Course: Cryptography and Network Security
Code: CS-34310
Branch: M.C.A - 4th Semester

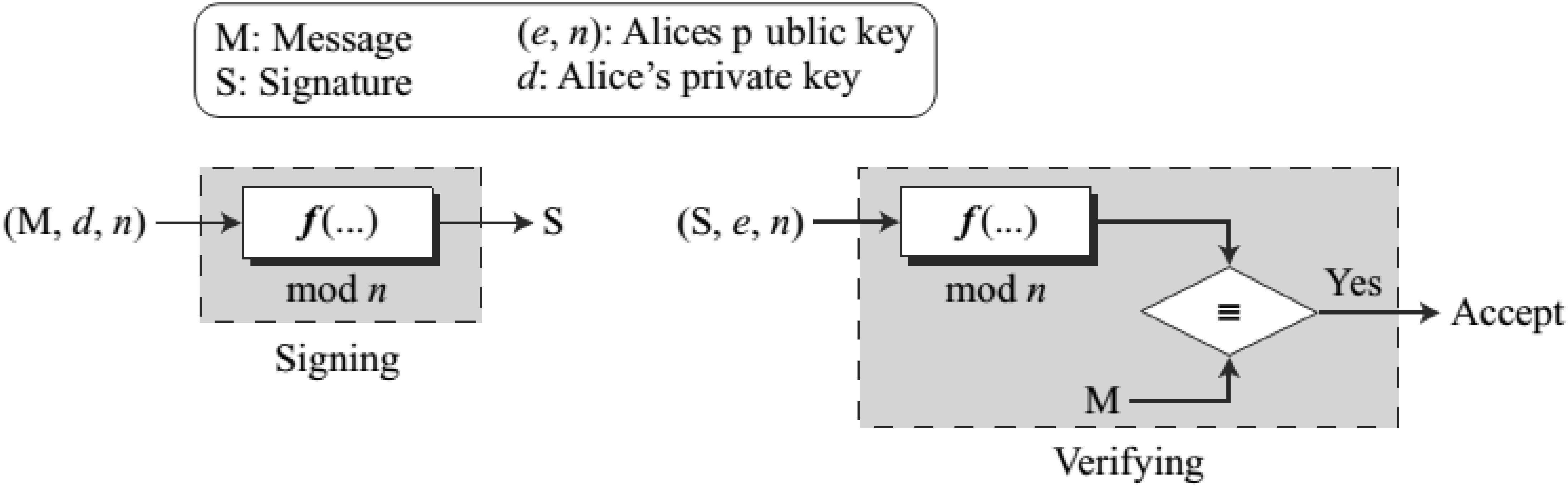
Lecture – 14 : DIGITAL SIGNATURE SCHEMES

Faculty & Coordinator : Dr. J Sathish Kumar (JSK)

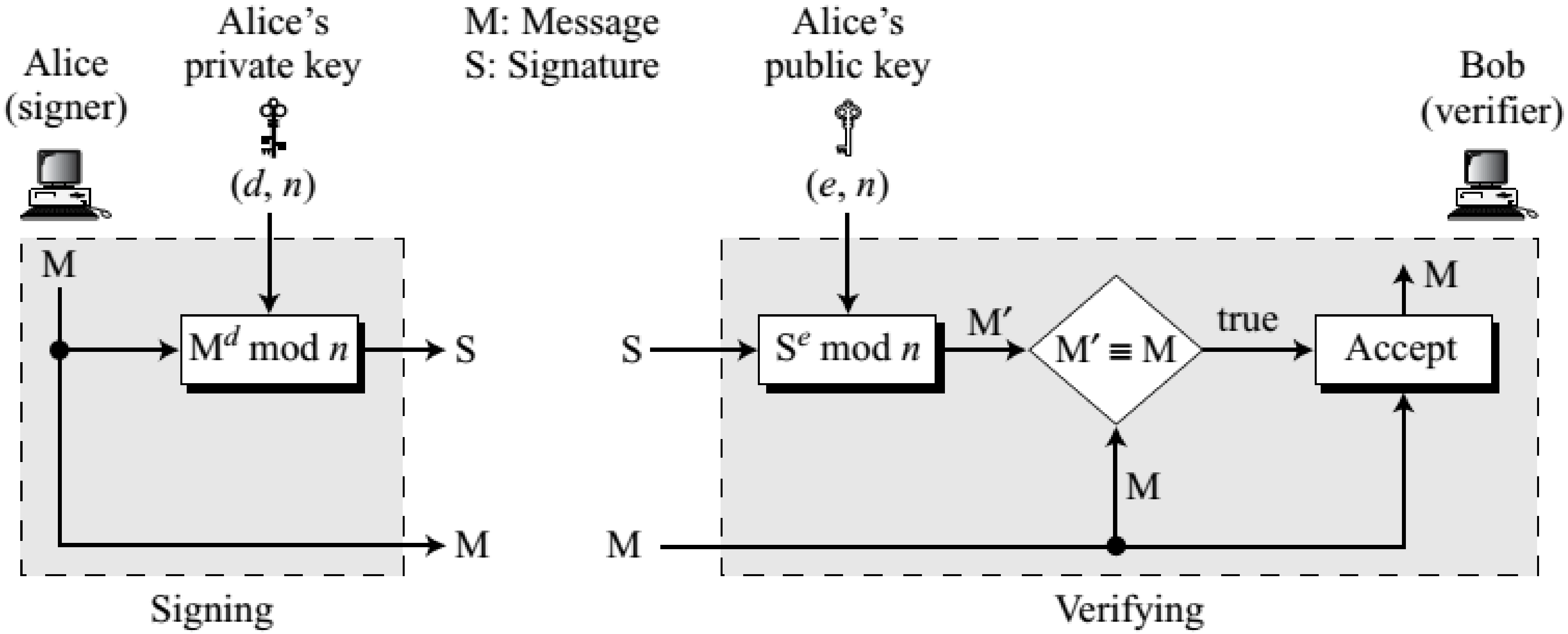
Department of Computer Science and Engineering

Motilal Nehru National Institute of Technology Allahabad,
Prayagraj-211004

RSA Digital Signature Scheme



RSA Digital Signature Scheme



$$M' \equiv M \pmod{n} \quad \rightarrow \quad S^e \equiv M \pmod{n} \quad \rightarrow \quad M^{d \times e} \equiv M \pmod{n}$$

RSA Digital Signature Scheme

- For the security of the signature, the value of p and q must be very large.
- As a trivial example, suppose that Alice chooses p = 823 and q = 953, and calculates n = 784319.
- The value of ϕ(n) is 782544. Now she chooses e = 313 and calculates d = 160009.
- At this point key generation is complete. Now imagine that Alice wants to send a message with the value of M = 19070 to Bob.
- She uses her private exponent, 160009, to sign the message

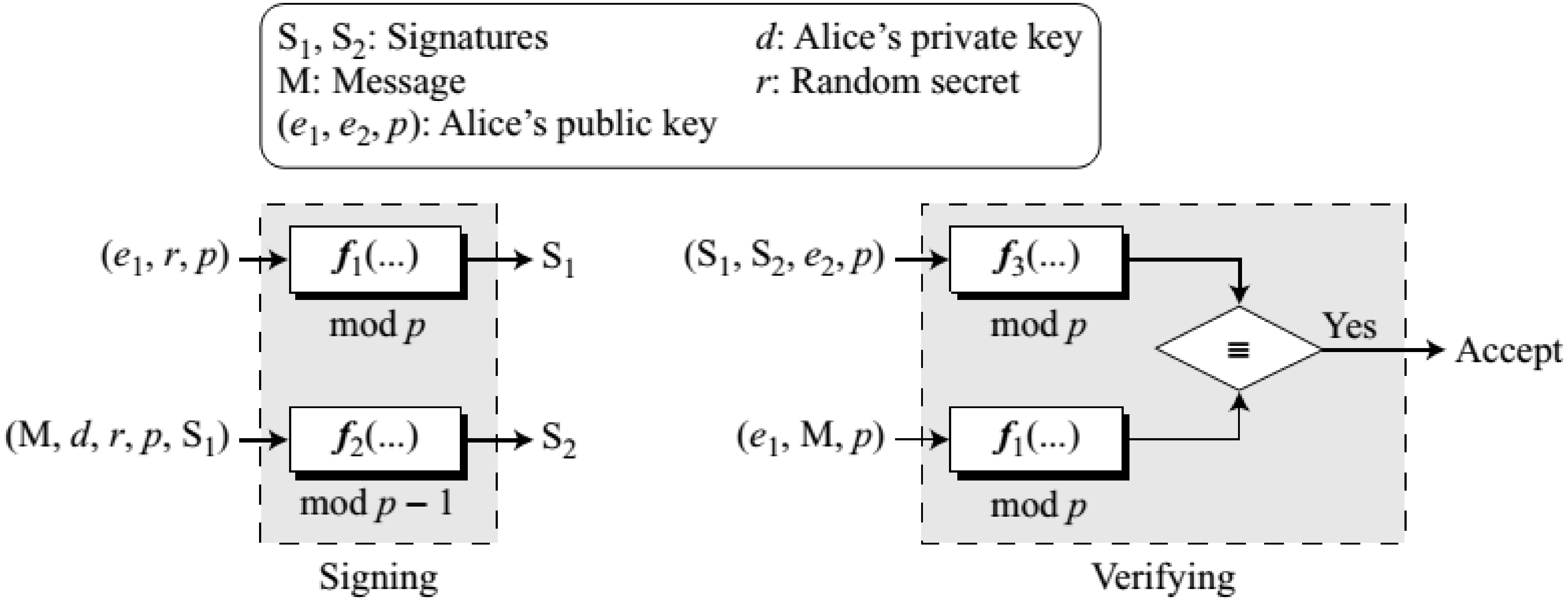
$$\text{M: } 19070 \quad \rightarrow \quad S = (19070^{160009}) \bmod 784319 = 210625 \bmod 784319$$

- Alice sends the message and the signature to Bob. Bob receives the message and the signature. He calculates

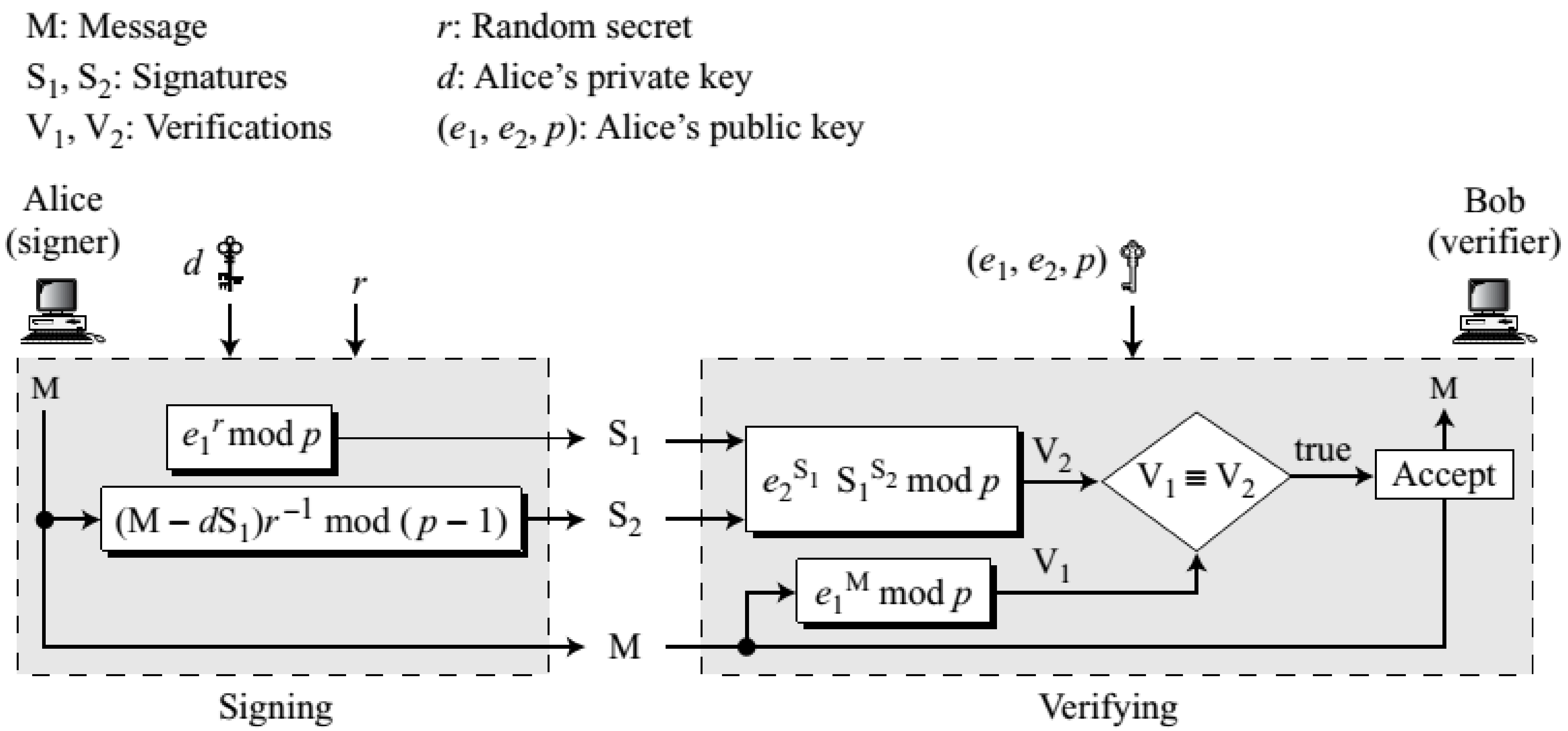
$$M' = 210625^{313} \bmod 784319 = 19070 \bmod 784319 \qquad \rightarrow \qquad M \equiv M' \bmod n$$

- Bob accepts the message because he has verified Alice's signature.

ElGamal Digital Signature Scheme



ElGamal Digital Signature Scheme



ElGamal Digital Signature Scheme

$V_1 \equiv V_2 \pmod{p} \rightarrow e_1^{M} \equiv e_2^{S_1} \times S_1^{S_2} \pmod{p} \equiv (e_1^{d})^{S_1} (e_1^{r})^{S_2} \pmod{p} \equiv e_1^{d S_1 + r S_2} \pmod{p}$

We get: $e_1^{M} \equiv e_1^{d S_1 + r S_2} \pmod{p}$

ElGamal Digital Signature Scheme

- Here is a trivial example. Alice chooses p = 3119, e1 = 2, d = 127 and calculates e2 = 2^{127} mod 3119 = 1702. She also chooses r to be 307. She announces e1, e2, and p publicly; she keeps d secret. The following shows how Alice can sign a message.

M = 320

$$S_1 = e_1^{\,r} = 2^{307} = 2083 \bmod 3119$$
$$S_2 = (M - d \times S_1) \times r^{-1} = (320 - 127 \times 2083) \times 307^{-1} = 2105 \bmod 3118$$

Alice sends M, S_1, and S_2 to Bob. Bob uses the public key to calculate V_1 and V_2.

$$V_1 = e_1^{\,M} = 2^{320} = 3006 \bmod 3119$$
$$V_2 = d^{\,S_1} \times S_1^{\,S_2} = 1702^{2083} \times 2083^{2105} = 3006 \bmod 3119$$

Because V_1 and V_2 are congruent, Bob accepts the message and he assumes that the message has been signed by Alice because no one else has Alice's private key, d.

ElGamal Digital Signature Scheme

Now imagine that Alice wants to send another message, M = 3000, to Ted. She chooses a new r, 107. Alice sends M, S_1, and S_2 to Ted. Ted uses the public keys to calculate V_1 and V_2.

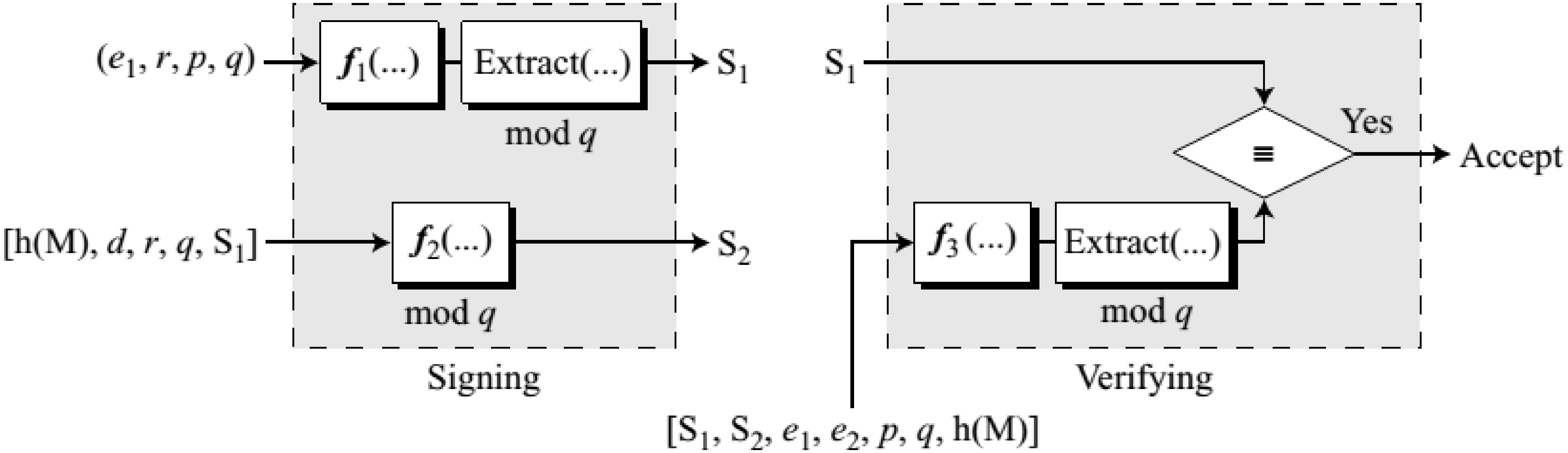
$$M = 3000$$
$$S_1 = e_1^{r} = 2^{107} = 2732 \bmod 3119$$
$$S_2 = (M - d \times S_1)\, r^{-1} = (3000 - 127 \times 2083) \times 107^{-1} = 2526 \bmod 3118$$

$$V_1 = e_1^{M} = 2^{3000} = 704 \bmod 3119$$
$$V_2 = d^{S_1} \times S_1^{S} = 1702^{2732} \times 2083^{2526} = 704 \bmod 3119$$

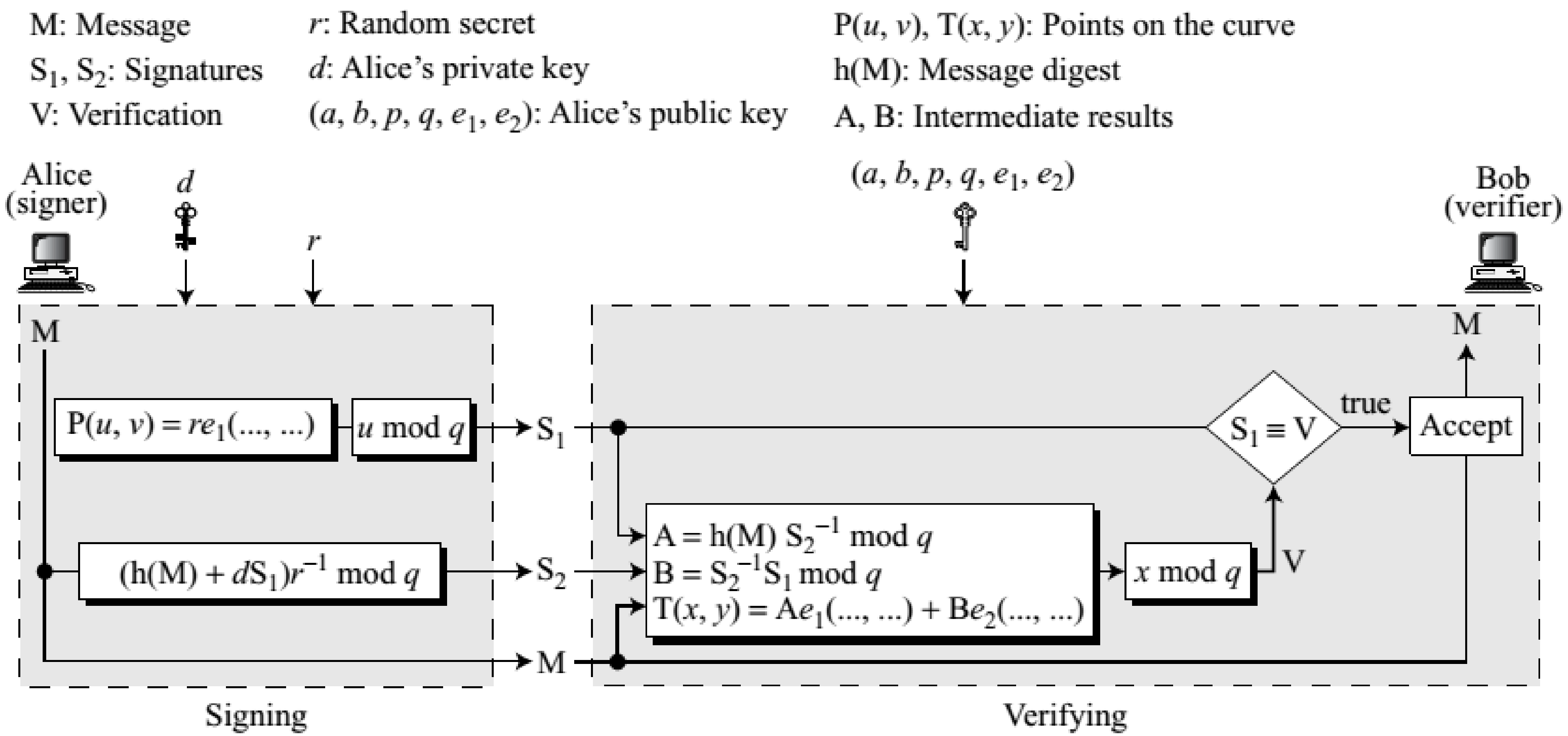
Because V_1 and V_2 are congruent, Ted accepts the message; he assumes that the message has been signed by Alice because no one else has Alice's private key, d. Note that any person can receive the message. The goal is not to hide the message, but to prove that it is sent by Alice.

Elliptic Curve Digital Signature Scheme

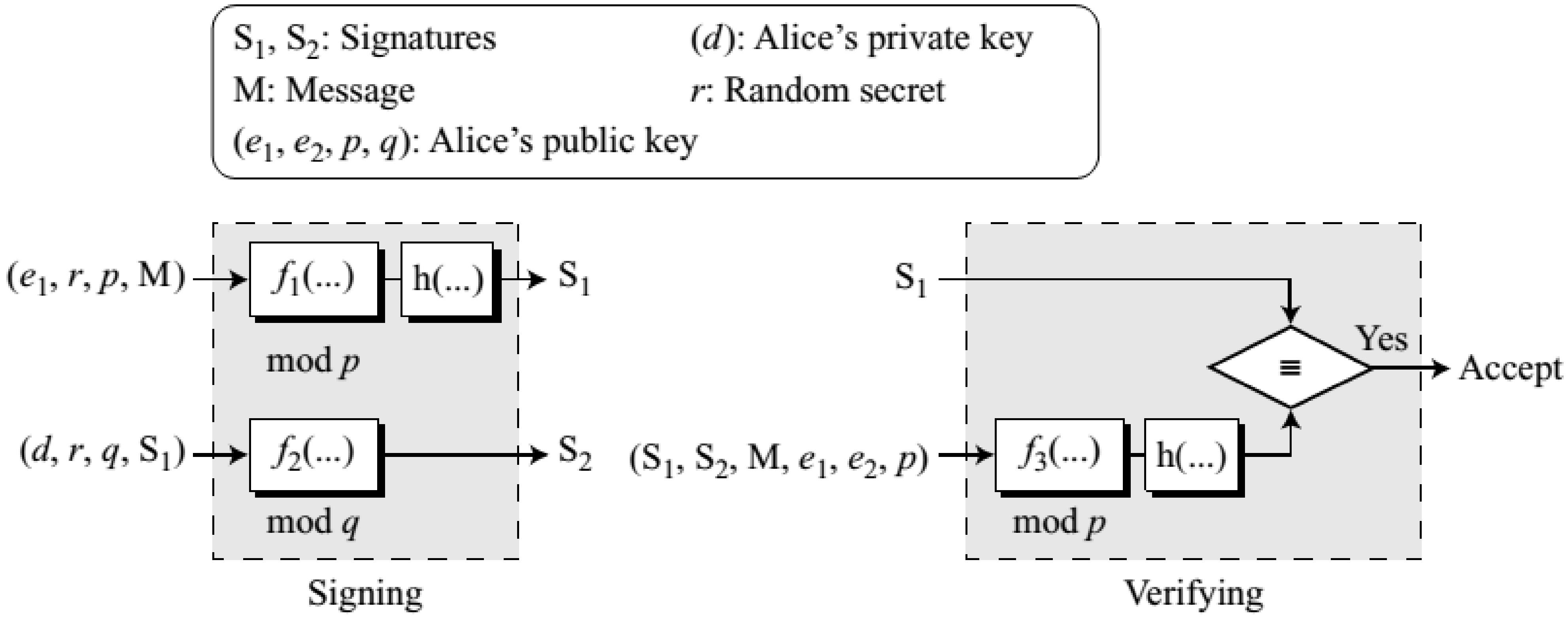
S_1, S_2: Signatures
M: Message
(a, b, p, q, e_1, e_2): Alice's public key
d: Alice's private key
r: Random secret



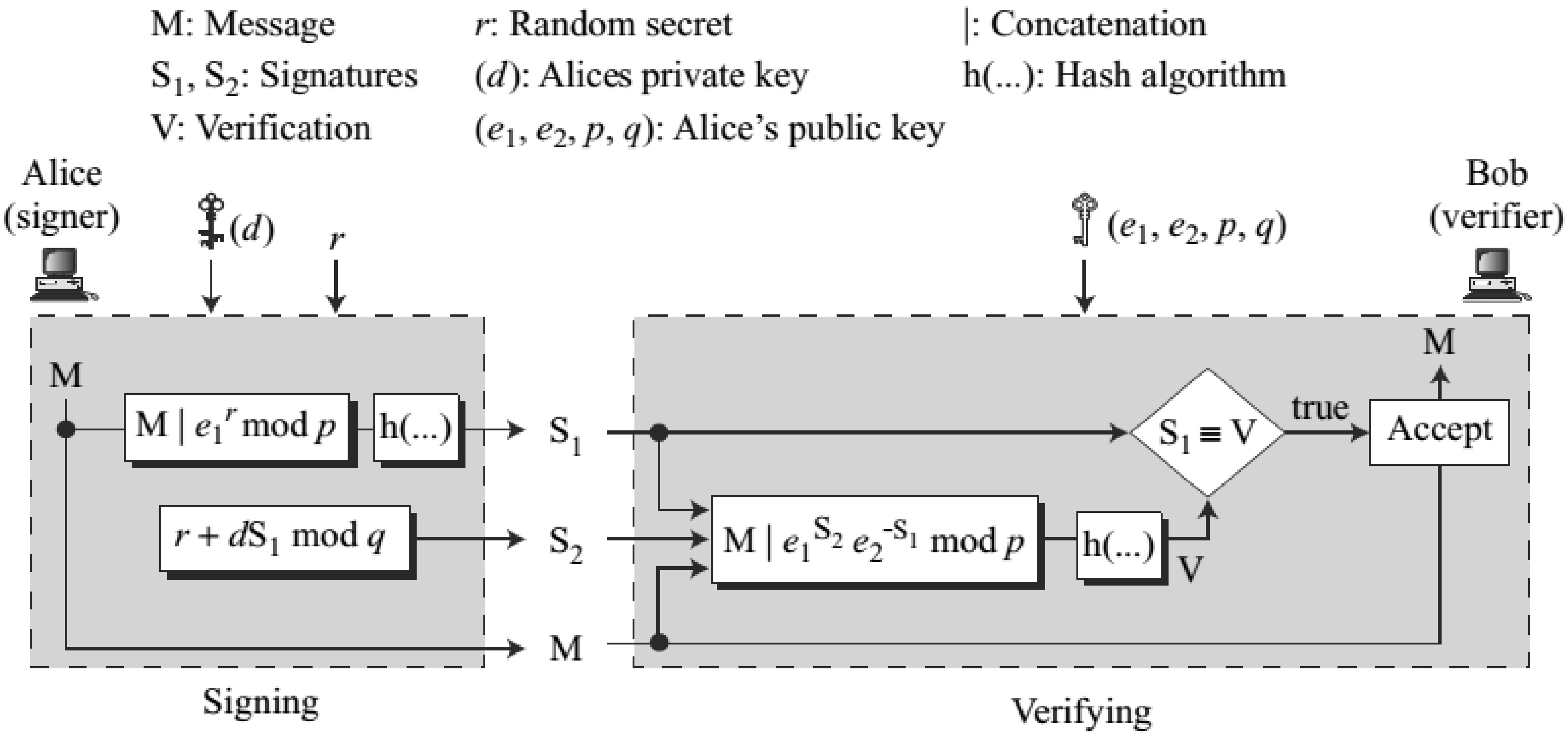
Elliptic Curve Digital Signature Scheme



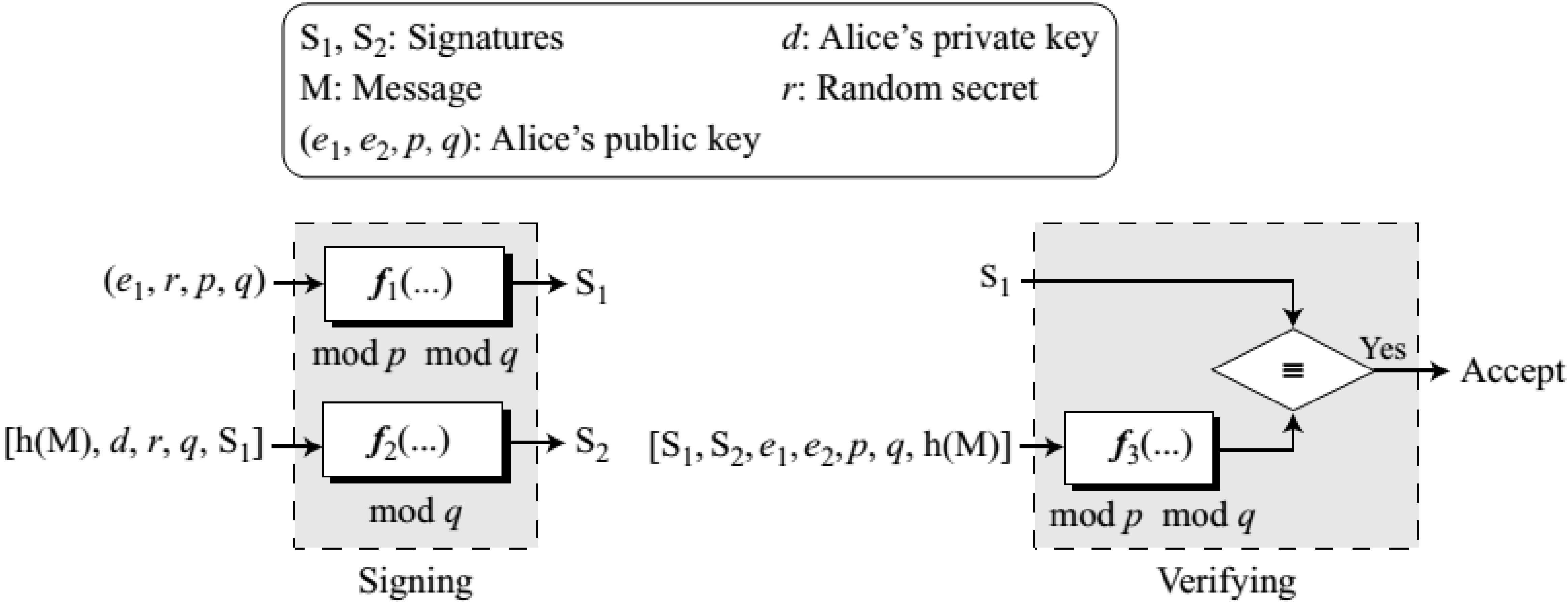
Schnorr Digital Signature Scheme



Schnorr Digital Signature Scheme



Digital Signature Standard (DSS)

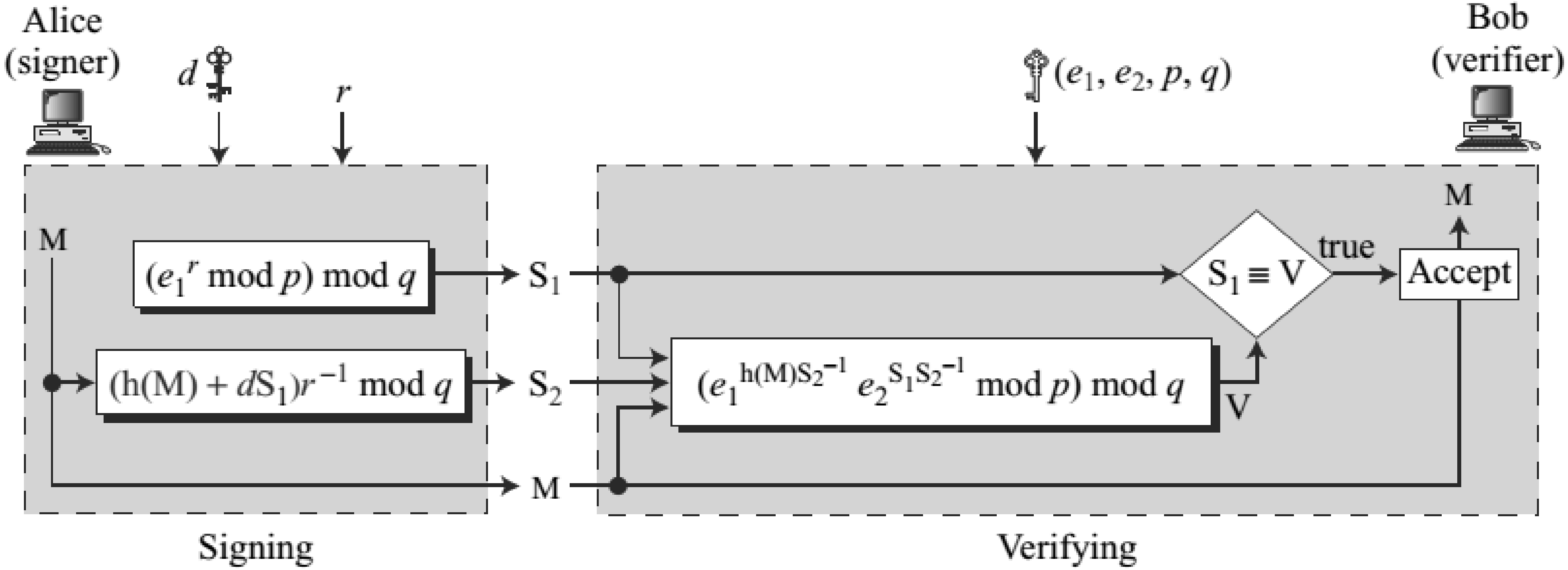


Digital Signature Standard (DSS)

M: Message — r: Random secret — h(M): Message digest

S_1, S_2: Signatures — d: Alices private key

V: Verification — (e_1, e_2, p, q): Alice's public key



Tutorial

- Explore the security attacks on schemes