Chapter 13
Digital Signatures

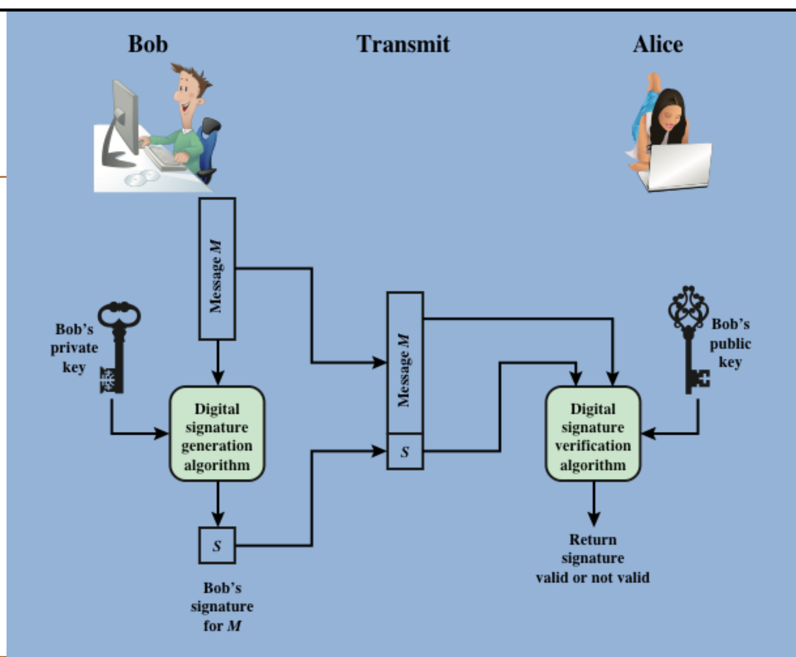


Figure 13.1 Generic Model of Digital Signature Process

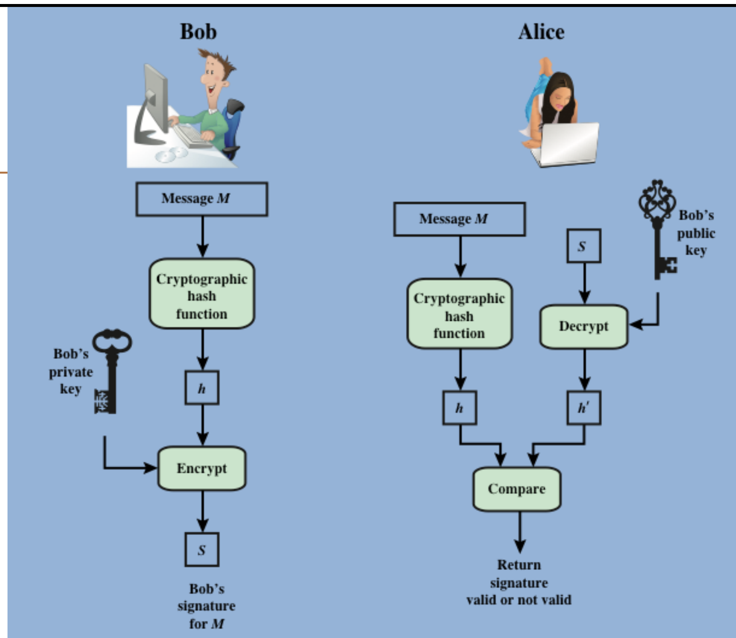


Figure 13.2 Simplified Depiction of Essential Elements of Digital Signature Process

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

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

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

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

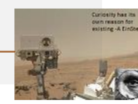
- C forges a signature for a particular message chosen by C

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



Direct Digital Signature

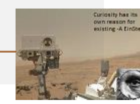
- Refers to a digital signature scheme that involves only the communicating parties
  - It is assumed that the destination knows the public key of the source

- Confidentiality can be provided by encrypting the entire message plus signature with a shared secret key
  - It is important to perform the signature function first and then an outer confidentiality function
  - In case of dispute some third party must view the message and its signature

- The validity of the scheme depends on the security of the sender's private key
  - If a sender later wishes to deny sending a particular message, the sender can claim that the private key was lost or stolen and that someone else forged his or her signature
  - One way to thwart or at least weaken this ploy is to require every signed message to include a timestamp and to require prompt reporting of compromised keys to a central authority

ElGamal Digital Signature

- Scheme involves the use of the private key for encryption and the public key for decryption
- Global elements are a prime number q and a, which is a primitive root of q
- Use private key for encryption (signing)
- Uses public key for decryption (verification)
- Each user generates their key
  - Chooses a secret key (number): $1 < x_A < q-1$
  - Compute their public key: $y_A = a^{x_A} \bmod q$



Schnorr Digital Signature

- Scheme is based on discrete logarithms
- Minimizes the message-dependent amount of computation required to generate a signature
  - Multiplying a $2n$-bit integer with an n-bit integer
- Main work can be done during the idle time of the processor
- Based on using a prime modulus p, with $p - 1$ having a prime factor q of appropriate size
  - Typically p is a 1024-bit number, and q is a 160-bit number

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

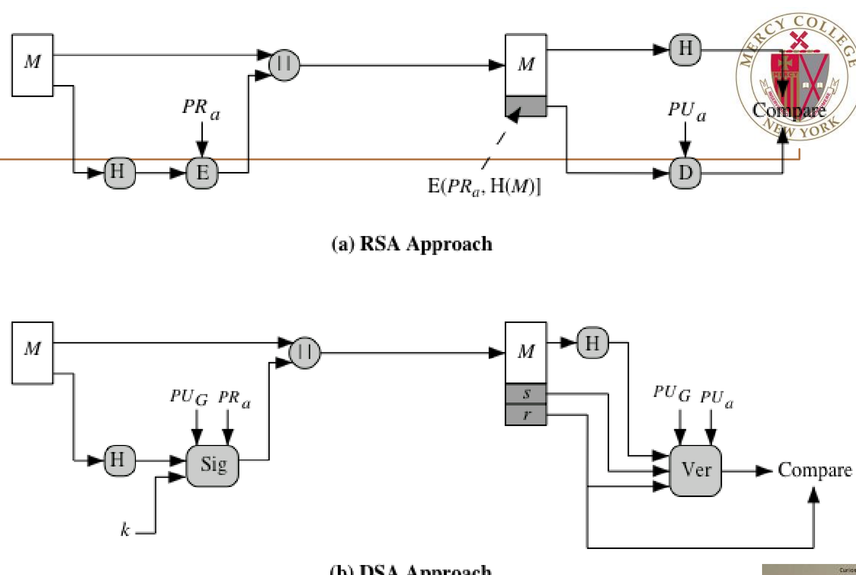


Figure 13.3 Two Approaches to Digital Signatures

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 of between 512 and 1024 bits in increments of 64 bits
q	prime divisor of $(p-1)$, where $2^{159} < q < 2^{160}$ i.e., bit length of 160 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 $0 < k < q$

Signing

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

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

Signature $= (r, s)$

Verifying

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

$u_1 = \left[\mathrm{H}(M')w\right] \bmod q$

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

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

TEST: $v = r'$

M	=	message to be signed
$\mathrm{H}(M)$	=	hash of M using SHA-1
M', r', s'	=	received versions of M, r, s

Figure 13.4 The Digital Signature Algorithm (DSS)

DSA Signing and Verifying

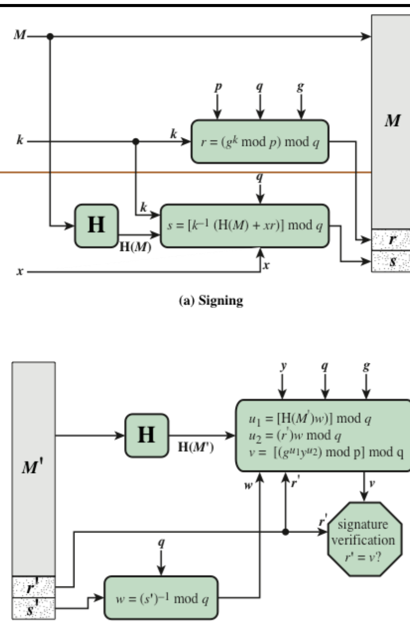


Figure 13.5 DSA Signing and Verifying

Elliptic Curve Digital Signature Algorithm (ECDSA)

Four elements are involved:

- All those participating in the digital signature scheme use the same global domain parameters, which define an elliptic curve and a point of origin on the curve
- A signer must first generate a public, private key pair
- A hash value is generated for the message to be signed; using the private key, the domain parameters, and the hash value, a signature is generated
- To verify the signature, the verifier uses as input the signer's public key, the domain parameters, and the integer *s;* the output is a value *v* that is compared to *r* ; the signature is verified if the *v* = *r*

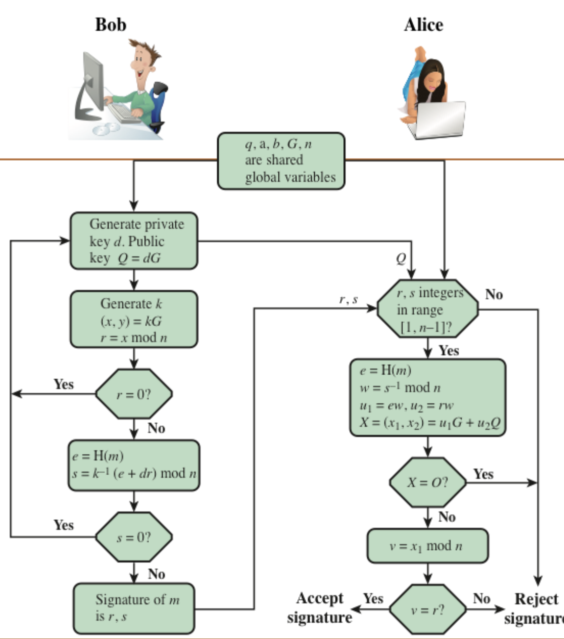
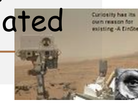


Figure 13.6 ECDSA Signing and Verifying

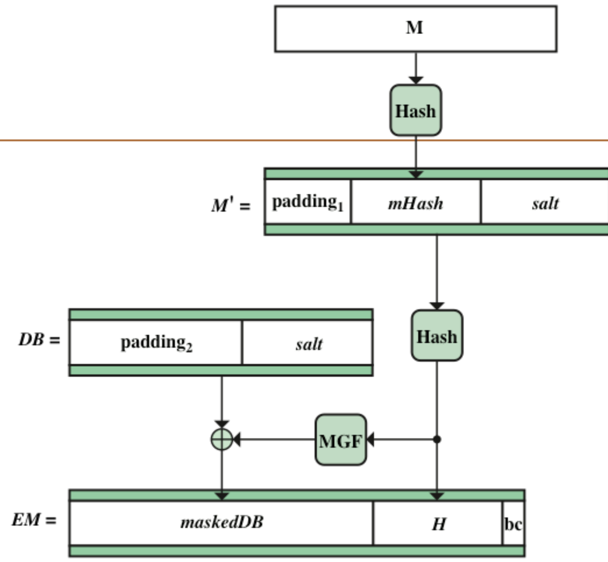
RSA-PSS

- RSA Probabilistic Signature Scheme
- Included in the 2009 version of FIPS 186
- Latest of the RSA schemes and the one that RSA Laboratories recommends as the most secure of the RSA schemes
- For all schemes developed prior to PSS is has not been possible to develop a mathematical proof that the signature scheme is as secure as the underlying RSA encryption/decryption primitive
- The PSS approach was first proposed by Bellare and Rogaway
- This approach, unlike the other RSA-based schemes, introduces a randomization process that enables the security of the method to be shown to be closely related to the security of the RSA algorithm itself



Mask Generation Function (MGF)

- Typically based on a secure cryptographic hash function such as SHA-1
  - Is intended to be a cryptographically secure way of generating a message digest, or hash, of variable length based on an underlying cryptographic hash function that produces a fixed-length output



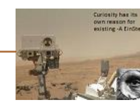


Figure 13.7 RSA-PSS Encoding

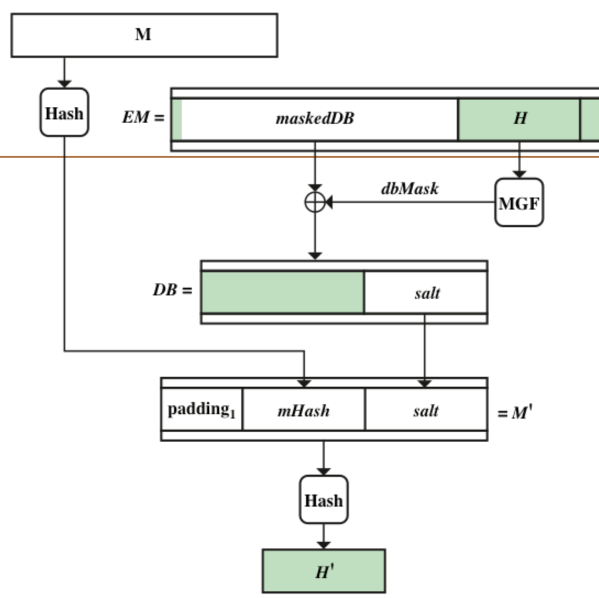


Figure 13.8 RSA-PSS EM Verification

Summary

- Digital signatures
  - Properties
  - Attacks and forgeries
  - Digital signature requirements
  - Direct digital signature
- Elgamal digital signature scheme
- RSA-PSS
  - Mask generation function
  - The signing operation
  - Signature verification
- NIST digital signature algorithm
  - The DSA approach
- Elliptic curve digital signature algorithm
  - Global domain parameters
  - Key generation
  - Digital signature generation and authentication
  - Schnorr digital signature scheme