SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

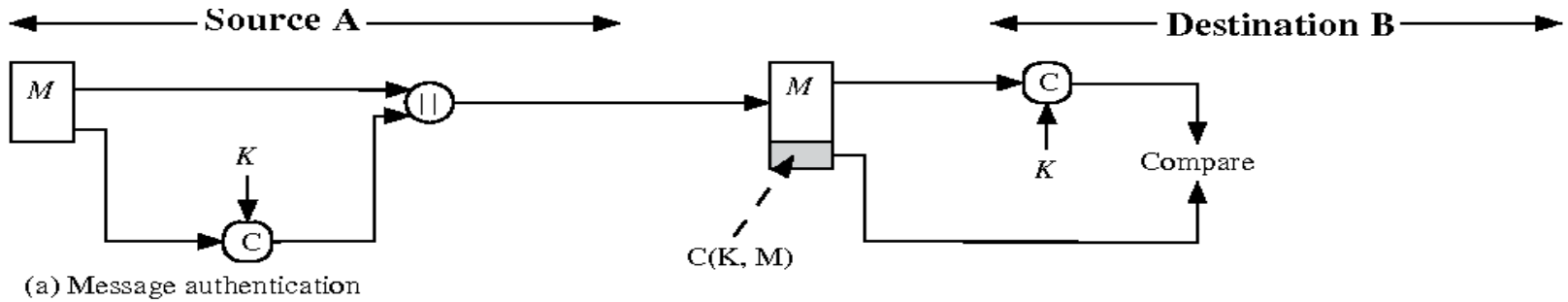
Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

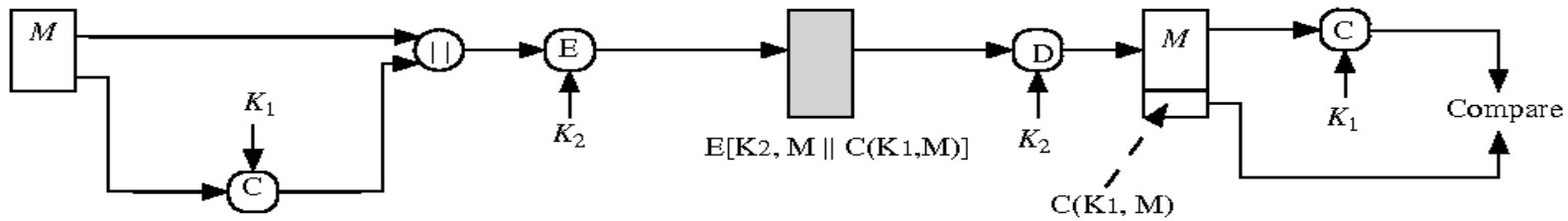
19ITB302-Cryptography and Network Security

UNIT-3 HASH FUNCTION AND DIGITAL SIGNATURE

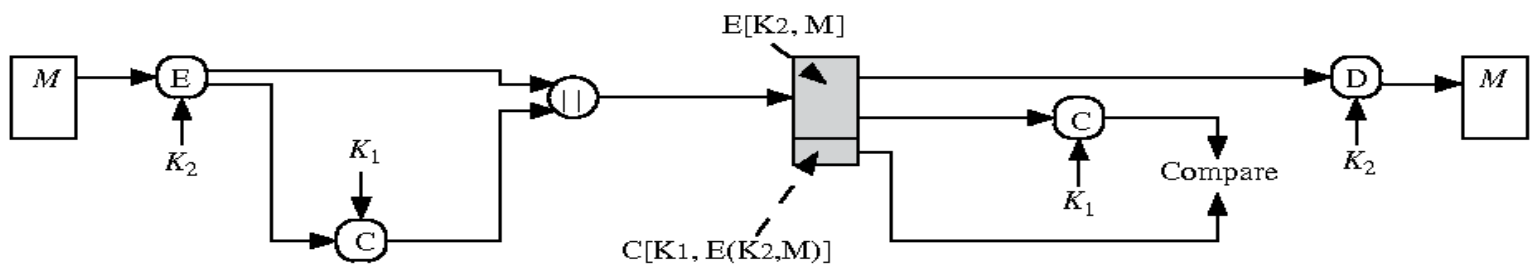
Message Authentication Code



(a) Message authentication



(b) Message authentication and confidentiality; authentication tied to plaintext



(c) Message authentication and confidentiality; authentication tied to ciphertext

REQUIREMENTS FOR MESSAGE AUTHENTICATION CODES

- The MAC is appended to the message at the source at a time when the message is assumed or known to be correct. The receiver authenticates that message by recomputing the MAC.
- If an opponent observes and, it should be computationally infeasible for the opponent to construct a message M' such that MAC (K, M') = MAC (K, M)
- MAC(K, M) should be uniformly distributed in the sense that for randomly chosen messages, M and M' , the probability that is MAC(K, M) = MAC(K, M') is 2^n, where n is the number of bits in the MAC

HMAC:

- Hash-based Message Authentication Code (HMAC) is a type of message authentication code (MAC) involving a cryptographic hash function and a secret cryptographic key. HMAC makes it possible to confirm the data integrity and authenticity of a message.
- HMAC is a great resistance towards cryptanalysis attacks as it uses the Hashing concept twice. HMAC consists of twin benefits of Hashing and MAC and thus is more secure than any other authentication code.

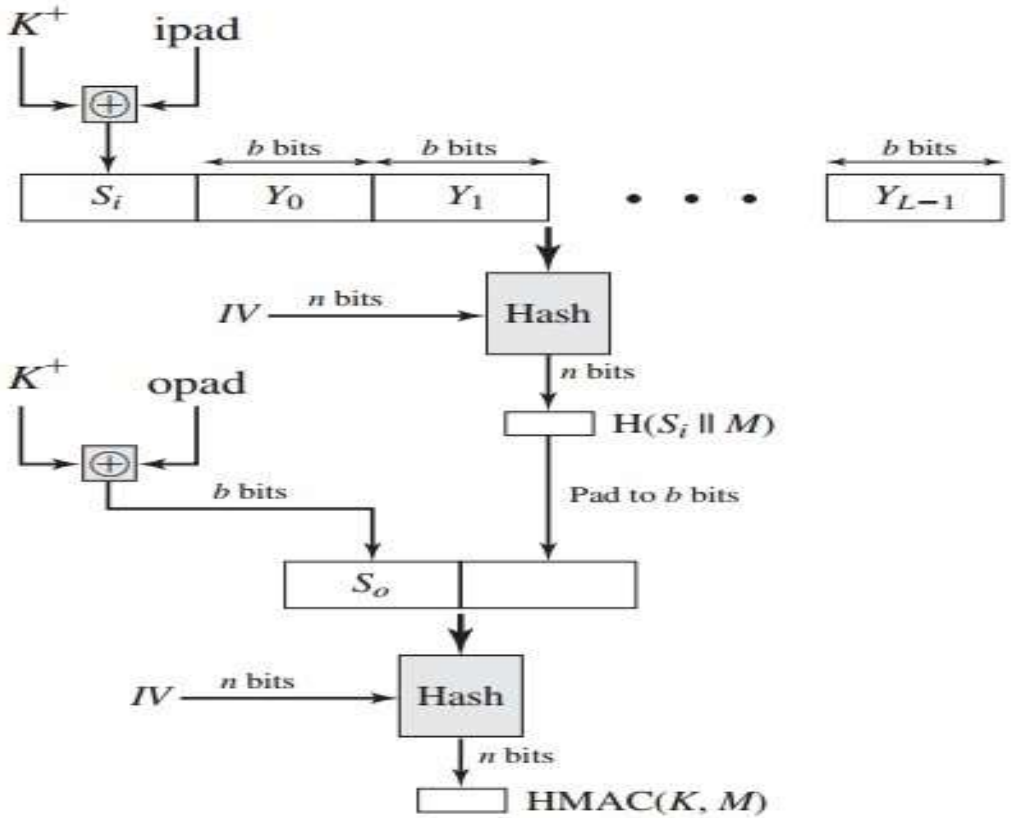


Figure 12.5 HMAC Structure

- H = embedded hash function (e.g.,MD5, SHA-1, RIPEMD-160)
- IV = initial value input to hash function
- M = message input to HMAC
- Yi = i th block of M, $0 \leq i \leq L - 1$
- L = number of blocks in M
- b = number of bits in a block
- n = length of hash code produced by embedded hash function
- K = secret key; recommended length is $\geq$ n; if key length is greater than b, the key is input to the hash function to produce an n-bit key.
- K+ = K padded with zeros on the left so that the result is b bits in length ipad = 00110110 (36 in hexadecimal) repeated b/8 times
- opad = 01011100 (5C in hexadecimal) repeated b/8 times

Digital Signature

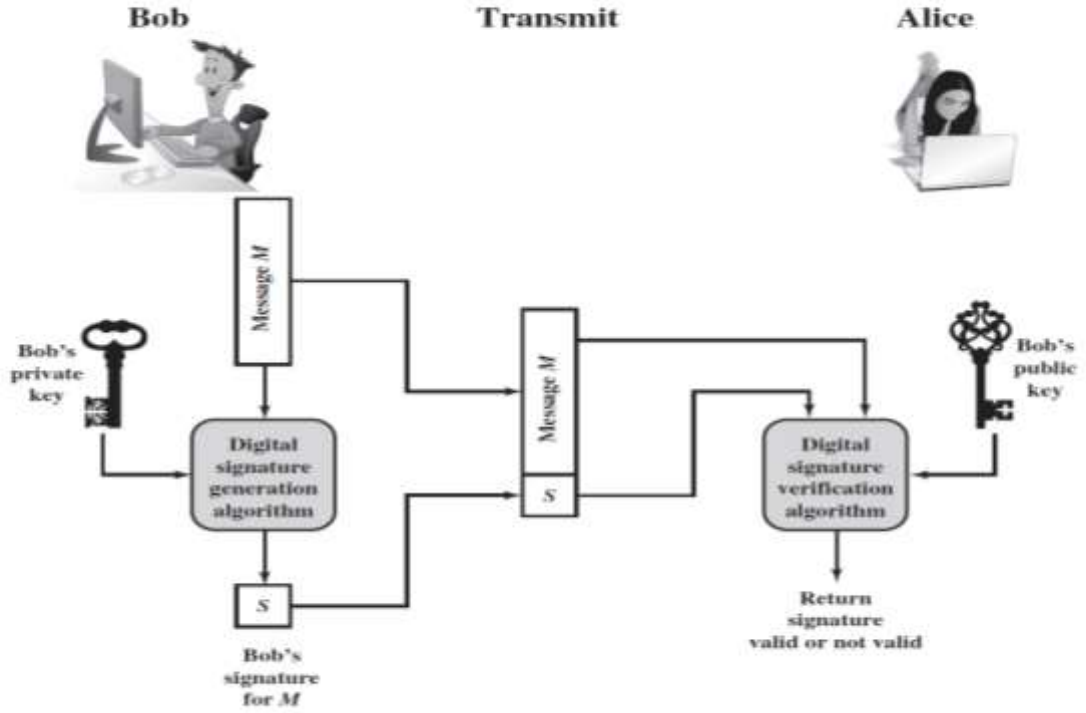


Figure 13.1 Generic Model of Digital Signature Process

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
- It must be practical to retain a copy of the digital signature in storage.

Digital Signature

- A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software, or digital document. These are some of the key features of it.

- Key Generation Algorithms: Digital signatures are electronic signatures, which assure that the message was sent by a particular sender. While performing digital transactions authenticity and integrity should be assured, otherwise, the data can be altered or someone can also act as if he were the sender and expect a reply.

- Signing Algorithms: To create a digital signature, signing algorithms like email programs create a one-way hash of the electronic data which is to be signed. The signing algorithm then encrypts the hash value using the private key (signature key). This encrypted hash along with other information like the hashing algorithm is the digital signature.
- This digital signature is appended with the data and sent to the verifier. The reason for encrypting the hash instead of the entire message or document is that a hash function converts any arbitrary input into a much shorter fixed-length value. This saves time as now instead of signing a long message a shorter hash value has to be signed and hashing is much faster than signing.

How Digital Signature Works

- The steps followed in creating a digital signature are:

- Message digest is computed by applying the hash function on the message and then message digest is encrypted using the private key of the sender to form the digital signature. (digital signature = encryption (private key of sender, message digest) and message digest = message digest algorithm (message)).
- A digital signature is then transmitted with the message. (message + digital signature is transmitted)
- The receiver decrypts the digital signature using the public key of the sender. (This assures authenticity, as only the sender has his private key so only the sender can encrypt using his private key which can thus be decrypted by the sender's public key).

ow Digital Signature Works

- The receiver now has the message digest.
- The receiver can compute the message digest from the message (actual message is sent with the digital signature).
- The message digest computed by receiver and the message digest (got by decryption on digital signature) need to be same for ensuring integrity.
- Message digest is computed using one-way hash function, i.e. a