Chapter 13

Digital Signatures

Digital Signatures

- The most important development from the work on public-key cryptography is the digital signature.
- The digital signature provides a set of security capabilities that would be difficult to implement in any other way.

Digital Signatures

- Digital signatures and seals are the electronic equivalent of handwritten signatures and seals
- They provide:
  - Authentication (of origin only)
  - Non-repudiation
  - Integrity
- Due to the requirement for non-repudiation only public-key cryptography can be used
  - Signature is tied to the user’s private key

Digital Signatures

- Digital signatures have legal significance in certain jurisdictions
  - They can be more difficult to forge than regular handwritten signatures

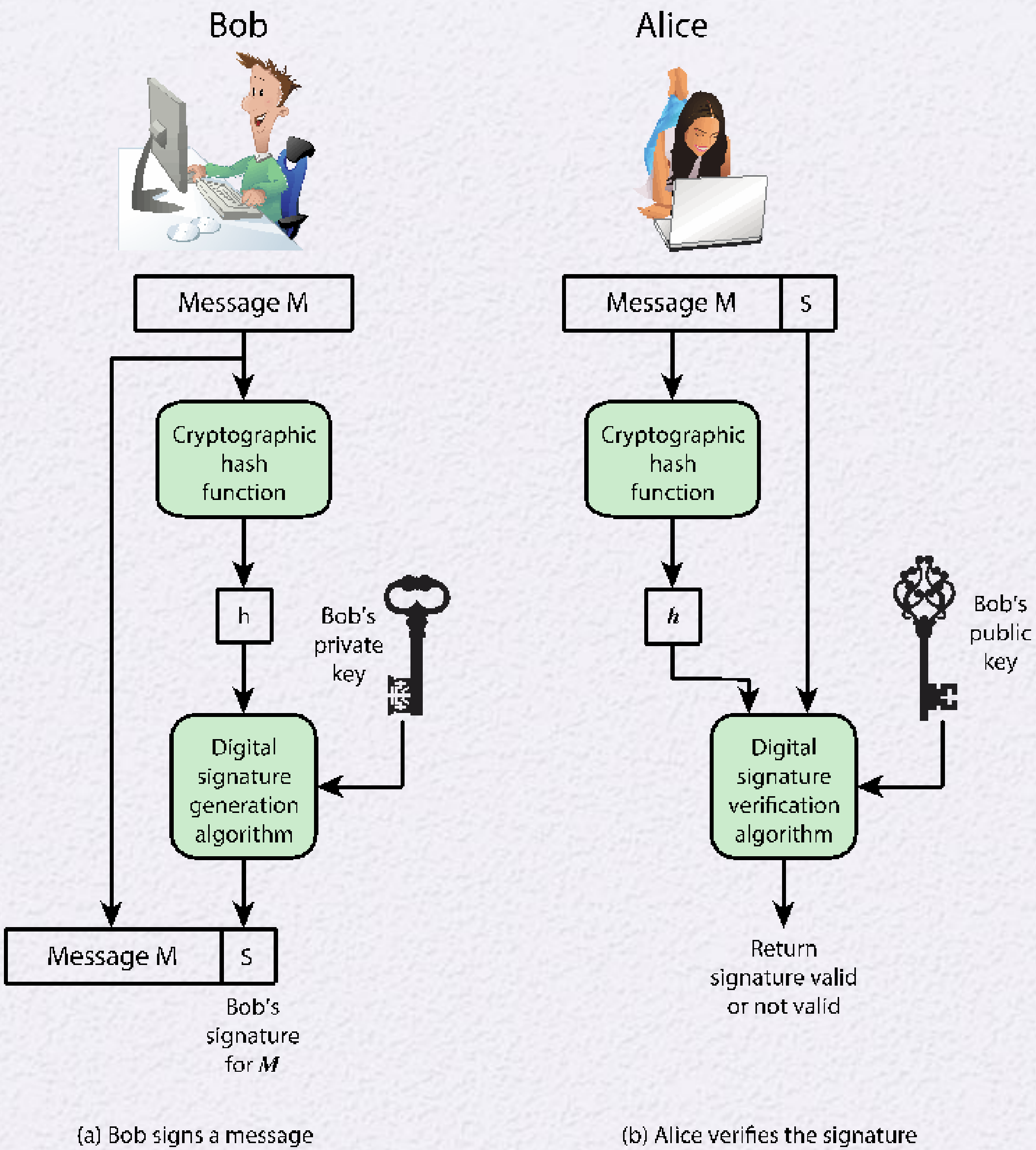


Figure 13.1 Simplified Depiction of Essential Elements of Digital Signature Process

Hash Values

- Apart from security, using the hash value to create the digital signature provides
  - Storage efficiency – the signature is easy to store
  - Computational efficiency – the signature can be computed and verified quickly
  - Compatibility – the signature scheme might require a fixed length input

Digital Signature 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

Attacks

- The goal of an attack against a digital signature is to create a forgery
  - Forge a signature for a message
  - Forge a message that matches a signature

Attacks

A: Victim
C: 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

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 may request from A signatures of messages that depend on previously obtained message-signature pairs

Forgeries

A: Victim
C: Attacker

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

Existential forgery

- C forges a signature for at least one message; C has no control over the message

Digital Signature Requirements

- The signature must be a bit pattern that depends on the message being signed
- The signature must use some information **unique to the sender** to prevent both forgery and denial
- It must be relatively easy to produce the digital signature
- It must be relatively easy to recognize and verify the digital signature
- It must be computationally infeasible to forge a digital signature, either by constructing a new message for an existing digital signature or by constructing a fraudulent digital signature for a given message
- It must be practical to retain a copy of the digital signature in storage

Digital Signature Notions

- While there are many formal definitions for the security of digital signature. The two most common ones you will encounter are:

1. EUF-CMA
    - Existential Unforgeability-Under Chosen Message Attack
    - The attacker's goal is to produce a valid signature for a message that they have not seen before, and the attacker is allowed to make multiple queries to the signer.
2. SeUF-CMA
    - Strong Existential Unforgeability-Under Chosen Message Attack
    - This ensures that not only can the attacker not create a new (message, signature) pair, but they also cannot create a different valid signature for a message that has already been signed.

NIST Digital Signature Algorithm

- Published by NIST as Federal Information Processing Standard FIPS 186
- Makes use of the Secure Hash Algorithm (SHA)
- The latest version, FIPS 186-3, also incorporates digital signature algorithms based on RSA and on elliptic curve cryptography

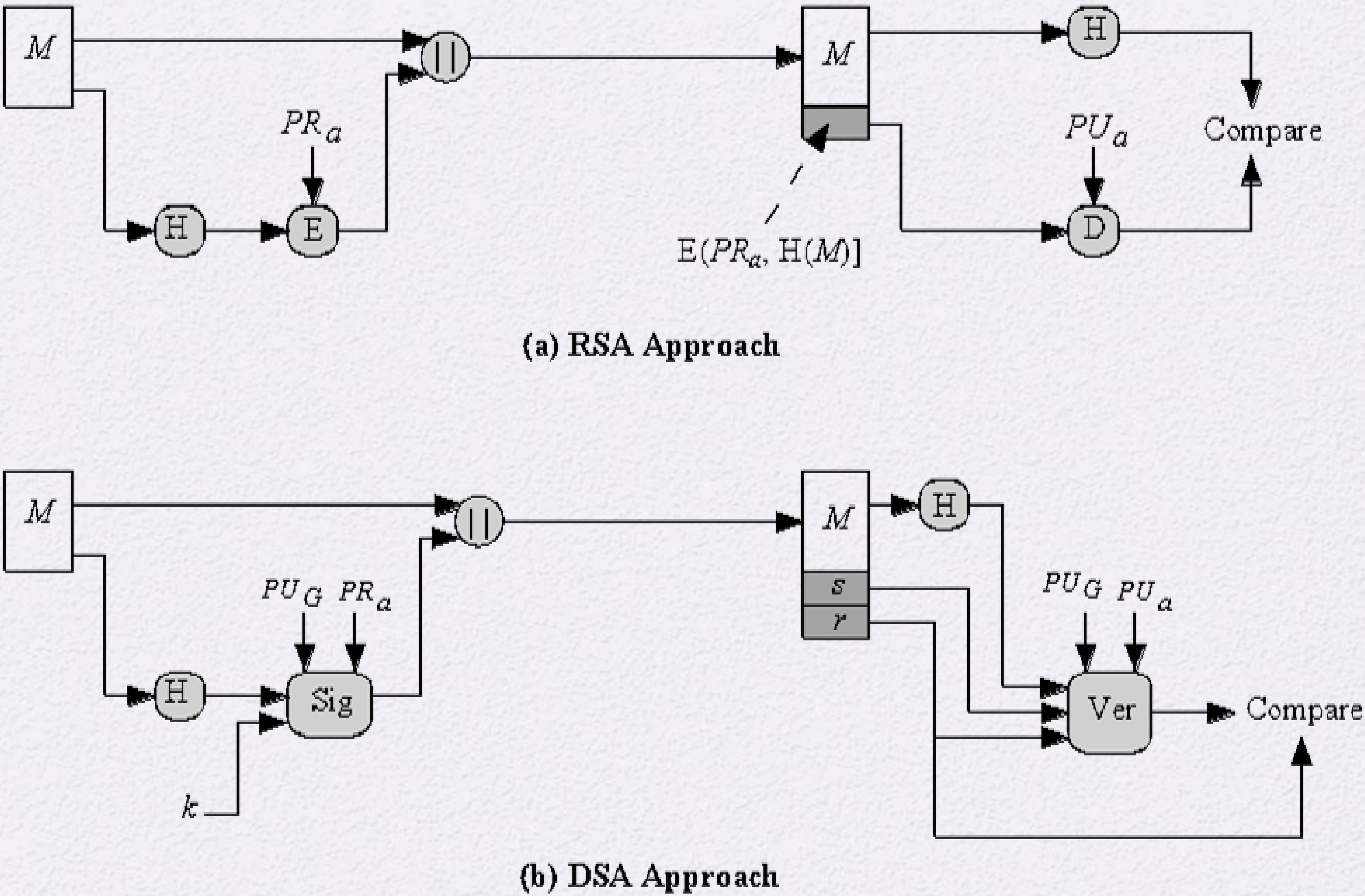


Figure 13.2 Two Approaches to Digital Signatures

Global Public-Key Components

p prime number where $2^{L-1} < p < 2^L$
for $512 \leq L \leq 1024$ and L a multiple of 64;
i.e., bit length L between 512 and 1024 bits
in increments of 64 bits

q prime divisor of $(p - 1)$, where $2^{N-1} < q < 2^N$
i.e., bit length of N bits

g $= h(\mathrm{p} - 1)/\mathrm{q}$ is an exponent mod p,
where h is any integer with $1 < h < (p - 1)$
such that $h^{(p-1)/q} \bmod p > 1$

User's Private Key

x random or pseudorandom integer with $0 < x < q$

User's Public Key

$y = g^x \bmod p$

User's Per-Message Secret Number

k random or pseudorandom integer with $0 < k < q$

Figure 13.3 The Digital Signature Algorithm (DSA)

Signing

$$r \;=\; (g^k \bmod p) \bmod q$$

$$s \;=\; [k^{-1}\,(H(M) + xr)] \bmod q$$

$$\text{Signature} = (r, s)$$

Verifying

$$w = (s')^{-1} \bmod q$$

$$u_1 = [H(M')w] \bmod q$$

$$u_2 = (r')w \bmod q$$

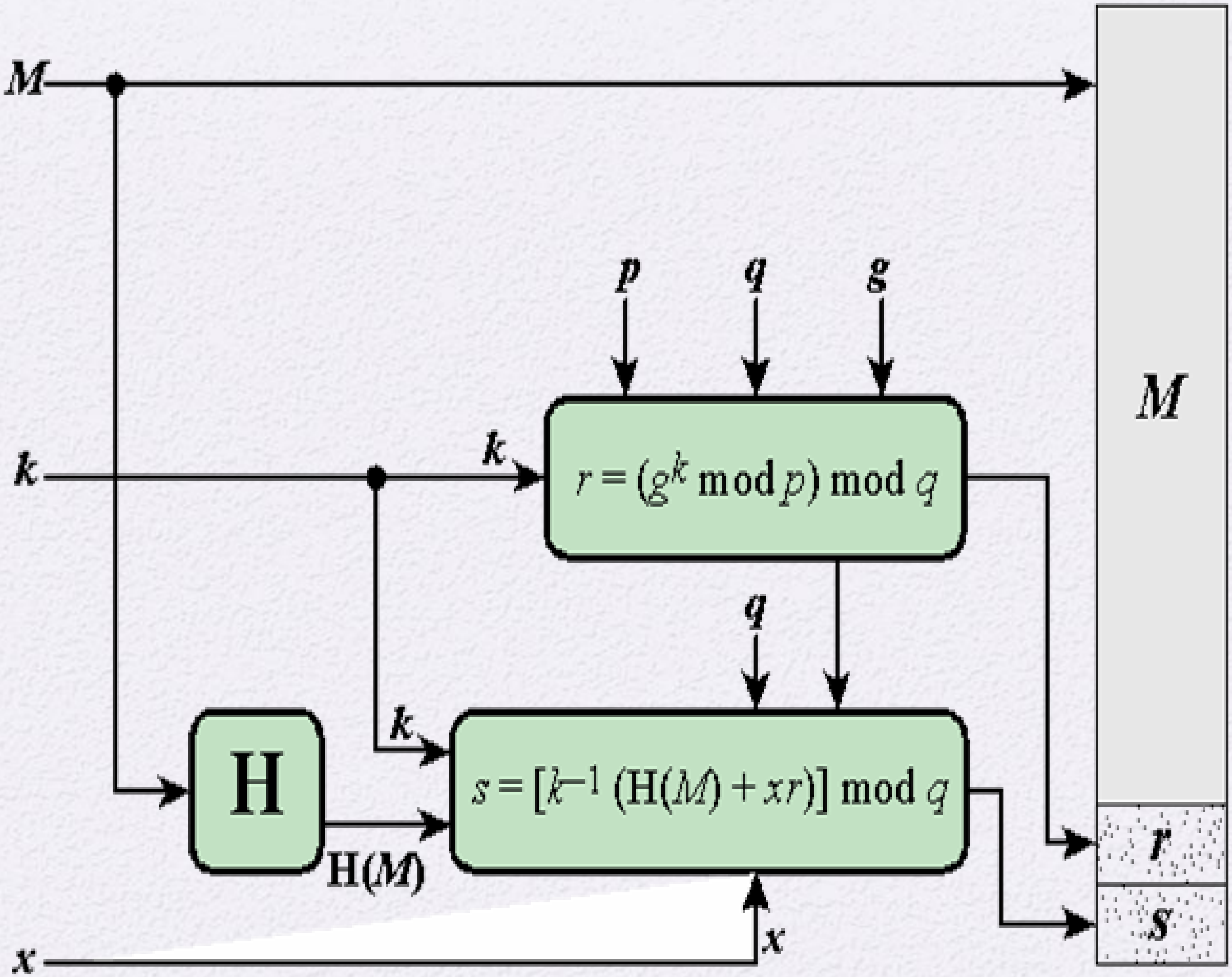
$$v = [(g^{u1} y^{u2}) \bmod p] \bmod q$$

$$\text{TEST: } v = r'$$

M = message to be signed

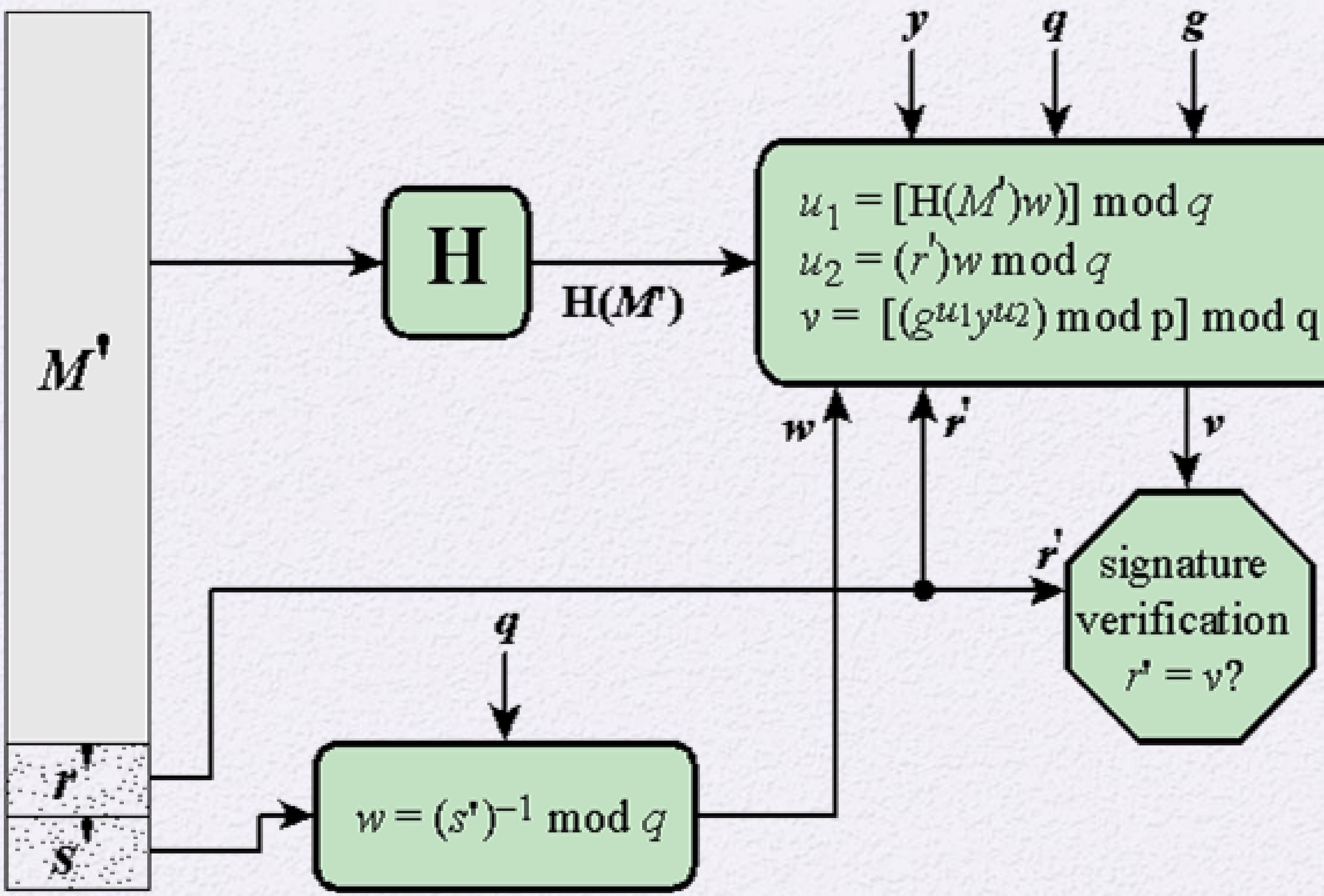
H(M) = hash of M using SHA-1

M', r', s' = received versions of M, r, s



(a) Signing

Figure 13.4 DSS Signing and Verifying



(b) Verifying

Figure 13.4 DSS Signing and Verifying

RSA Signature scheme: $\rightarrow$ Not secure

$Pr = \langle N, d \rangle$ Bob Alice

$PU = \langle N, e \rangle$

$\xrightarrow{(m, \sigma)}$

Sign msg m: $\leftarrow Pr, m$

$$\sigma := m^d \bmod N$$

verify: $\leftarrow PU, (m, \sigma)$

$$\sigma^e \stackrel{?}{=} m$$

Attack 1:

No-msg attack : Goal: give a valid (m, σ)

The attacker picks a random σ.

$$m = \sigma^e \bmod N$$

return: (m, σ)

Attack 2: Arbitrary msg attack:

1. Pick any msg m
2. Pick a random m_1
3. $m_2 = \frac{m}{m_1} \bmod N$ $\qquad m = m_1 \cdot m_2$
4. $(m_1, \sigma_1), (m_2, \sigma_2)$ He uses σ_1, σ_2 to calculate σ:
5. $\sigma^e = (\sigma_1 \cdot \sigma_2)^e = (m_1^d \cdot m_2^d)^e = m_1 \cdot m_2 = m$

Hash RSA signature $\rightarrow$ secure

$$\sigma := [H(m)]^d \bmod N$$

Summary

- Present an overview of the digital signature process

Understand the NIST digital signature scheme