THE DSS APPROACH

- The DSS uses an algorithm that is designed to provide only the digital signature function.
- Unlike RSA, it cannot be used for encryption or key exchange.
- Nevertheless, it is a public-key technique.

TWO APPROACHES TO DIGITAL SIGNATURES

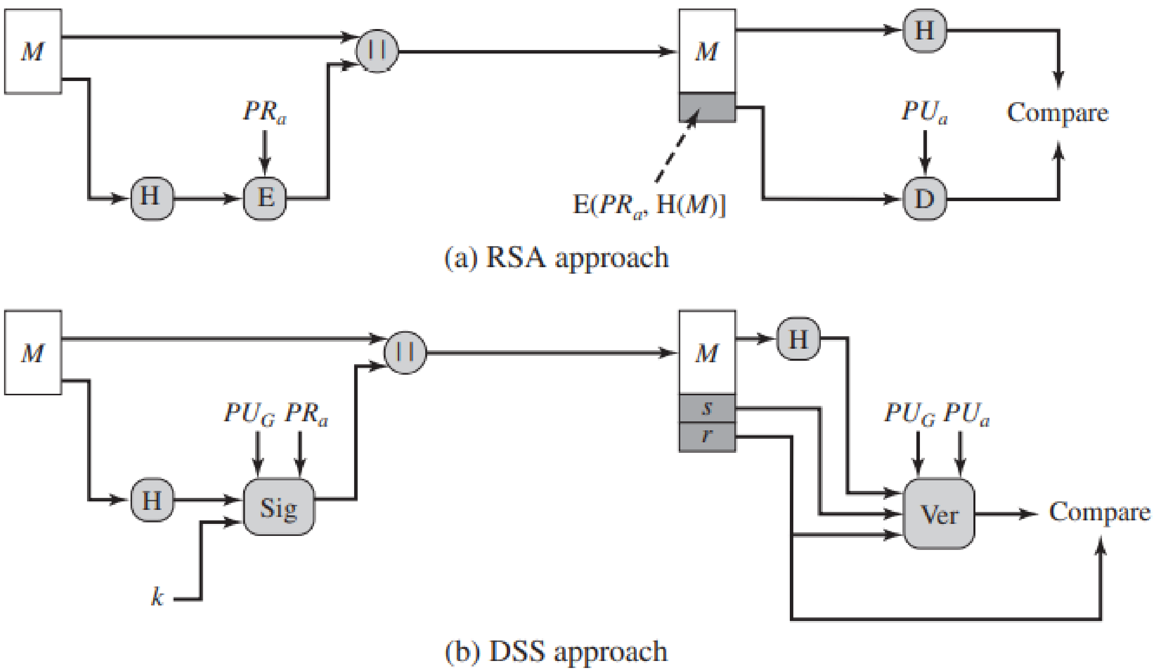


Figure 13.3 Two Approaches to Digital Signatures

Reference :William Stallings, Cryptography and Network Security: Principles and Practice, PHI 3rd Edition, 2006

- In the RSA approach, the message to be signed is input to a hash function that produces a secure hash code of fixed length.
- This hash code is then encrypted using the sender's private key to form the signature.
- Both the message and the signature are then transmitted.
- The recipient takes the message and produces a hash code.
- The recipient also decrypts the signature using the sender's public key.
- If the calculated hash code matches the decrypted signature, the signature is accepted as valid. Because only the sender knows the private key, only the sender could have produced a valid signature.
- The DSS approach also makes use of a hash function.

- The hash code is provided as input to a signature function along with a random number k generated for this particular signature.
- The signature function also depends on the sender's private key (PR_a) and a set of parameters known to a group of communicating principals.
- We can consider this set to constitute a global public key (PU_G).
- The result is a signature consisting of two components, labeled s and r.
- At the receiving end, the hash code of the incoming message is generated.
- This plus the signature is input to a verification function.
- The verification function also depends on the global public key as well as the sender's public key PUa, which is paired with the sender's private key.
- The output of the verification function is a value that is equal to the signature component if the signature is valid.
- The signature function is such that only the sender, with knowledge of the private key, could have produced the valid signature

THE DIGITAL SIGNATURE ALGORITHM

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

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

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

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

Figure 13.4 The Digital Signature Algorithm (DSA)

Reference :William Stallings, Cryptography and Network Security: Principles and Practice, PHI 3rd Edition, 2006

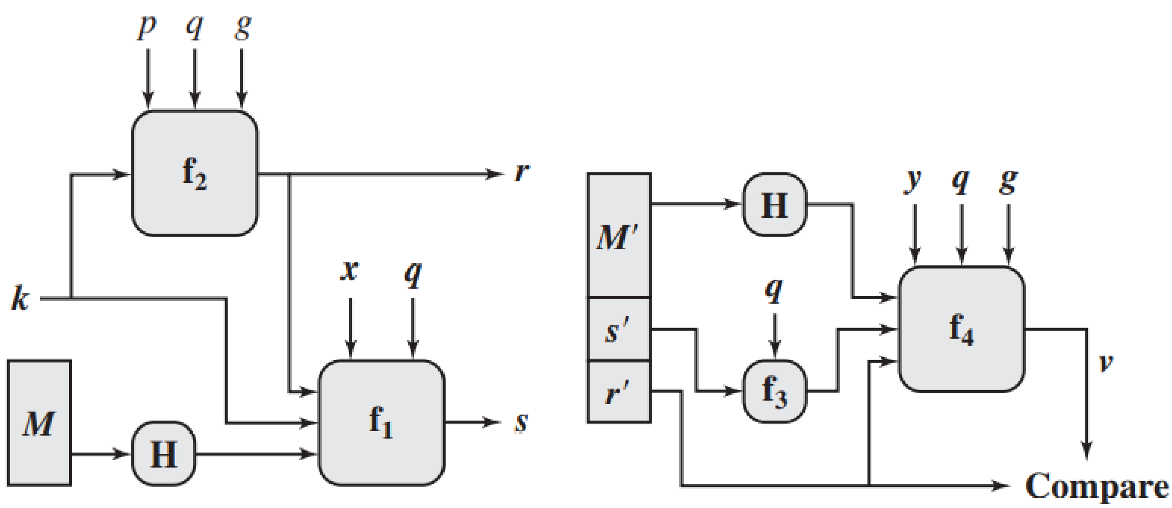
THE DIGITAL SIGNATURE ALGORITHM

- There are three parameters that are public and can be common to a group of users.
- A 160-bit prime number q is chosen.
- Next, a prime number p is selected with a length between 512 and 1024 bits such that q divides (p-1).
- Finally, g is chosen to be of the form $h^{(p-1)/q}$ mod p where h is an integer between 1 and (p-1). with the restriction that must be greater than 1^2.
- Thus, the global public-key components of DSA have the same for as in the Schnorr signature scheme.
- With these numbers in hand, each user selects a private key and generates a public key.The private key x must be a number from 1 to (q-1) and should be chosen randomly or pseudorandomly. The public key is calculated from the private key as $y=g^x$ mod p.
- The calculation of y given x is relatively straightforward. However, given the public key y, it is believed to be computationally infeasible to determine x, which is the discrete logarithm of y to the base g, mod p.

THE DIGITAL SIGNATURE ALGORITHM

- To create a signature, a user calculates two quantities, r and s, that are functions of the public key components (p,q,g), the user's private key (x), the hash code of the message H(M), and an additional integer k that should be generated randomly or pseudorandomly and be unique for each signing.
- At the receiving end, verification is performed using the formulas shown in Figure.
- The receiver generates a quantity v that is a function of the public key components, the sender's public key, and the hash code of the incoming message.
- If this quantity matches the r component of the signature, then the signature is validated.

DSS SIGNING AND VERIFYING



$s = f_1(H(M), k, x, r, q) = (k^{-1}(H(M) + xr)) \bmod q$

$r = f_2(k, p, q, g) = (g^k \bmod p) \bmod q$

(a) Signing

$w = f_3(s', q) = (s')^{-1} \bmod q$

$v = f_4(y, q, g, H(M'), w, r')$

$= ((g^{(H(M')w) \bmod q} y^{r'w \bmod q}) \bmod p) \bmod q$

(b) Verifying

Figure 13.5 DSS Signing and Verifying

Reference :William Stallings, Cryptography and Network Security: Principles and Practice, PHI 3rd Edition, 2006