Cryptography and Security

Cunsheng DING

HKUST, Hong Kong

Version 3

Lecture 14: Digital Signature Standard

Main Topics of This Lecture

1. The need for digital signatures.
2. Basic requirements for digital signatures.
3. The digital signature scheme with public-key ciphers.
4. The Digital Signature Standard (DSA), also called the Digital Signature Algorithm (DSA).

The Need for Digital Signature

Scenario: Assume that Alice and Bob share a secret key k_1 for the keyed hash function and another one k_2 for a one-key cipher. Consider the following authentication protocol.

$$\text{Alice} \quad \longrightarrow \quad E_{k_2}[m||h_{k_1}(m)] \quad \longrightarrow \quad \text{Bob}$$

Problems: Assume that Alice sends an authenticated message to Bob.

- Bob may forge a message and claim that it came from Alice.
- Alice can deny sending the message.

Solution: Digital signature, analogous to handwritten signature.

Basic Requirements (1)

- The signature must depend 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.

Basic Requirements (2)

- 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 with Public-Key Cryptosystems

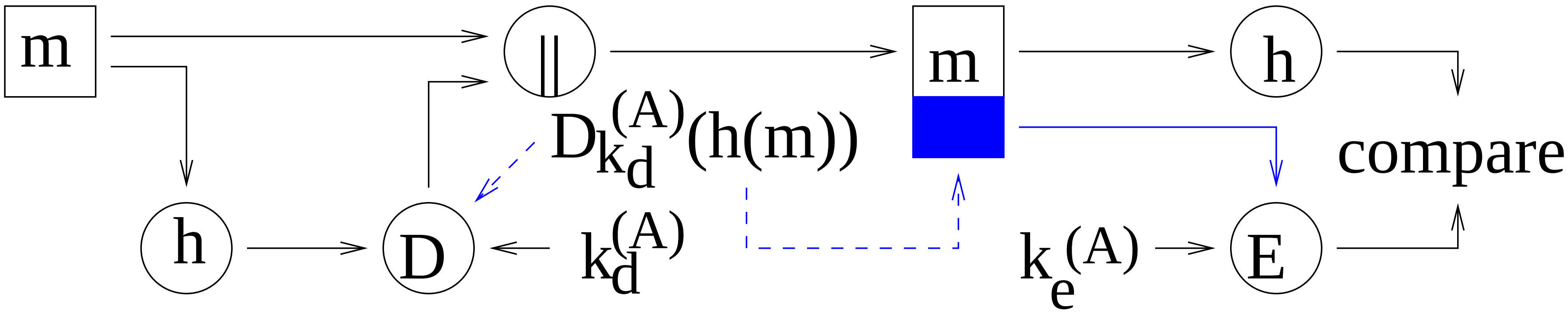
Definition: It involves only the communicating parties (source, destination).

Protocol: Let h be a hash function. Assume that Alice and Bob share a secret key k of a one-key cipher, and have exchanged their public keys.

$$\text{Alice} \quad \longrightarrow \quad E_k\left(m || D_{k_d^{(A)}}[h(m)]\right) \quad \longrightarrow \quad \text{Bob}$$

Question: Which of the basic requirements for digital signature are met!

Two Approaches to Digital Signatures: RSA



(a) RSA approach

Recall: The protocol needs a hash function and public-key cryptosystem. Alice sends $m||D_{k_d}^{(A)}(h(m))$ to Bob. Then Bob verifies the sender and message.

Two Approaches to Digital Signatures: DSS

DSS building blocks: a hash function h, a set of parameters known to a group of communicating participants – global public-key $k_e^{(G)}$, a signature function sig, and a verification function ver.

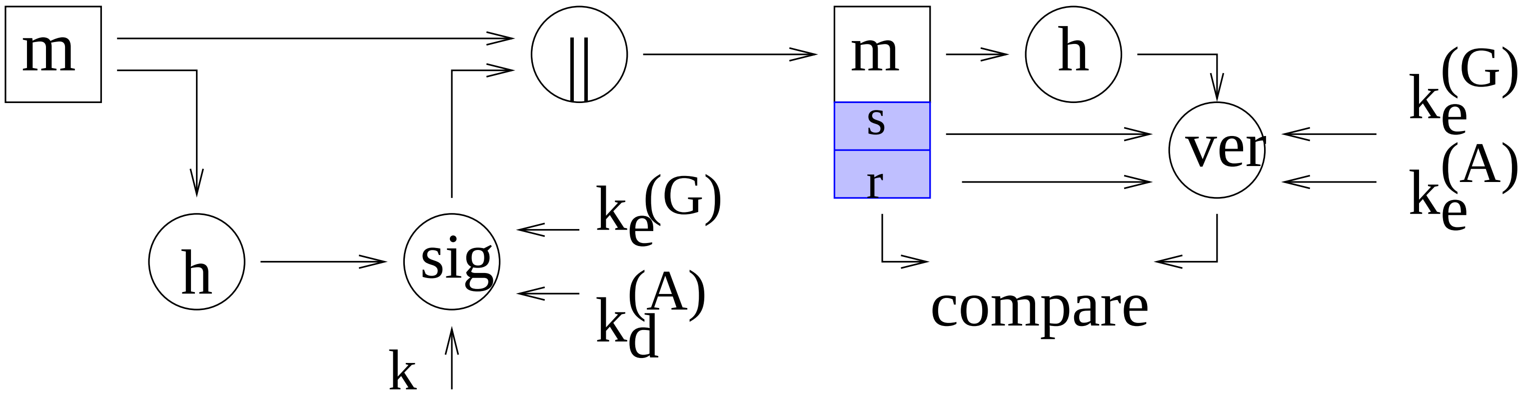
Each user A has a private key $k_d^{(A)}$ for signing, and a public key $k_e^{(A)}$ for verifying. Therefore

$$sig = sig\left[k, k_e^{(G)}, k_d^{(A)}, h(m)\right],$$

$$ver = ver\left[k_e^{(G)}, k_e^{(A)}, h(m), sig(m)\right],$$

Where k is a random secret number for this m.

Two Approaches to Digital Signatures: DSS



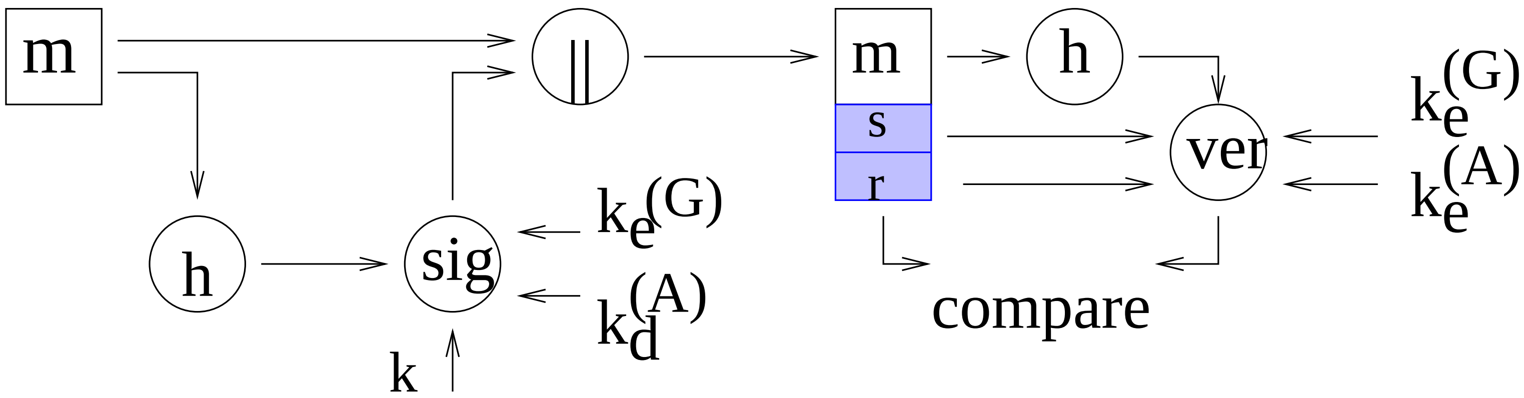
(b) DSS Approach

Signing: To sign, A generates a random number k for this session and computes the signature:

$$(s, r) = sig\left[h(m), k, k_e^{(G)}, k_d^{(A)}\right].$$

A will then send $m||s||r$ to the receiver.

Two Approaches to Digital Signatures: DSS



(b) DSS Approach

Verifying: After "partitioning" a received text into $m'||s'||r'$, where r' has the same length as r and s' has the same length as s, the receiver uses the public parameters $k_e^{(G)}$, $k_e^{(A)}$, and h to compute

$$v = ver\left[h(m'), r', s', k_e^{(G)}, k_e^{(A)}\right].$$

The receiver then verifies the signature by checking $v = r'$.

DSS: Description of Building Blocks

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.

q**:** prime divisor of $(p-1)$, where $2^{159} < q < 2^{160}$, i.e., bit length of 160.

g**:** $= h^{(p-1)/q} \bmod p$, where h is any integer with $1 < h < (p-1)$ such that $h^{(p-1)/q} \bmod p > 1$.

DSS: Description of Building Blocks

User's parameters

User's private key: x, a random or pseudo-random integer with $0 < x < q$.

User's public key: $y = g^x \bmod p$.

User's per-message secret number: k, a random or pseudo-random integer with $0 < k < q$.

DSS: Signing

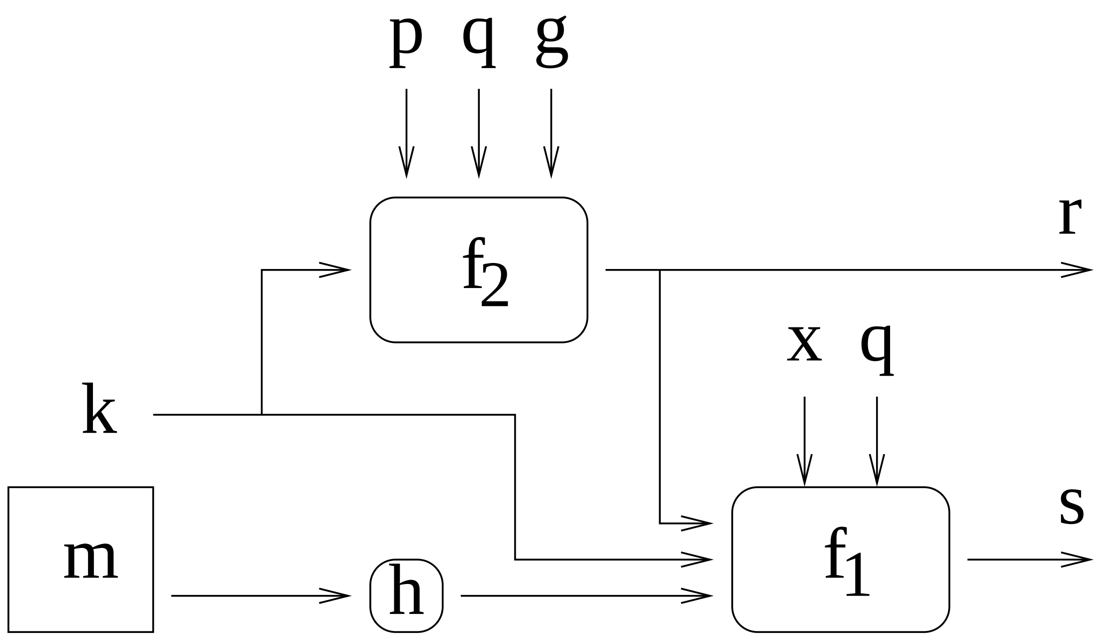
A uses her private key x, the public key components (p, q, g), and a random integer k to compute

- $r = (g^k \bmod p) \bmod q$.
- $s = [k^{-1}(h(m) + xr)] \bmod q$,

where h is the hash algorithm SHA-1 with a 160-bit hash value. If $s = 0$, then A has to choose another random k and recompute the signature so that $s \neq 0$. Note that $\Pr(s = 0) = 2^{-160}$.

The signature of m is (s, r).

DSS: Pictorial Description of Signing



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

$$s = f_1(h(m), k, x, r, q) = [k^{-1}(h(m) + xr)] \bmod q.$$

DSS: Verifying

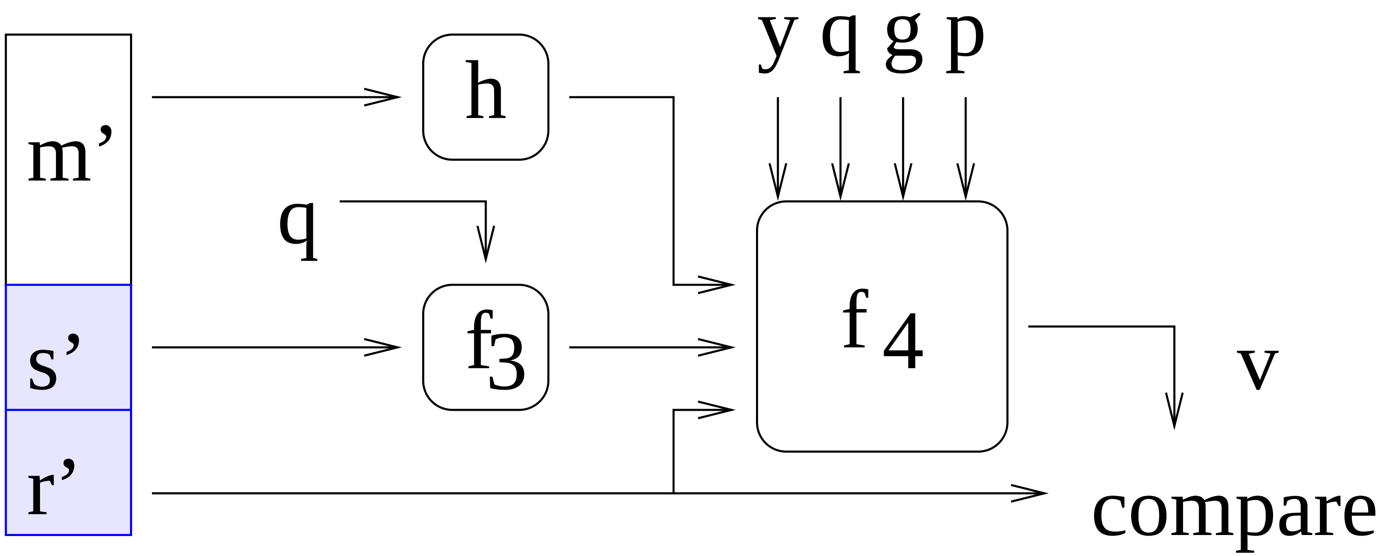
Let $m'||s'||r'$ be the received data. The receiver uses A's public key y and the public parameters (p, q, g) to compute

- $w = (s')^{-1} \bmod q$.
- $u_1 = [h(m')w] \bmod q$.
- $u_2 = (r')w \bmod q$.
- $v = [(g^{u_1} y^{u_2}) \bmod p] \bmod q$.

Finally, test whether $v = r'$.

Remark: The proof of correctness appears later.

DSS: Pictorial Description of Verification



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

$$v = f_4(y, p, q, g, h(m'), w, r') = [g^{(h(m')w) \bmod q} y^{r'w \bmod q} \bmod p] \bmod q$$

Correctness of the Verification Method

Lemma 1: $k = (h(m) + xr)s^{-1} \bmod q$.

Proof: Note that $s \neq 0 \bmod q$. By definition

$$s = [k^{-1}(h(m) + xr)] \bmod q.$$

Multiplying both sides with $s^{-1}k$ gives

$$k = [s^{-1}(h(m) + xr)] \bmod q.$$

Correctness of the Verification Method

Lemma 2: $g^{x(rs^{-1} \bmod q)} \bmod p = g^{xrs^{-1} \bmod q} \bmod p$

Proof: By Euler's theorem, $g^q \bmod p = h^{p-1} \bmod p = 1$. Let

$$rs^{-1} = qS + R$$

for some S and $0 \le R < q$ and

$$xR = qT + Z$$

for some T and $0 \le Z < q$. Then

$$g^{x(rs^{-1} \bmod q)} \bmod p = g^{xR} \bmod p = (g^q)^T g^Z \bmod p = g^Z \bmod p$$

and

$$g^{xrs^{-1} \bmod q} \bmod p = g^{xR \bmod q} \bmod p = g^Z \bmod p.$$

The desired conclusion then follows.

Correctness of the Verification Method

Theorem: Let

- $u_1 = [h(m)s^{-1}] \bmod q$.
- $u_2 = rs^{-1} \bmod q$.
- $v = [(g^{u_1} y^{u_2}) \bmod p] \bmod q$.

Then $v = r$.

Remark: The verification uses this result and checks whether $v = r$.

Correctness of the Verification Method

Proof of Theorem: By Lemma 1 and by definition

$$
\begin{aligned}
v &= [(g^{u_1} y^{u_2}) \bmod p] \bmod q \\
&= [(g^{[h(m)s^{-1}] \bmod q} \times y^{rs^{-1} \bmod q}) \bmod p] \bmod q \\
&= [(g^{[h(m)s^{-1}] \bmod q} \times g^{x(rs^{-1} \bmod q)}) \bmod p] \bmod q \\
&= [(g^{[h(m)s^{-1}] \bmod q} \times g^{xrs^{-1} \bmod q}) \bmod p] \bmod q \text{ (by Lemma 2)} \\
&= [g^{[h(m)s^{-1}+xrs^{-1}] \bmod q} \bmod p] \bmod q \\
&= [g^{[h(m)+xr]s^{-1} \bmod q} \bmod p] \bmod q \\
&= [g^k \bmod p] \bmod q \text{ (by Lemma 1)} \\
&= r.
\end{aligned}
$$

Security of the Digital Signature Standard

Question: Is it possible to derive the private key from the public parameters?

Answer: The public parameters are

$$(p, g, q), y.$$

The private key x is only related to those parameters by

$$y = g^x \bmod p.$$

So one has to solve this discrete-logarithm-like problem, which is believed to be hard in general. Note that p and q are very large. Notice that g may be or may not be a primitive root modulo p.

Security of the Digital Signature Standard

Question: Is it possible to derive the private key x from some $m||s||r$?

Answer: Recall that

- $r = (g^k \bmod p) \bmod q$.
- $s = [k^{-1}(h(m) + xr)] \bmod q$.

Note that the random integer k is used only for one message. Having more than one $m||s||r$ does not help.

Observation: Solving the first equation directly is to solve a discrete-logarithm-like problem, which is believed to be hard.

Security of the Digital Signature Standard

Question: Is it possible to derive the private key x from some $m||s||r$?

Answer: Solving the set of equations yields

- $r = (g^{s^{-1}h(m)}(g^{s^{-1}r})^x \bmod p) \bmod q.$

Hence, this is to solve a discrete-logarithm-like problem, which is believed to be hard.

Historical Development of the DSS

- It was adopted as a standard on December 1 of 1994 by NIST.
- It makes use of SHA1.
- It is quite different from the digital signature system based on the RSA public-key cipher.
- In new versions of the DSS, new versions of SHA are used. In such case, the sizes of p and q are increased accordingly.