

Lecture 11: Overview of Public-Key Cryptography

Yuan Xue

In this lecture, we give an overview to the *public-key* cryptography, which is also referred to as *asymmetric* cryptography. We will first introduce the background of public-key cryptography. Then we will study the model of public-key cryptosystem, and the requirement to design a practical public-key algorithm.

I. BACKGROUND

The concept of public-key (asymmetric) cryptography evolved from an attempt to address the following two difficult problems associated with the secret-key (symmetric) cryptosystem.

- *Key distribution.* Symmetric encryption requires a shared secret key. As we have seen in the previous lecture, this leads to the problem of key distribution, which in turn requires secret sharing either between the communicating parties or between the communicating hosts and the KDC.
- *Non-repudiation.* In symmetric encryption, the sending party may deny sending the message, because it is possible for the receiver to forge a message based on the same secret key. Accordingly, the receiving party may claim the receipt of a message from the sender which is actually forged by himself.

To address both problems, Diffie and Hellman achieved an important breakthrough in 1976. The proposed scheme was radically different from all previous approaches to cryptography. First, it uses a pair of different keys in contrast to one shared key in symmetric encryption. Second, it is based on mathematical functions instead of substitution and permutation. The proposed scheme is called public-key (asymmetric) cryptography, which is one of the greatest revolutions in the history of cryptography. Its use of two keys has profound consequences in facilitating key distribution, and providing digital signature.

However, the scheme proposed by Diffie and Hellman is not a general-purpose encryption algorithm. It can only provide secure secret key exchange. Thus it presents a challenge for the cryptologists to design a general-purpose encryption algorithm that satisfies the public-key encryption requirements. One of the first responses to the challenge was developed in 1977 by Rivest, Shamir, Adleman at MIT, so called RSA. Since then, the RSA scheme has become the most widely accepted and implemented general-purpose approach to public-key encryption.

II. PUBLIC-KEY CRYPTOSYSTEM MODEL

In this section, we look at the overall framework for public-key cryptography. Public-key cryptosystem uses a pair of different but related keys – one for encryption, the other for decryption; one is placed in a public register (public key), the other is kept secret (private key). It is required that given only knowledge of the cryptographic algorithm and the private key, it is computationally infeasible to determine the private key. In some algorithms, such as RSA, either public key or private key can be used for encryption, with the other one used for decryption. Fig. 1 illustrates that different security services can be provided with different usages of the keys.

In Fig. 1(a), sender A encrypts plaintext P using B's public key K_{UB} ¹. The ciphertext $C = E(K_{UB}, P)$ is transmitted to B. B, in possession of the matching private key K_{RB} , is able to decrypt the ciphertext C and retrieve the plaintext $P = D(K_{RB}, C)$. No one else can decrypt the message without B's private key K_{RB} . Thus the *confidentiality* of incoming communication to B is assured as long as B's private key is kept secret.

In Fig. 1(b), A encrypts P with its private key K_{RA} . The ciphertext $C = E(K_{RA}, P)$ received by B can be decrypted using A's public key K_{UA} : $P = D(K_{UA}, C)$. Without A's private key, it is impossible to change the message. Therefore, the public-key encryption provides authentication in terms of both source and data integrity. At the same time, the entire message serves as a digital signature. This is because the message was encrypted using A's private key, and only A could have prepared the message. It is important to emphasize that this encryption process does not provide confidentiality. This is, the message is safe from alteration but not from eavesdropping. This is because any observer can decrypt the message by using A's public key.

To provide both authentication and confidentiality for data delivery, a double use of the public-key encryption can be applied as shown in Fig. 1(c). In this scenario, the message is first encrypted by A's private key, which provides authentication. Then B's public key is applied again to provide confidentiality. At the receiver side, B's private key will first be used for decryption, followed by the decryption using A's public key.

The essential steps to use public-key encryption are summarized as follows.

- 1) Generate a pair of keys. For example, A generates the public key K_{UA} , and the private key K_{RA} .
- 2) Publish the public key *e.g.*, K_{UA} , while keeping the private key secret. Users have the access to a collection of public keys from their communication parties.

¹As a convention, K_U is used for public key; K_R is used for private key.

- 3) Use one of the above models to encrypt the message to achieve different security goals and deliver the message. The received message is decrypted using the corresponding schemes in the models.

Based on the public-key encryption framework, the following requirements need to be satisfied to design a public-key encryption algorithm.

- 1) It is computationally infeasible for an opponent, knowing the public key K_U , and the encryption and decryption algorithms E , D , to determine the companion private key K_R .
- 2) It is computationally infeasible for an opponent, knowing the public key K_U and the ciphertext C which is encrypted via this key $C = E(K_U, P)$, to determine the plaintext P .

For practical use, the following features are also preferred in a public-key encryption algorithm.

- 1) It is computationally easy to generate a pair of keys (public key and private key).
- 2) It is computationally easy to encrypt a message using either public or private key, and decrypt it via the companion key.

In the next two lectures, we will first examine the RSA algorithm, which is the most important algorithm that is used for public-key encryption. Next we will study Diffie-Hellman algorithm, and show how it can be used for the distribution of secret keys. We will also examine the distribution of public keys in this lecture.

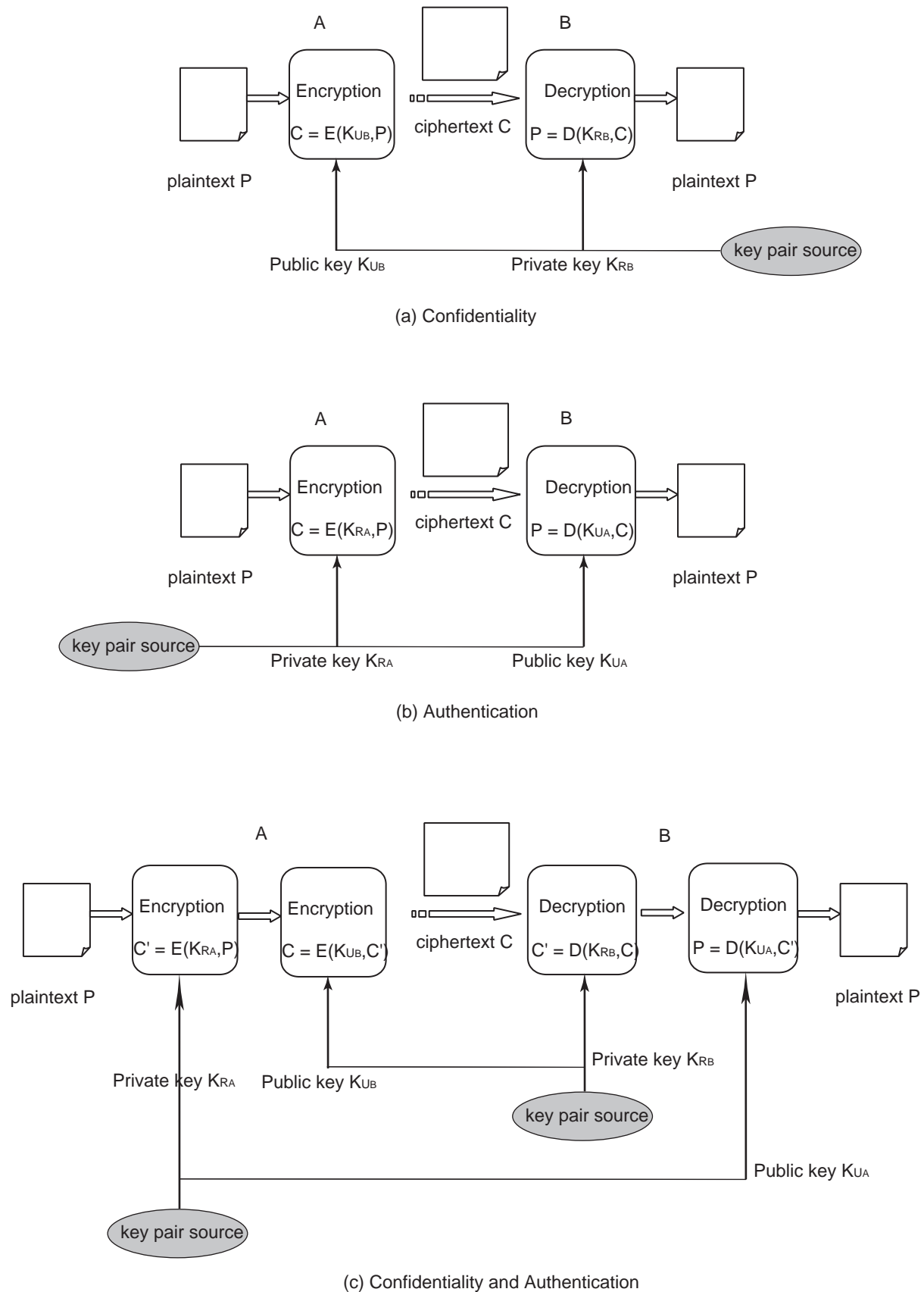


Fig. 1. Asymmetric Cryptosystem Models.