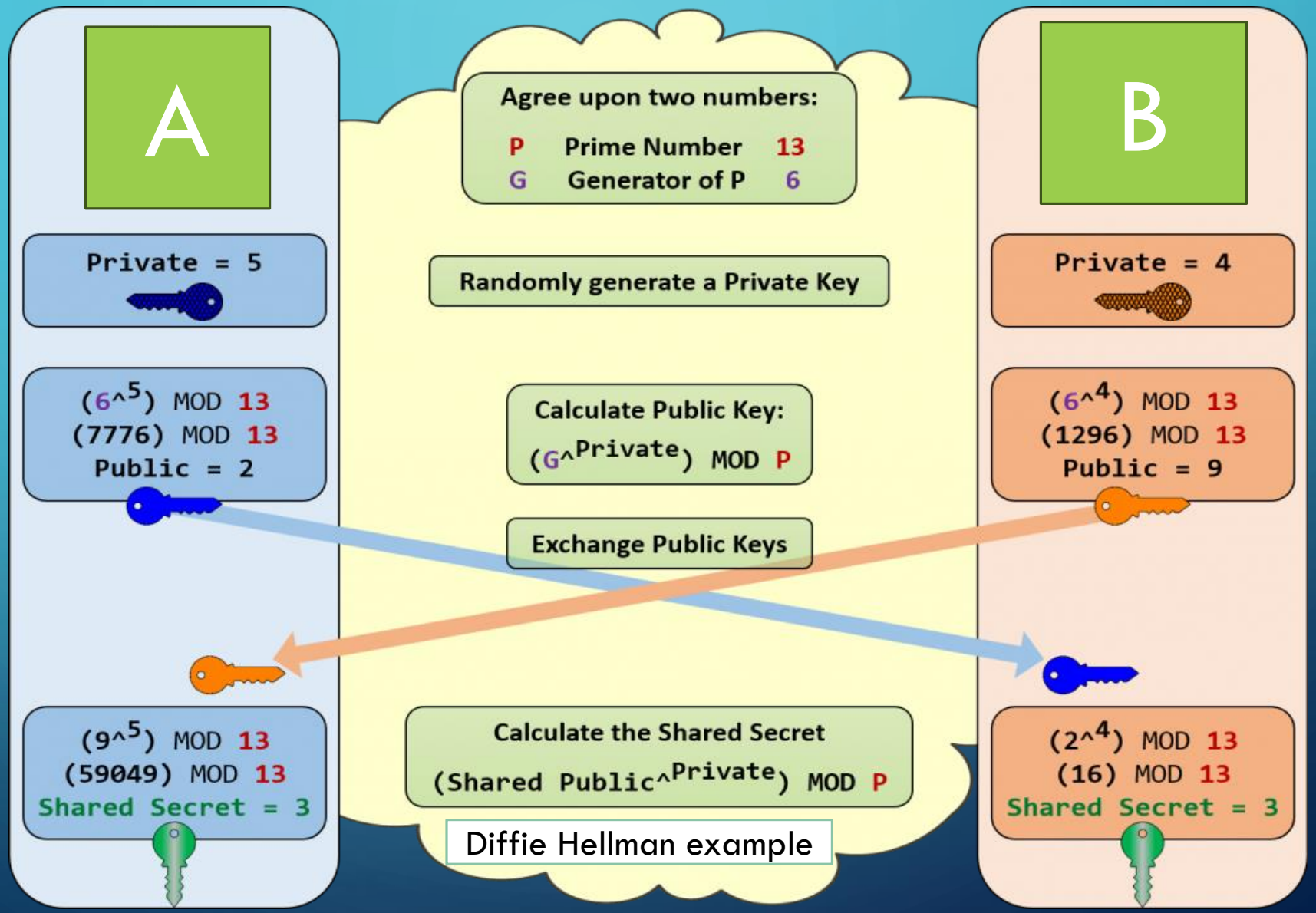
Asymmetric Cryptography (Diffie-Hellman)

- The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret communications while exchanging data over a public network using the elliptic curve to generate points and get the secret key using the parameters.
- For the sake of simplicity and practical implementation of the algorithm, we will consider only 4 variables, one prime (P) and (G), a primitive root of P, and two private values (a) and (b).
- P and G are both publicly available numbers.
- Users (say Alice and Bob) pick private values (a) and (b) and they generate a key and exchange it publicly.
- The opposite person receives the key and that generates a secret key, after which they have the same secret key to encrypt.

Asymmetric Cryptography (Diffie-Hellman)

Alice	Bob
Public Keys available = P, G	Public Keys available = P, G
Private Key Selected = a	Private Key Selected = b
Key generated $x = G^a \bmod P$	Key generated = $y = G^b \bmod P$
Exchange of generated keys takes place	
Key received = y	key received = x
Generated Secret Key = $k_a = y^a \bmod P$	Generated Secret Key = $k_b = x^b \bmod P$
Users now have a symmetric secret key to encrypt	

Asymmetric Cryptography (Diffie-Hellman)



Asymmetric Cryptography (Diffie-Hellman)

- **Step 1:** Alice and Bob get public numbers P = 23, G = 9
- **Step 2:** Alice selected a private key a = 4 an Bob selected a private key b = 3
- **Step 3:** Alice and Bob compute public values
  - Alice: x =(9^4 mod 23) = (6561 mod 23) = 6
  - Bob: y = (9^3 mod 23) = (729 mod 23) = 16
- **Step 4:** Alice and Bob exchange public numbers
- **Step 5:** Alice receives public key y =16 and
  - Bob receives public key x = 6
- **Step 6**: Alice and Bob compute symmetric keys
  - Alice: $k_a = y^a \bmod p = 65536 \bmod 23 = 9$
  - Bob: $k_b = x^b \bmod p = 216 \bmod 23 = 9$
- **Step 7**: 9 is the shared secret.

Asymmetric Cryptography

- **ElGamal Cryptosystem**
  - Similar concepts to Diffie-Hellman
  - Used in digital signature standard and secure email.
- **Elliptic Curve Cryptosystem**
  - Uses elliptic curve athematic (instead of modular athematic in RSA)
  - Equivalent security to RSA with smaller keys (better performance)
  - Used for key exchange and digital signatures

Assignment 3

- Let p = 3 and q= 7, and e = 7, encrypt the message M=4 using RSA.
- Let p = 11 and q = 3, and e = 3, d = 7, decrypt the message C=17.
- Let p = 17 and q = 11, and e= 7, encrypt the message M=88.
- Let P=15, G=9, and a=3, b=5, compute the shared key using Diffie-Hellman.