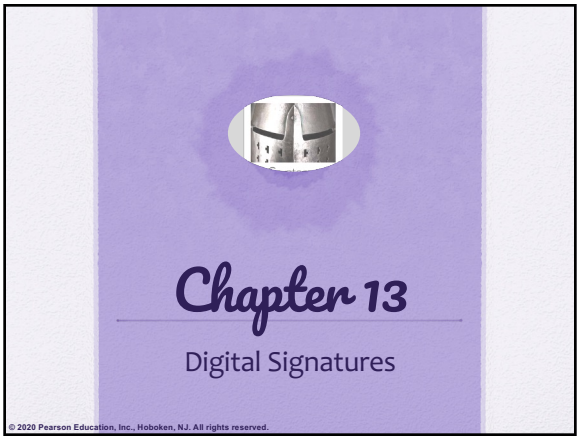


1

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.

2

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

3

Digital Signatures

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

4

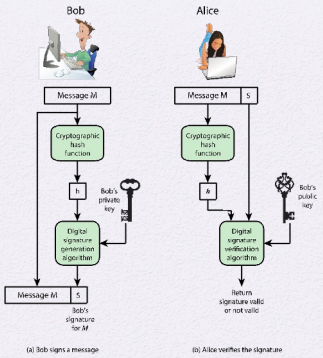


Figure 13.1 Simplified Depiction of Essential Elements of Digital Signature Process

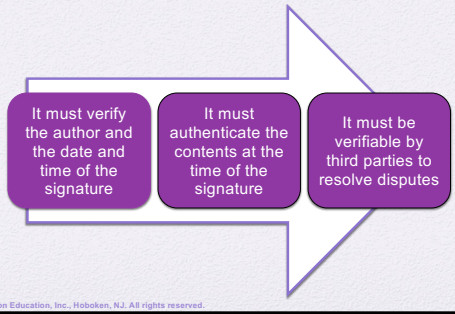
5

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

6

Digital Signature Properties



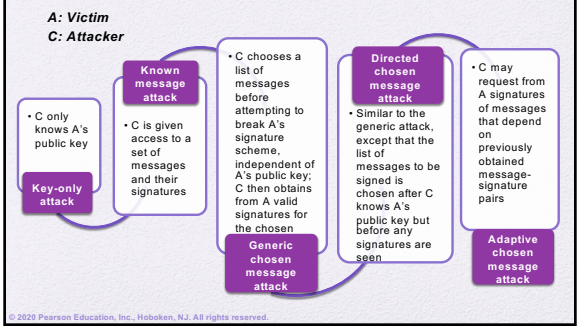
7

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

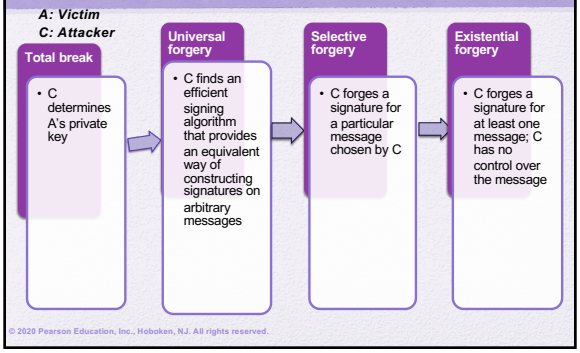
8

Attacks



9

Forgeries



10

Digital Signature Notions

- While there are many formal definitions for the security of digital signature. The two most common ones you will encounter are:
- EUF-CMA
  - Existential Unforgeability-Under Chosen Message Attack
- SeUF-CMA
  - Strong Existential Unforgeability-Under Chosen Message Attack

11

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

12

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

13

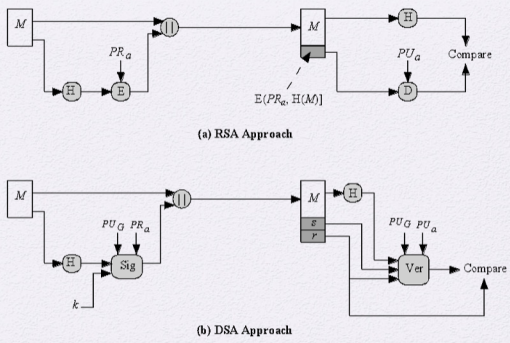


Figure 13.2 Two Approaches to Digital Signatures

14

Global Public-Key Components

p prime number where $2^{L-1} < p < 2^L$ for $512 \le L \le 1024$ 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 $= h^{(p-1)/q} \bmod p$, 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)

15

Signing

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

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

Signature $= (r, s)$

Verifying

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

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

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

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

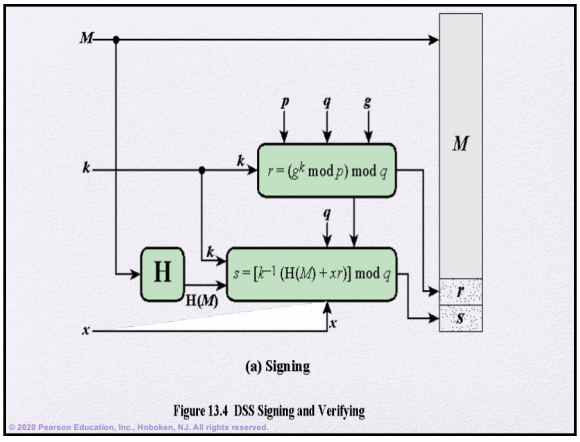
TEST: $v = r'$

M = message to be signed

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

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

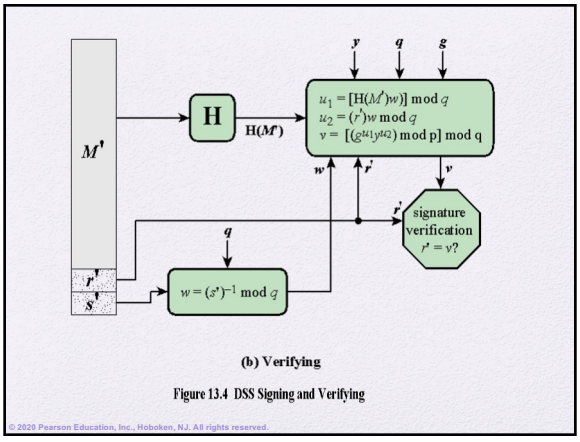
16



(a) Signing

Figure 13.4 DSS Signing and Verifying

17



(b) Verifying

Figure 13.4 DSS Signing and Verifying

18

Summary

- Present an overview of the digital signature process

Understand the NIST digital signature scheme

19