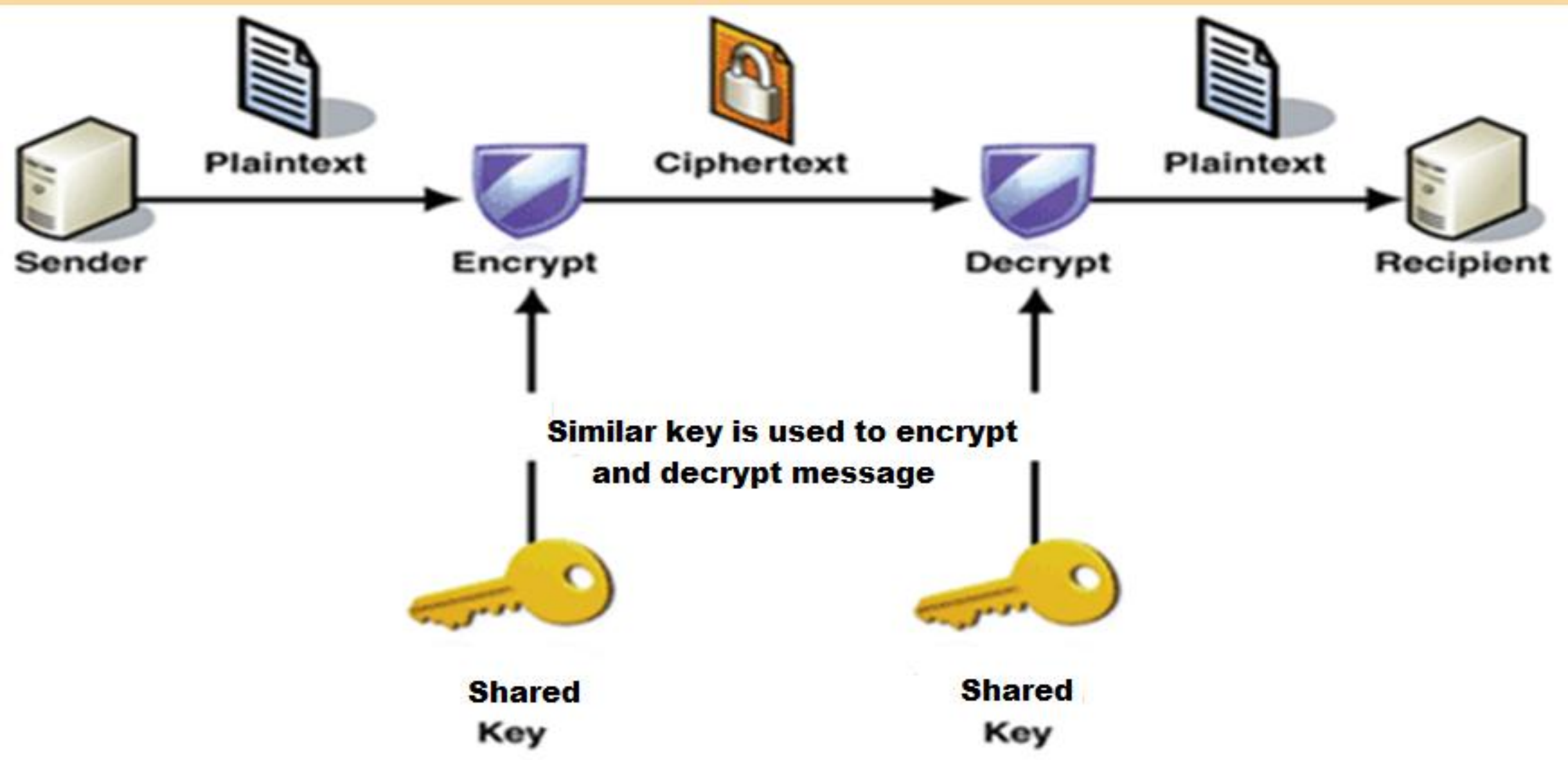
Info Security Technology

Topic 5
Cryptography
(Asymmetric key)

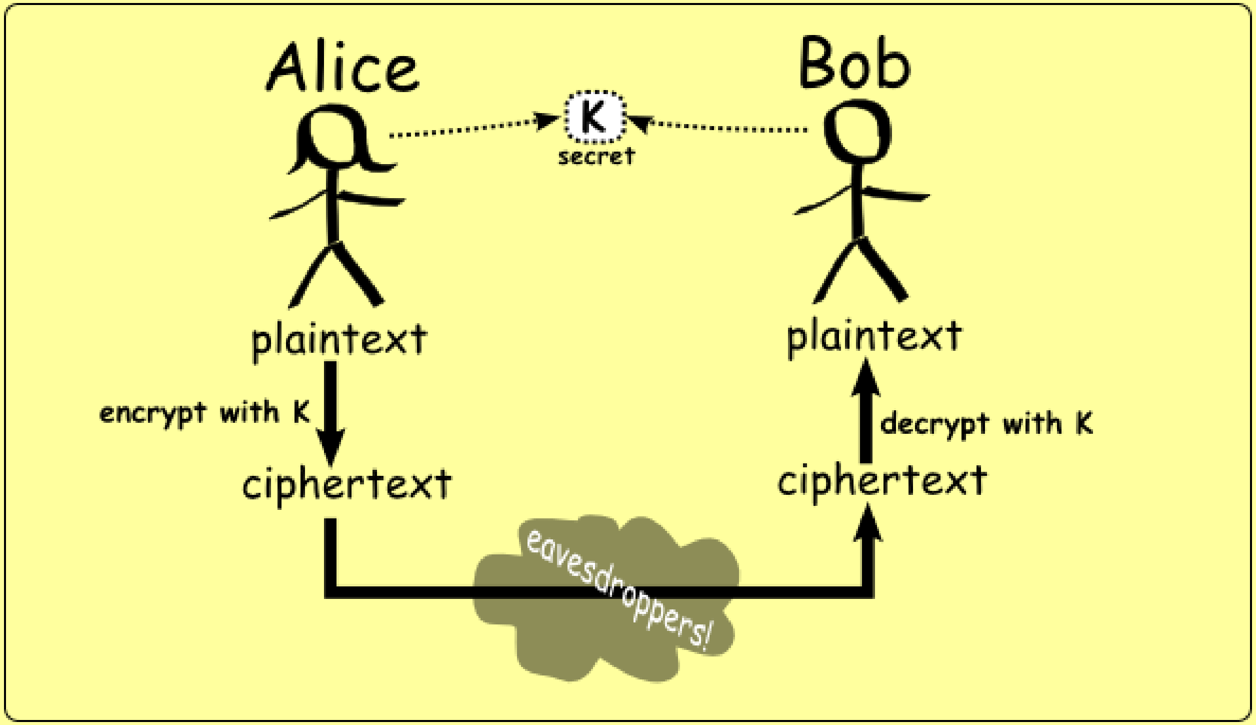
Objectives

- Understand and apply Asymmetric key cryptography
- Understand the various algorithms used in Asymmetric key cryptography
- Compare Asymmetric and Symmetric key algorithm
- How to crack a password using Hash

Symmetric Algorithm



Man-in-the-middle



Symmetric Encryption

- **Symmetric encryption** requires the sender and the receiver to have the **same key**.
  - All symmetric algorithms are based upon this **shared secret** principle.

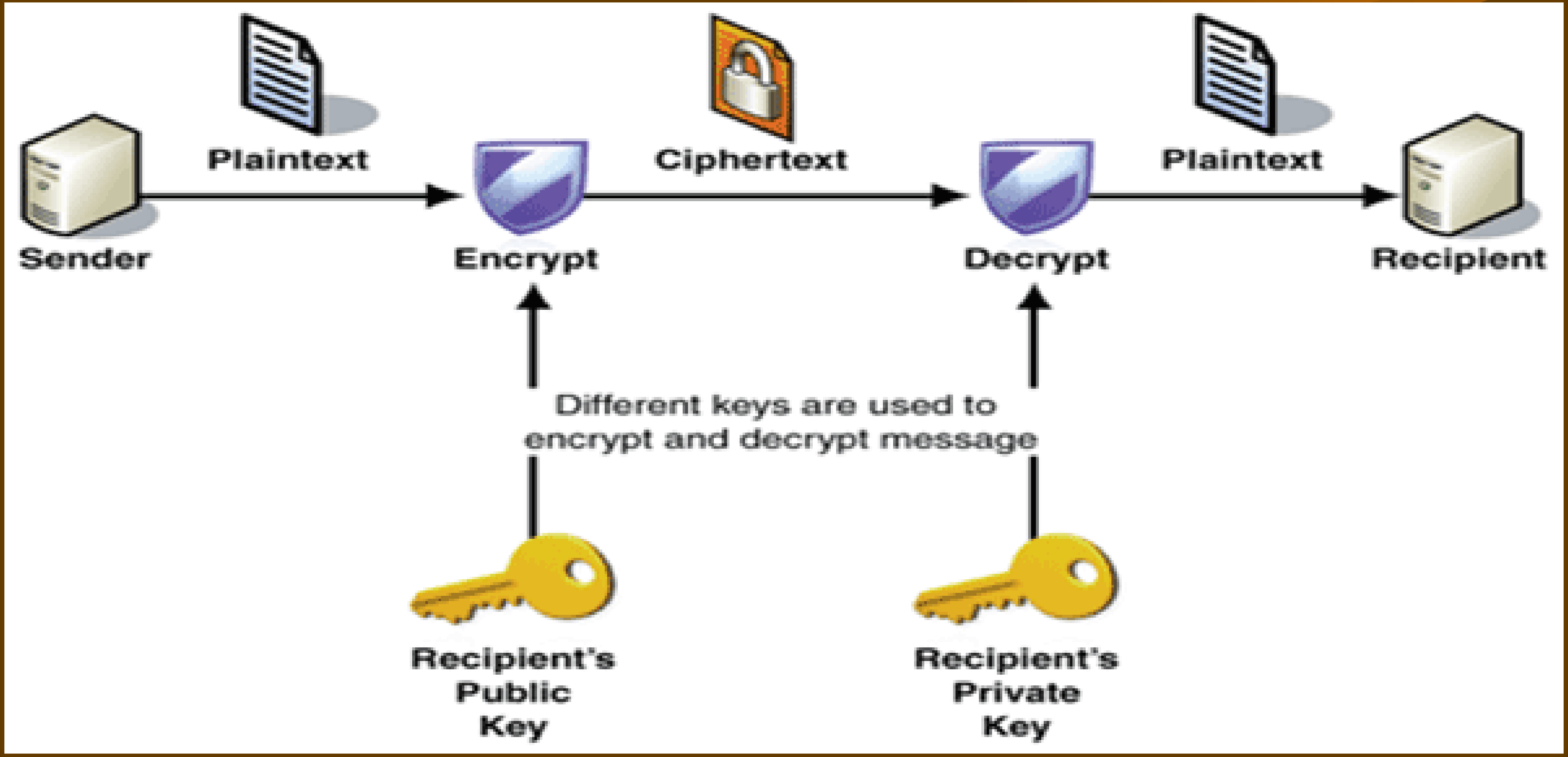
- Symmetric encryption involves a cryptographic key, requiring key management.

- For symmetric algorithms, the most important lesson is to store and send the key only by known secure means.

Symmetric Key Summary

- Symmetric algorithms are comparatively **faster** and have fewer computational requirements.
- Shared key may be lost. NO way to know if message was read or edited by anyone.
- Their main weakness is that two geographically distant parties need to have a **key that match** exactly with each other.

Asymmetric Algorithm

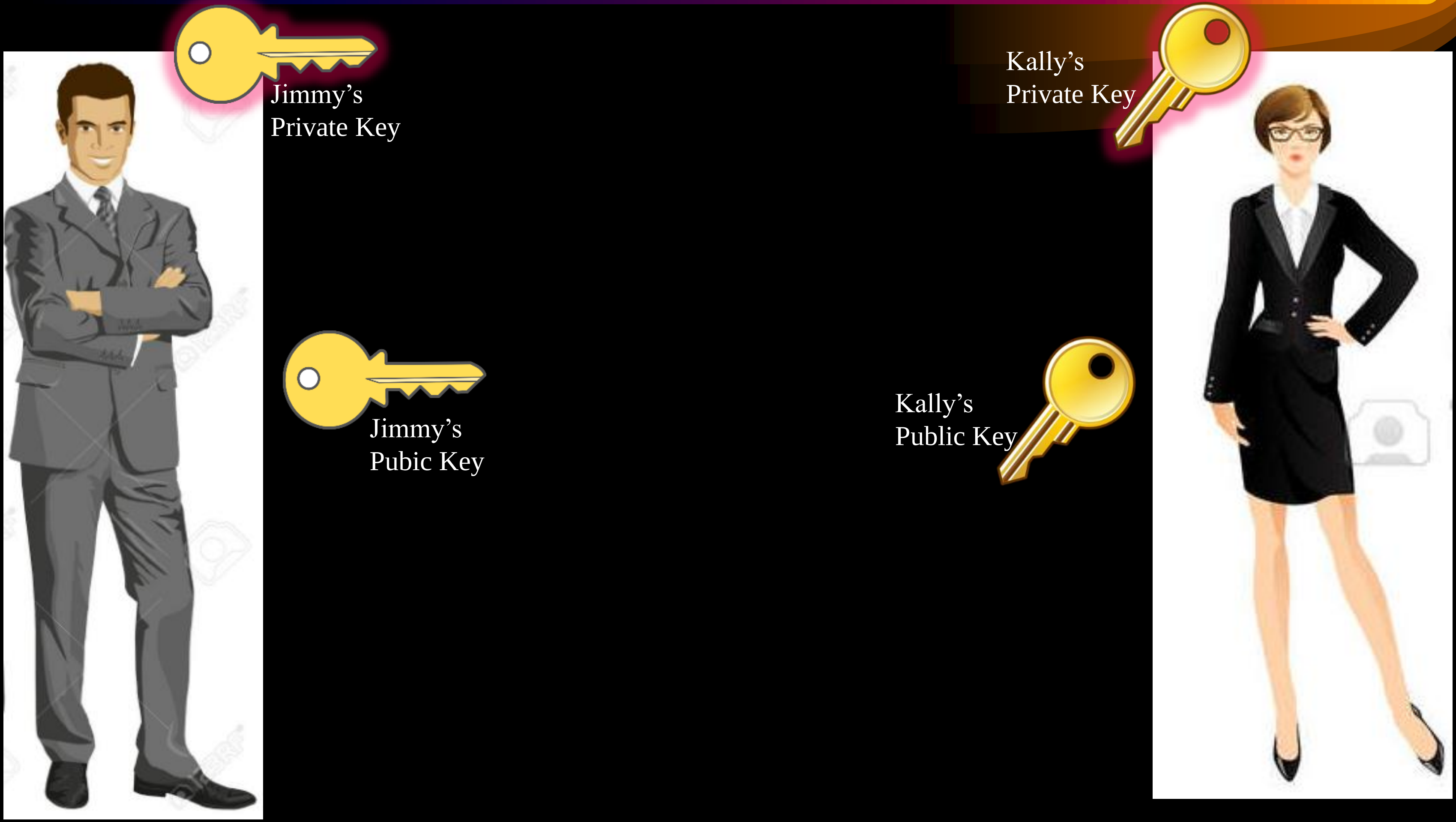


Asymmetric Encryption

- **Asymmetric Cryptography**:
  - Is also known as **public key cryptography**.
  - Uses **two keys** instead of one.
  - Public key systems typically work using difficult math problems known as **trapdoor functions**.

- Some of the popular asymmetric protocols are:
  - RSA, Diffie-Hellman, ECC, and ElGamal

2 keys



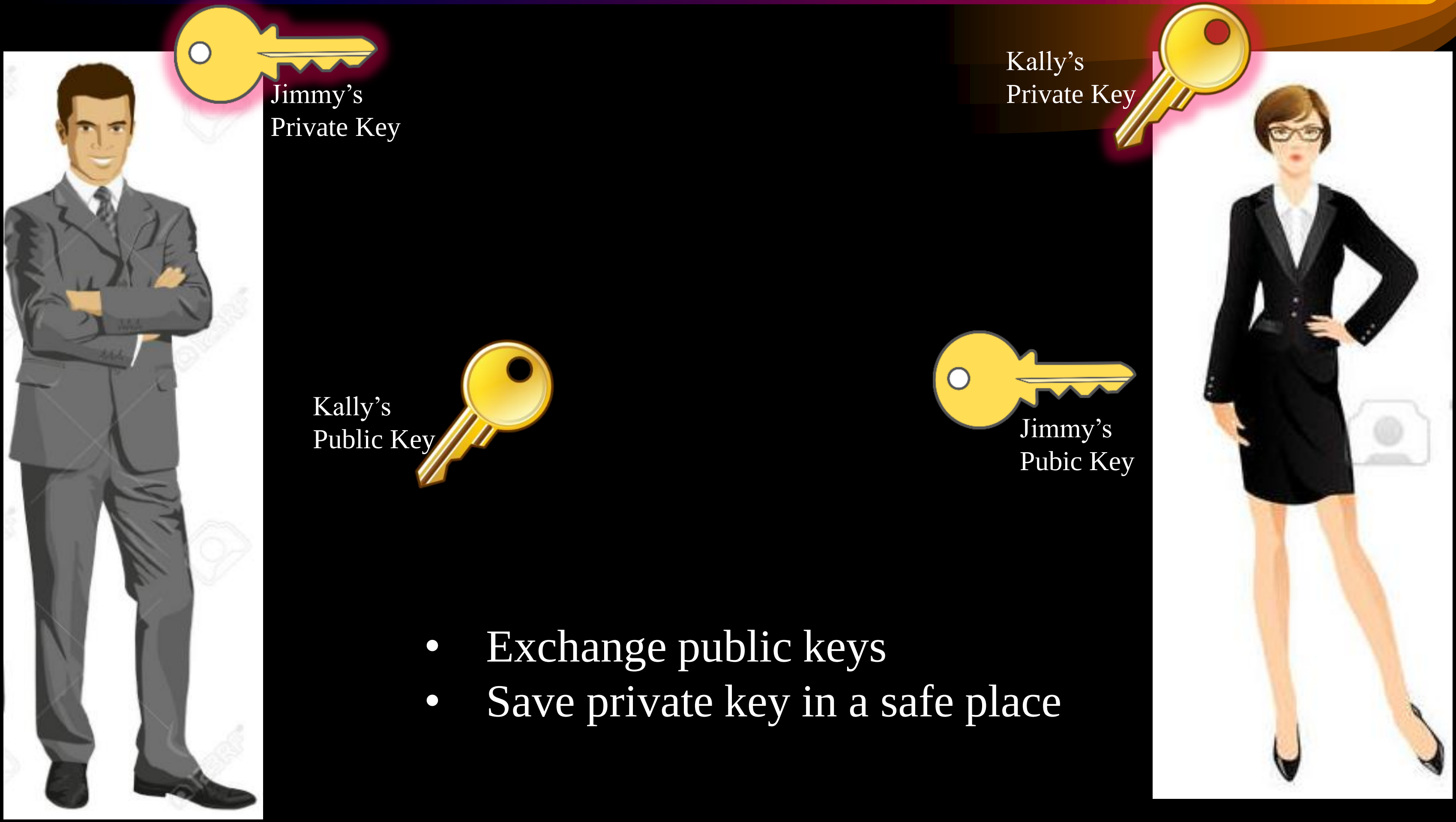
Jimmy's Private Key

Jimmy's Pubic Key

Kally's Private Key

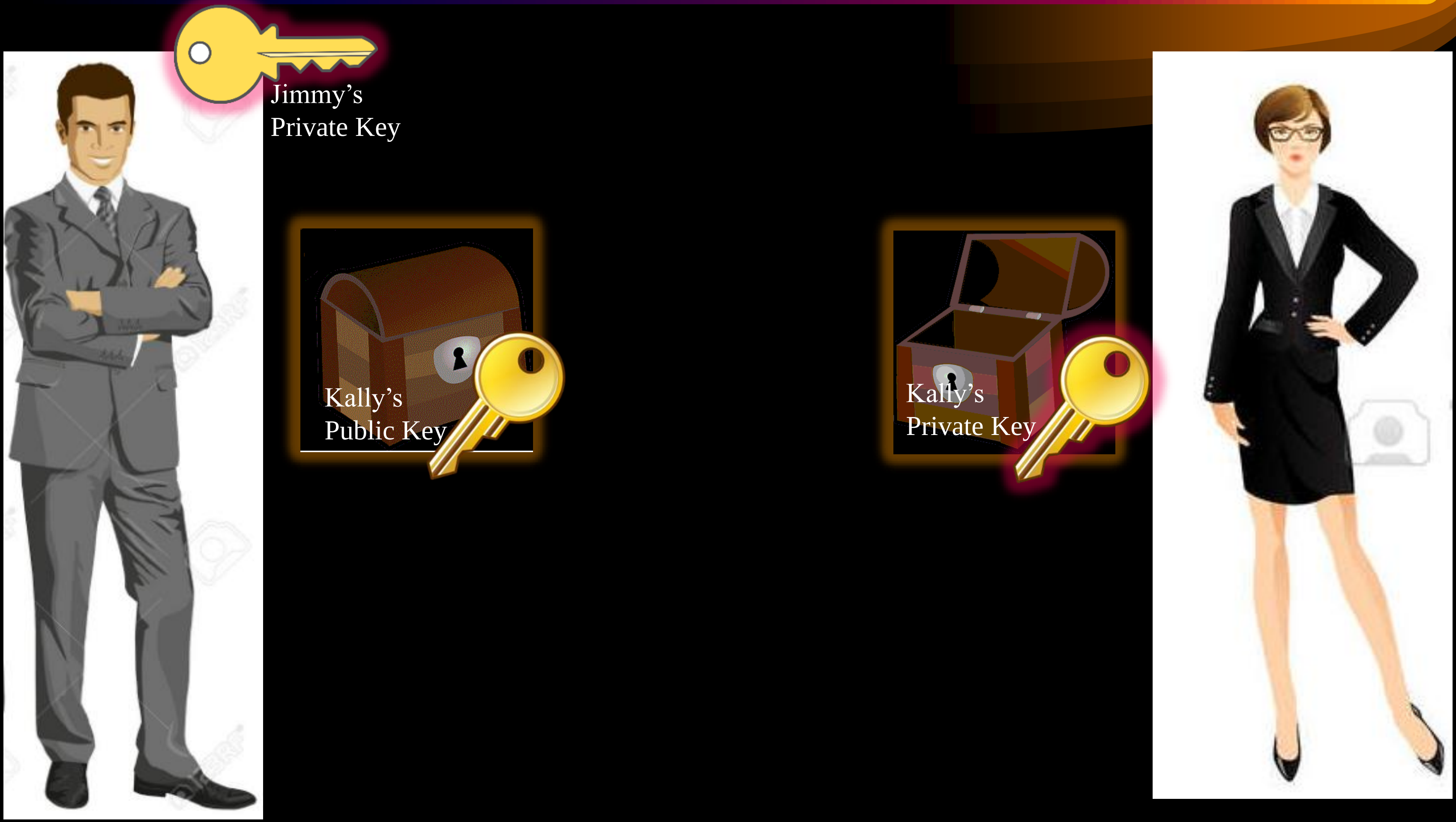
Kally's Public Key

2 keys



- Exchange public keys
- Save private key in a safe place

2 keys



Jimmy's Private Key

Kally's Public Key

Kally's Private Key

RSA

- **RSA** is one of the first public key cryptosystems invented.
  - It can be used for both **encryption** and **digital signatures**.

- This algorithm uses the **product of two very large prime numbers** (from 100 to 200 digits) to generate one key for decryption and another for encryption.

- RSA's security has withstood the test of over 20 years of analysis, but in software it can be **100 times slower than DES**.

Symmetric versus Asymmetric Algorithms

Algorithm Types	Description
Symmetric	■ Uses ONE key to: ❑ **Encrypt data** ❑ **Decrypt data** ■ Is fast & efficient
Asymmetric	■ Uses TWO related keys: ❑ **Public key to encrypt data** ❑ **Private key to decrypt data** ❑ **OR vice versa** ■ Is more secure than symmetric encryption ■ Is slower than symmetric encryption

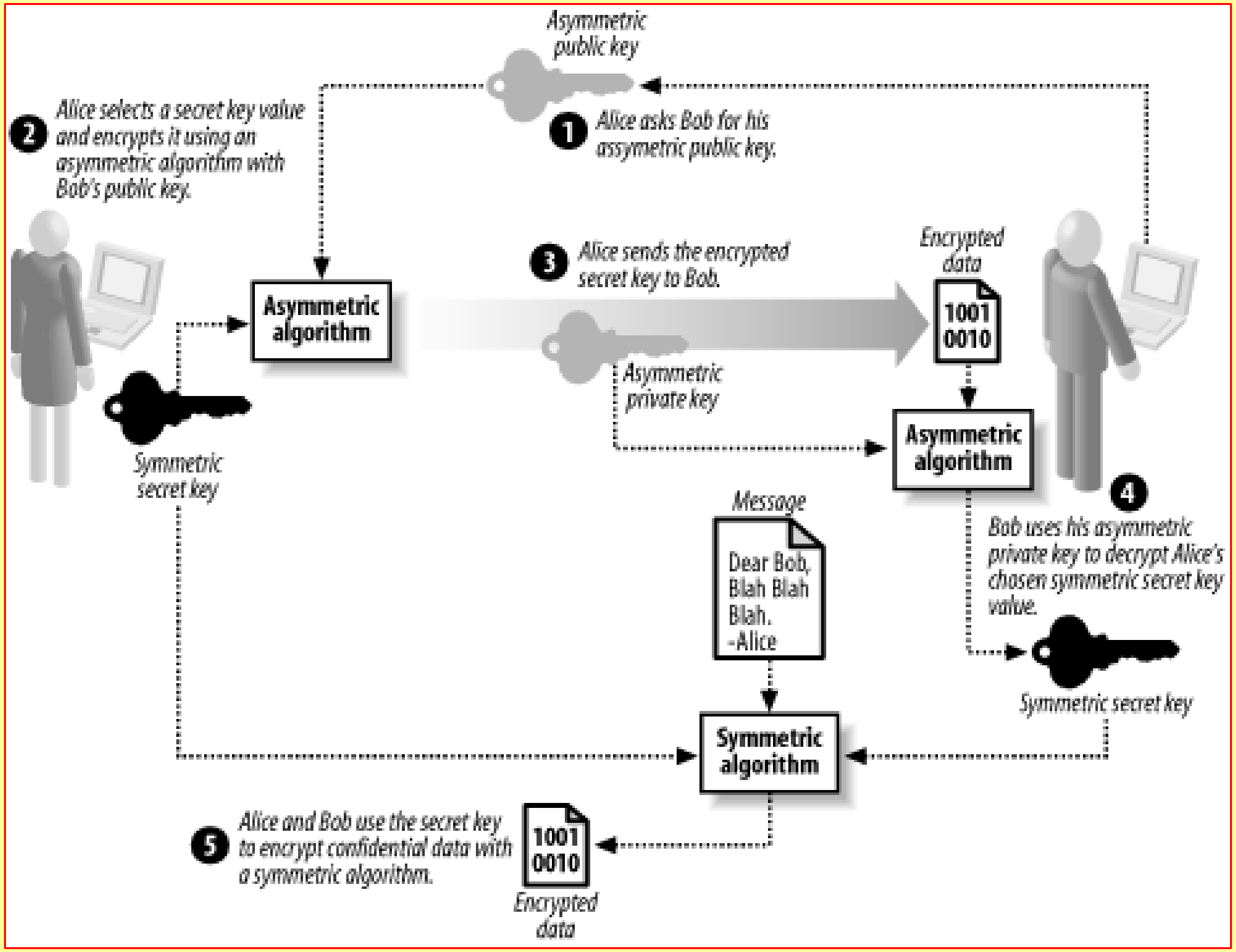
Thinker

- Asymmetric Algorithm not suitable for large file size (slow) but is secure.
- Symmetric Algorithm is fast and suitable for large file size but not secure!
- Can we combine them ?!

Electronic Key Exchange

- Public (Asymmetric) key, the slower protocol, is used to exchange the private key, and then the communication uses the faster symmetric key protocol.
- This process is known as **electronic key exchange**.

Electronic Key Exchange



1. Alice asks Bob for his assymetric public key.

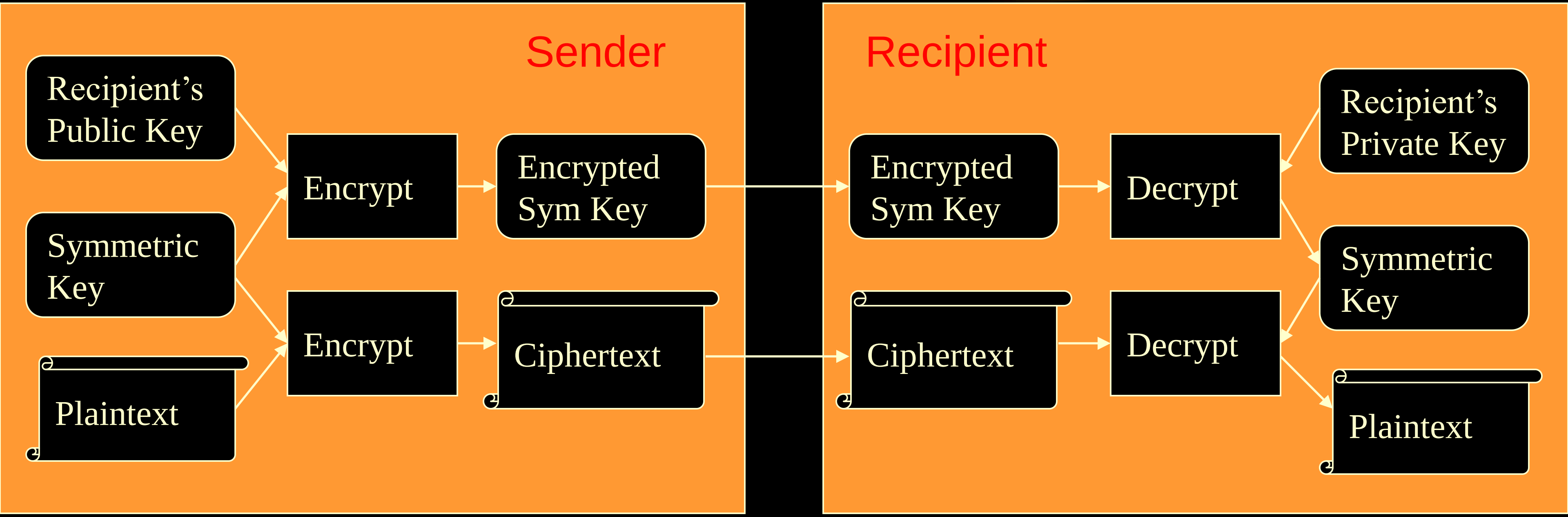
2. Alice selects a secret key value and encrypts it using an asymmetric algorithm with Bob's public key.

3. Alice sends the encrypted secret key to Bob.

4. Bob uses his asymmetric private key to decrypt Alice's chosen symmetric secret key value.

5. Alice and Bob use the secret key to encrypt confidential data with a symmetric algorithm.

Electronic Key Exchange



Sender

Recipient's Public Key

Symmetric Key

Plaintext

Encrypt

Encrypt

Encrypted Sym Key

Ciphertext

Recipient

Encrypted Sym Key

Ciphertext

Decrypt

Decrypt

Recipient's Private Key

Symmetric Key

Plaintext

Diffie-Hellman

- **Diffie-Hellman**:
  - Used in the electronic key exchange method of the Secure Sockets Layer (**SSL**) protocol.
  - Used by the **SSH and IPsec** protocols.
  - Enables the **sharing of a secret key** between two people who have not contacted each other before.

Diffie-Hellman

- The protocol, like RSA, **uses large prime numbers to work**.

- It is very effective because it protects a temporary, automatically generated secret key that is only good for a single communication session.

Asymmetric Summary

- Asymmetric encryption creates the possibility of **digital signatures.**

- It **corrects weakness of symmetric cryptography**.

Usage of Cryptography

- Confidentiality
- Integrity
- Nonrepudiation
- Authentication
- Digital signature

Confidentiality

- **Confidentiality** is the ability to keep secret some piece of data.
- **Symmetric encryption** is favored to store and transmit data.
- **Asymmetric cryptography** does protect confidentiality. Size and speed make it more efficient at protecting the confidentiality of small units such as for electronic key exchange.

Integrity

- When a message is sent, both the sender and the recipient need to know that the message was not altered in transmission.

- This **integrity** is provided with one-way **hash** functions and **digital signatures**.

- A hash value is combined with asymmetric cryptography by taking the message's hash value and encrypting it with the user's private key. [**creating digital signature**]

- Anyone with the user's public key decrypts the hash and compares it to the locally computed hash. [**verifying digital signature**]

Nonrepudiation

- **Nonrepudiation** means that the senders cannot later deny that they sent the message.
- It is based upon public key cryptography and the principle of only you knowing your private key. [**refer to integrity slide**]

Authentication

- **Authentication** is being able to prove one’s identity.

- Authentication can be done by a **password, token, or biometric**.

- **Digital certificates** are one form of such tokens.

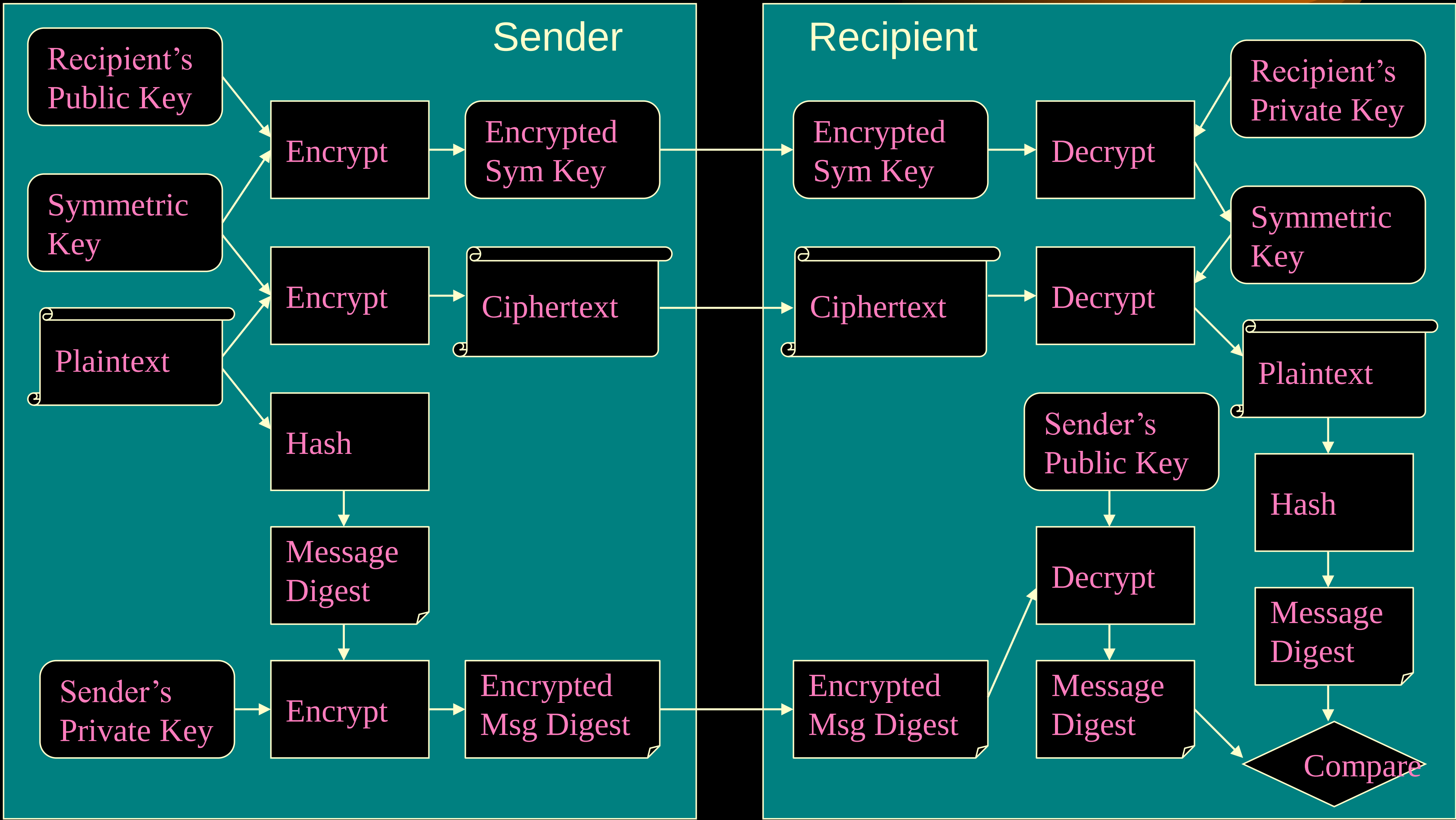
- **Asymmetric encryption** is better suited than symmetric encryption to prove one’s identity.

Digital Signatures

- **Digital signatures** are based upon both hashing functions and asymmetric cryptography.

- Hashing functions are used to create a digest of a unique message and easily reproducible by both parties. This ensures that the message integrity is complete. [**refer to integrity slide**]

Digital Signature Operation



Sender

Recipient's Public Key

Symmetric Key

Plaintext

Encrypt

Encrypt

Hash

Message Digest

Sender's Private Key

Encrypt

Encrypted Sym Key

Ciphertext

Encrypted Msg Digest

Recipient

Encrypted Sym Key

Ciphertext

Encrypted Msg Digest

Decrypt

Decrypt

Sender's Public Key

Decrypt

Message Digest

Recipient's Private Key

Symmetric Key

Plaintext

Hash

Message Digest

Compare

Summary

- Algorithms
- Hashing
- Symmetric Encryption
- Asymmetric Encryption
- Usage