The Islamic University
College of Technical Engineering
Department of Computer Technical Engineering

Fourth Stage

Security

Lecture 16

Asst. Lec. Yousif Samer Mudhafar

Email: yousif.samir19@gmail.com

Diffie Hellman

- The **Diffie–Hellman (DH) key exchange** technique was first defined in their paper in 1976.
- **DH** key exchange is a method of exchanging public i.e.(non-secret) information to obtain a shared secret.
- **DH is not an encryption algorithm.**
- DH key exchange has the following important properties:

1. The resulting shared secret cannot be computed by either of the parties without the cooperation of the other.
2. A third party observing all the messages transmitted during DH key exchange cannot deduce the resulting shared secret at the end of the protocol.

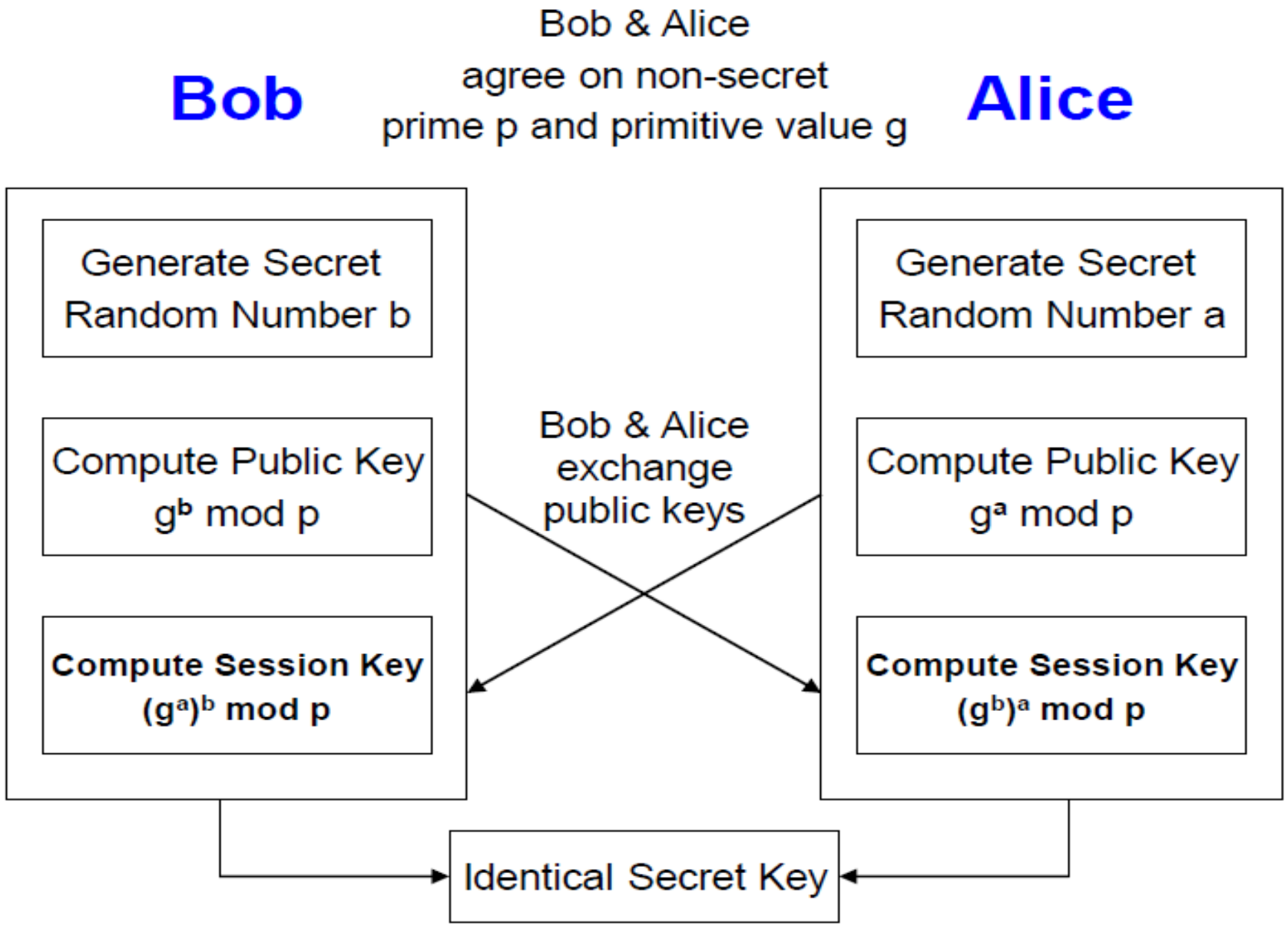
Diffie-Hellman Key Exchange

The system parameters (which are public) are:

- **a large prime number p –typically 1024 bits in length.**
- **a primitive root (g) to p.**

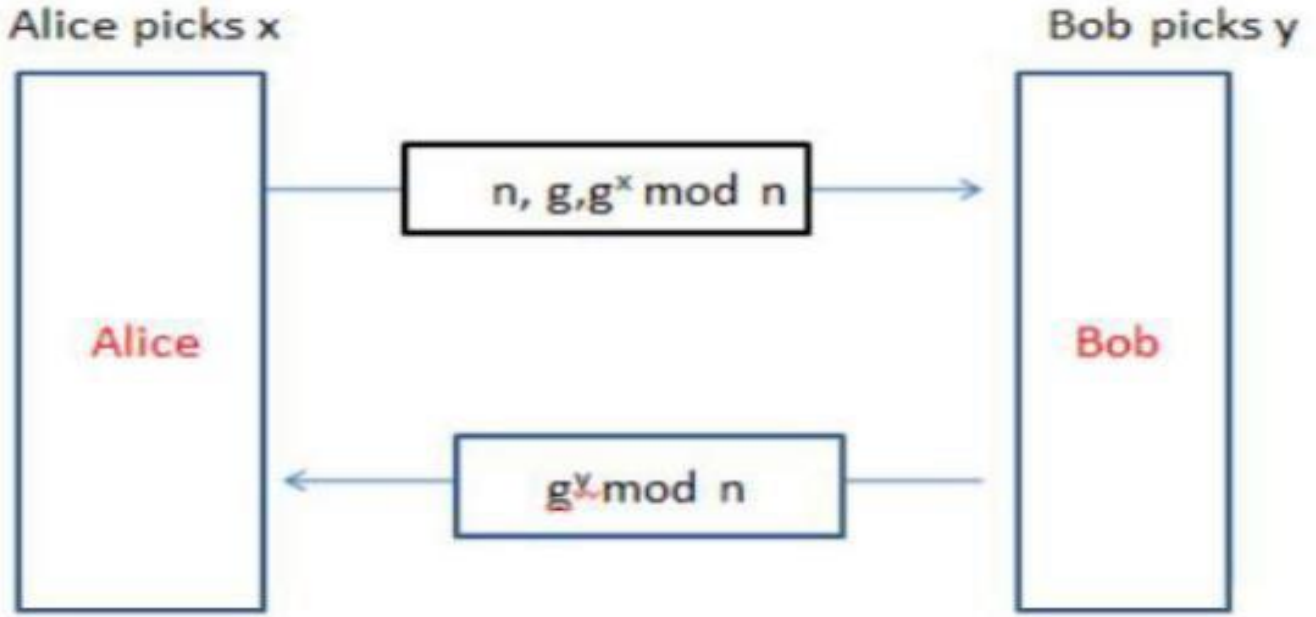
1. Alice generates a private random value **a**, calculates $g^a \pmod p$ and sends it to Bob. Meanwhile Bob generates a private random value **b**, calculates $g^b \pmod p$ and sends it to Alice.
2. Alice takes g^b and her private random value **a** to compute $(g^b)^a = g^{ab} \pmod p$.
3. Bob takes g^a and his private random value **b** to compute $(g^a)^b = g^{ab} \pmod p$.
4. Alice and Bob adopt $g^{ab} \pmod p$ as the shared secret.

Diffie-Hellman Mathematical Analysis



Diffie-Hellman Key Exchange

The result is that the two sides have exchanged a secret value as shown in Figure below. The security of the Diffie-Hellman key exchange lies in the fact that, while it is relatively easy to calculate exponentials modulo a prime, it is very difficult to calculate discrete logarithms. For large primes, the latter task is considered infeasible.



Alice Computes

$$(g^Y \bmod n)^X \bmod n$$

$$K = g^{XY} \bmod n$$

Bob Computes

$$(g^X \bmod n)^Y \bmod n$$

$$K = g^{XY} \bmod n$$

Figure 1: The Diffie-Hellman Key Exchange Algorithm.

Diffie-Hellman Key Exchange

Here is an example. Key exchange is based on the use of the prime number $q = 353$ and a primitive root of 353, in this case $\alpha = 3$. A and B select secret keys $X_A = 97$ and $X_B = 233$, respectively. Each computes its public key:

$$\text{A computes } Y_A = 3^{97} \bmod 353 = 40.$$

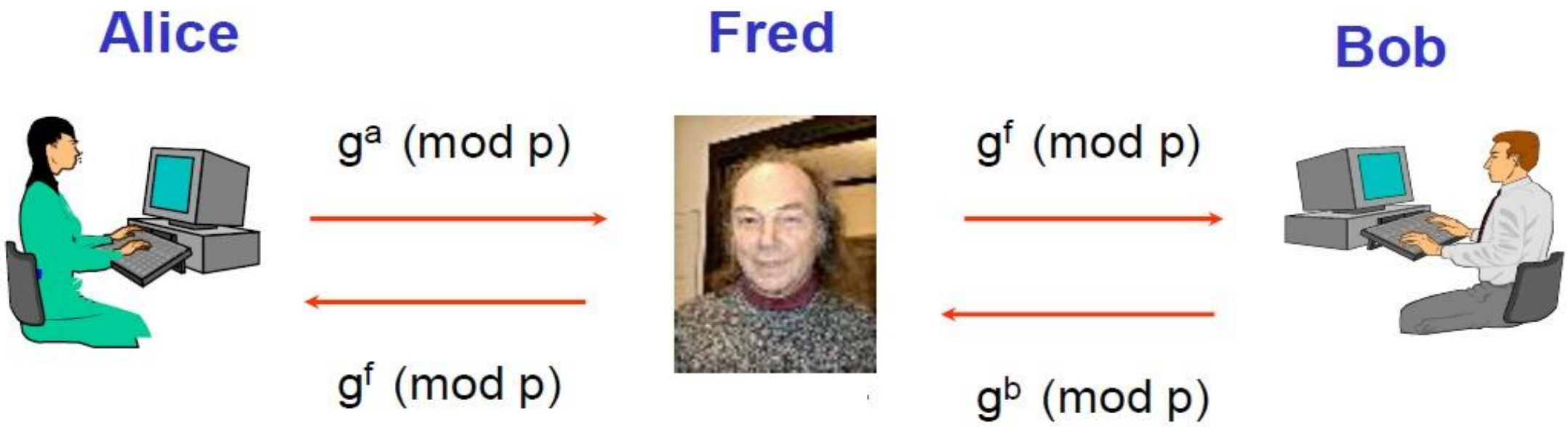
$$\text{B computes } Y_B = 3^{233} \bmod 353 = 248.$$

After they exchange public keys, each can compute the common secret key:

$$\text{A computes } K = (Y_B)^{X_A} \bmod 353 = 248^{97} \bmod 353 = 160.$$

$$\text{B computes } K = (Y_A)^{X_B} \bmod 353 = 40^{233} \bmod 353 = 160.$$

Man-in-the-middle attack



1. What will happen when Alice tries to send a message to Bob, encrypted with a key based on her DH shared secret?
2. Can Fred obtain the correct DH shared secret that would have been established had he not interfered?

Man-in-the-middle attack

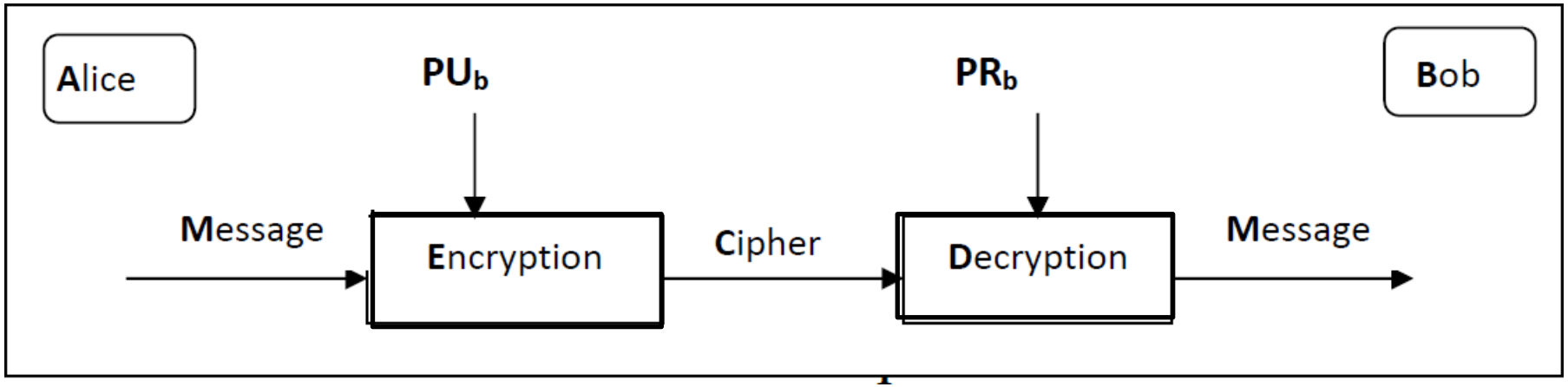
"The Diffie-Hellman key exchange is vulnerable to a man-in-the-middle attack. In this attack, **an opponent Carol intercepts Alice's public value and sends her own public value to Bob**. When Bob transmits his public value, Carol substitutes it with her own and sends it to Alice.

Public Key Cryptography

The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems associated with symmetric encryption. ***The first problem*** is that of key distribution. Key distribution under symmetric encryption requires either (1) that two communicants already share a key, which somehow has been distributed to them; or (2) the use of a key distribution center.

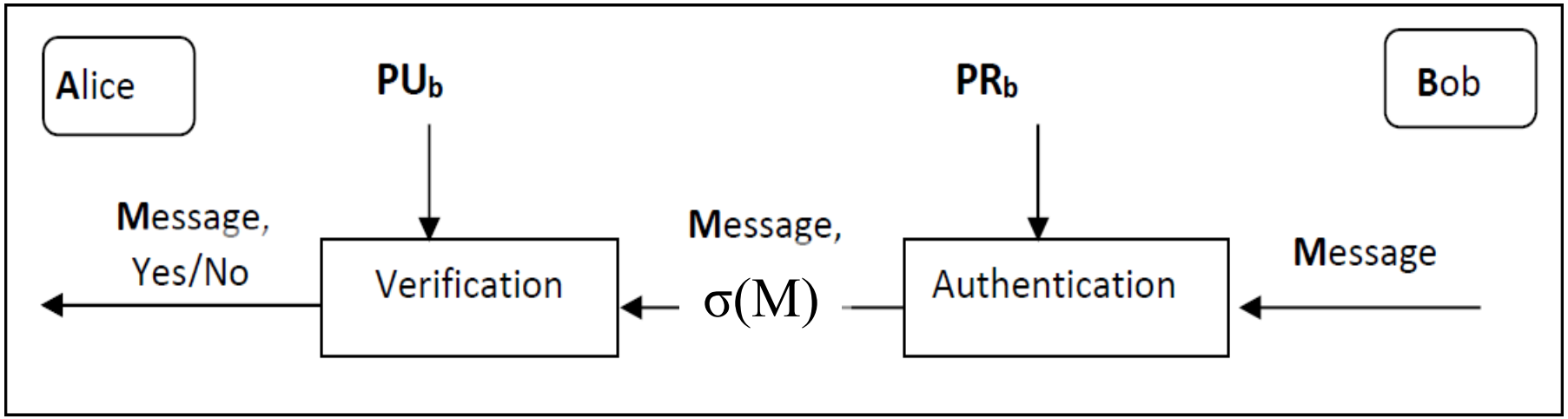
The second is the authentication the way that identifies the sender.

The first use of public key cryptography is for **encrypting messages** to Bob. Anyone who wishes to send an encrypted message to Bob will use Bob's public key. To decrypt the message, Bob's private key is needed, and only Bob knows it.



Public Key Cryptography

The second use of public key cryptography is **for message signing** (***authentication)*** by Bob. Bob's private key is used by Bob to generate a signature. Any one is able to verify Bob's signature using Bob's public key. Denote the signature of the message M by σ(M).



Private-Key and Public-Key

The table below summarizes some of the important aspects of symmetric (Private-Key Cryptography) and public-key encryption. To discriminate between the two, we refer to the key used in symmetric encryption as a secret key (Private-Key Cryptography) and asymmetric (Public-Key Cryptography).

Private-Key Cryptography	Public-Key Cryptography
• The same algorithm with the same key is used for encryption and decryption • Key is shared by both sender and receiver • Also known as symmetric, both parties are equal • Provide **secrecy** • For example DES cipher algorithm.	• One algorithm is used for encryption and decryption with a pair of keys, one for encryption and one for decryption. • sender and receiver used different keys • Asymmetric since parties are not equal • Provide **secrecy**, **authentication** and **session keys**. • For example RSA cipher algorithm.

Simple Secret Key Distribution

An extremely simple scheme was put forward by Merkle, as illustrated in the following Figure . If A wishes to communicate with B, the following procedure is employed:

1. A generates a public/private key pair {PUa, PRa} and transmits a message to B consisting of PUa and an identifier of A, IDA.

2. B generates a secret key, Ks, and transmits it to A, encrypted with A's public key.

3. A computes D(PRa, E(PUa, Ks)) to recover the secret key. Because only A can decrypt the message, only A and B will know the identity of Ks.

4. A discards PUa and PRa and B discards PUa.