

13.1 DIGITAL SIGNATURES

5a)

Properties

Message authentication protects two parties who exchange messages from any third party. However, it does not protect the two parties against each other. Several forms of dispute between the two are possible.

I

Bob

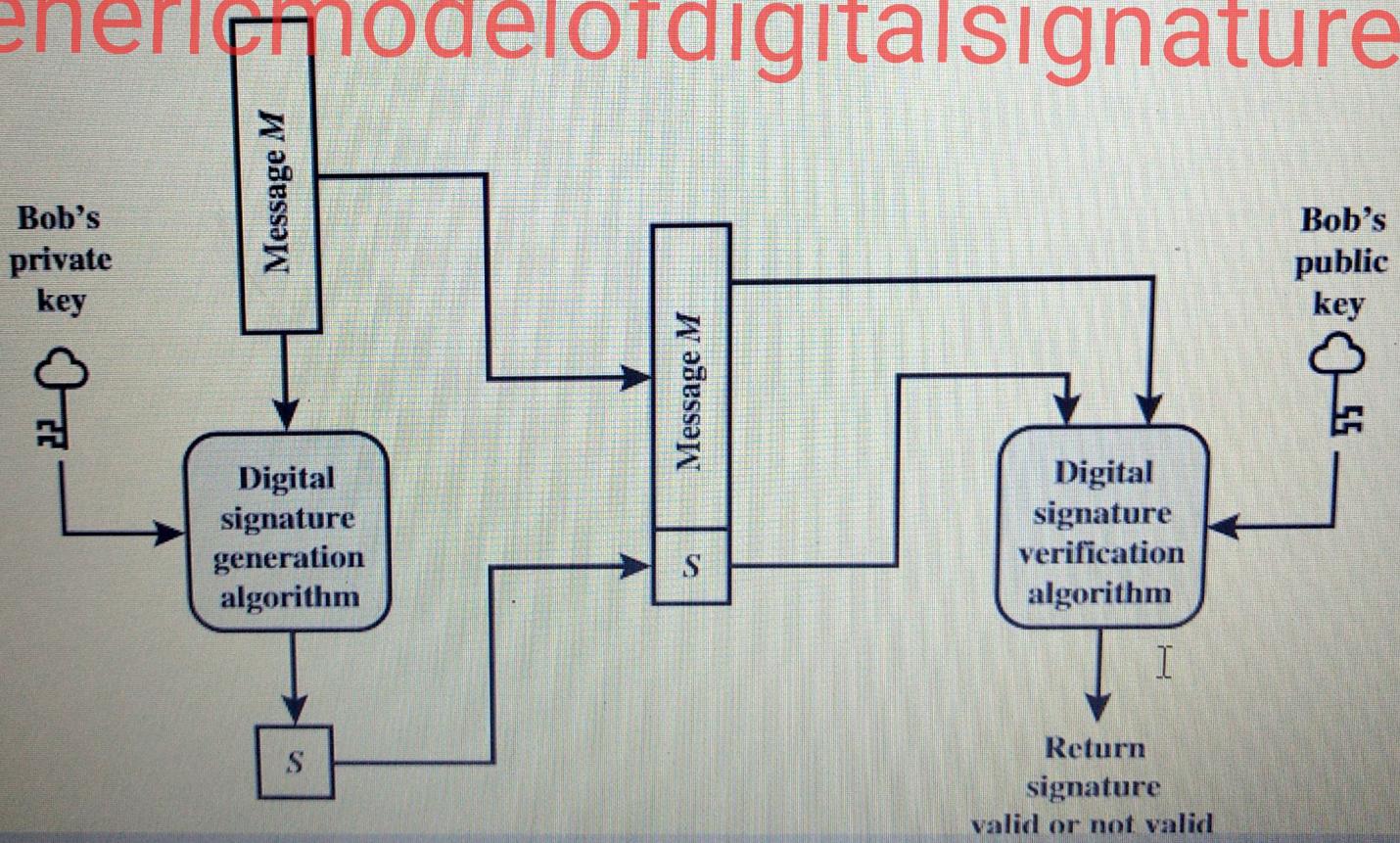


Transmit

Alice



generic model of digital signature



Bob's
signature
for M

Figure 13.1 Generic Model of Digital Signature Process

For example, suppose that John sends an authenticated message to Mary, using one of the schemes of Figure 12.1. Consider the following disputes that could arise.

1. Mary may forge a different message and claim that it came from John. Mary would simply have to create a message and append an authentication code using the key that John and Mary share.
2. John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.

Both scenarios are of legitimate concern. Here is an example of the first scenario: An electronic funds transfer takes place, and the receiver increases the amount of funds transferred and claims that the larger amount had arrived from the sender. An example of the second scenario is that an electronic mail message contains instructions to a stockbroker for a transaction that subsequently turns out badly. The sender pretends that the message was never sent.

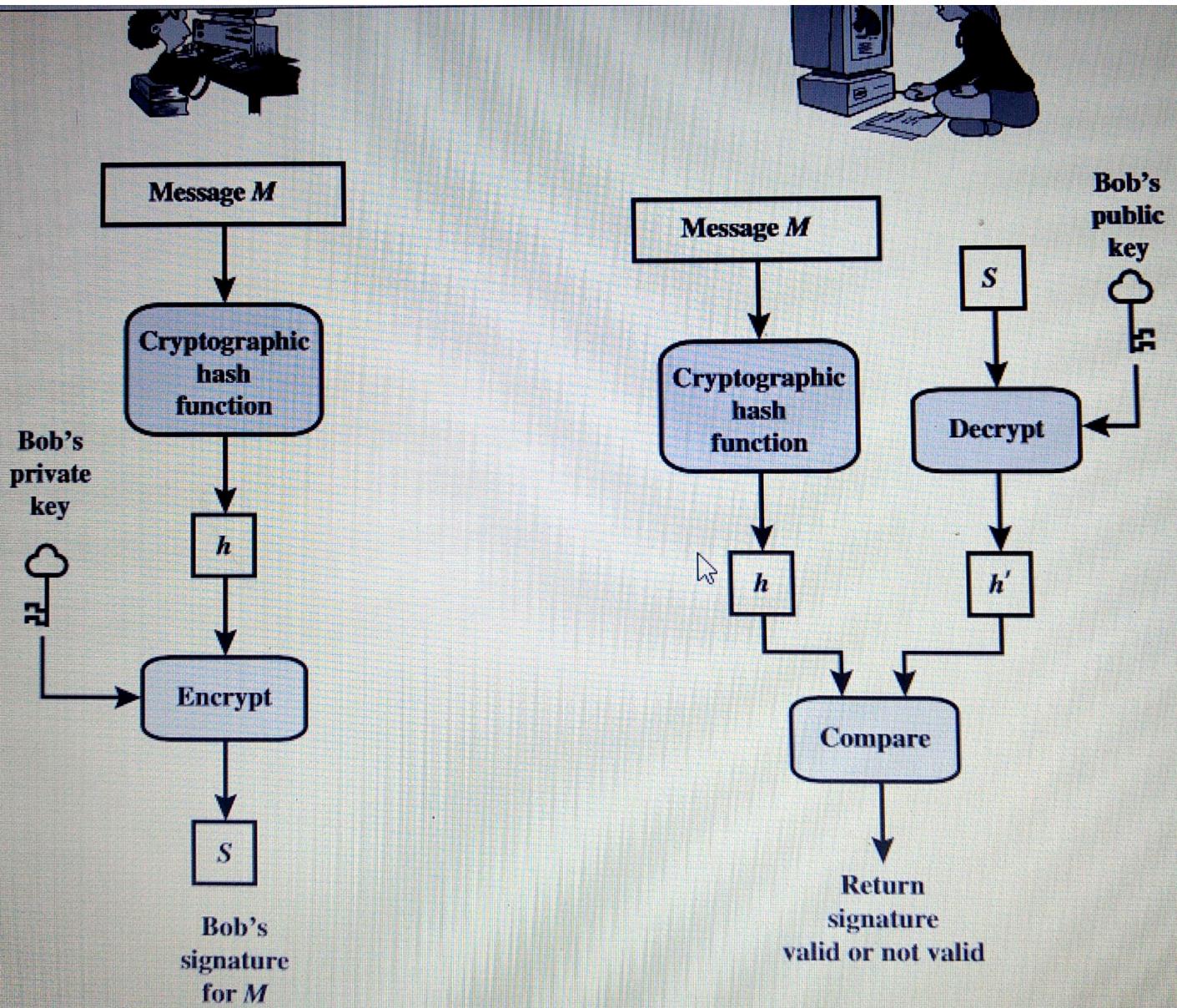




Figure 13.2 Simplified Depiction of Essential Elements of Digital Signature Process

In situations where there is not complete trust between sender and receiver, something more than authentication is needed. The most attractive solution to this problem is the digital signature. The digital signature must have the following properties:

- It must verify the author and the date and time of the signature.
- It must authenticate the contents at the time of the signature.
- It must be verifiable by third parties, to resolve disputes.

Thus, the digital signature function includes the authentication function.

Attacks and Forgeries

[GOLD88] lists the following types of attacks, in order of increasing severity. Here A denotes the user whose signature method is being attacked, and C denotes the attacker.

- **Key-only attack:** C only knows A's public key.
- **Known message attack:** C is given access to a set of messages and their signatures.
- **Generic chosen message attack:** C chooses a list of messages before attempting to break A's signature scheme, independent of A's public key. C then obtains from A valid signatures for the chosen messages. The attack is generic, because it does not depend on A's public key; the same attack is used against everyone.
- **Directed chosen message attack:** Similar to the generic attack, except that the list of messages to be signed is chosen after C knows A's public key but before any signatures are seen.
- **Adaptive chosen message attack:** C is allowed to use A as an "oracle." This means the A may request signatures of messages that depend on previously obtained message-signature pairs.

[GOLD88] then defines success at breaking a signature scheme as an outcome in which C can do any of the following with a non-negligible probability:

- **Total break:** C determines A's private key.
- **Universal forgery:** C finds an efficient signing algorithm that provides an equivalent way of constructing signatures on arbitrary messages.
- **Selective forgery:** C forges a signature for a particular message chosen by C.

13.2 ELGAMAL DIGITAL SIGNATURE SCHEME

5b)

Before examining the NIST Digital Signature standard, it will be helpful to understand the ElGamal and Schnorr signature schemes. Recall from Chapter 10, that the ElGamal encryption scheme is designed to enable encryption by a user's public key with decryption by the user's private key. The ElGamal signature scheme involves the use of the private key for encryption and the public key for decryption [ELGA84, ELGA85].

Before proceeding, we need a result from number theory. Recall from Chapter 8 that for a prime number q , if α is a primitive root of q , then

$$\alpha, \alpha^2, \dots, \alpha^{q-1}$$

are distinct $(\bmod q)$. It can be shown that, if α is a primitive root of q , then

1. For any integer m , $\alpha^m \equiv 1 (\bmod q)$ if and only if $m \equiv 0 (\bmod q - 1)$.
2. For any integers, i, j , $\alpha^i \equiv \alpha^j (\bmod q)$ if and only if $i \equiv j (\bmod q - 1)$.

13.2 / ELGAMAL DIGITAL SIGNATURE SCHEME 401

As with ElGamal encryption, the global elements of **ElGamal digital signature** are a prime number q and α , which is a primitive root of q . User A generates a private/public key pair as follows.

1. Generate a random integer X_A , such that $1 < X_A < q - 1$.
2. Compute $Y_A = \alpha^{X_A} \bmod q$.
3. A's private key is X_A ; A's public key is $\{q, \alpha, Y_A\}$.

To sign a message M , user A first computes the hash $m = H(M)$, such that m is an integer in the range $0 \leq m \leq q - 1$. A then forms a digital signature as follows.

1. Choose a random integer K such that $1 \leq K \leq q - 1$ and $\gcd(K, q - 1) = 1$. That is, K is relatively prime to $q - 1$.
2. Compute $S_1 = \alpha^K \bmod q$. Note that this is the same as the computation of C_1 for ElGamal encryption.
3. Compute $K^{-1} \bmod (q - 1)$. That is, compute the inverse of K modulo $q - 1$.
4. Compute $S_2 = K^{-1}(m - X_A S_1) \bmod (q - 1)$.
5. The signature consists of the pair (S_1, S_2) .

Any user B can verify the signature as follows.

1. Compute $V_1 = \alpha^m \bmod q$.
2. Compute $V_2 = (Y_A)^{S_1}(S_1)^{S_2} \bmod q$.

The signature is valid if $V_1 = V_2$. Let us demonstrate that this is so. Assume that the equality is true. Then we have

$\alpha^m \bmod q = (Y_A)^{S_1}(S_1)^{S_2} \bmod q$	assume $V_1 = V_2$
$\alpha^m \bmod q = \alpha^{X_A S_1} \alpha^{K S_2} \bmod q$	substituting for Y_A and S_1
$\alpha^{m - X_A S_1} \bmod q = \alpha^{K S_2} \bmod q$	rearranging terms
$m - X_A S_1 \equiv K S_2 \bmod (q - 1)$	property of primitive roots
$m - X_A S_1 \equiv K K^{-1} (m - X_A S_1) \bmod (q - 1)$	substituting for S_2