Message Integrity, Cryptographic Hash Functions and Digital Signatures: Part 3

Gaurav S. Kasbekar
Dept. of Electrical Engineering
IIT Bombay

References

- J. Kurose, K. Ross, "*Computer Networking: A Top Down Approach*", Sixth Edition, Pearson Education, 2013
- C. Kaufman, R. Perlman, M. Speciner, "*Network Security: Private Communication in a Public World*", Pearson Education, 2nd edition, 2002
- A. Tanenbaum, D. Wetherall, "Computer Networks", Fifth Edition, Pearson Education, 2012

Digital Signatures

Objectives

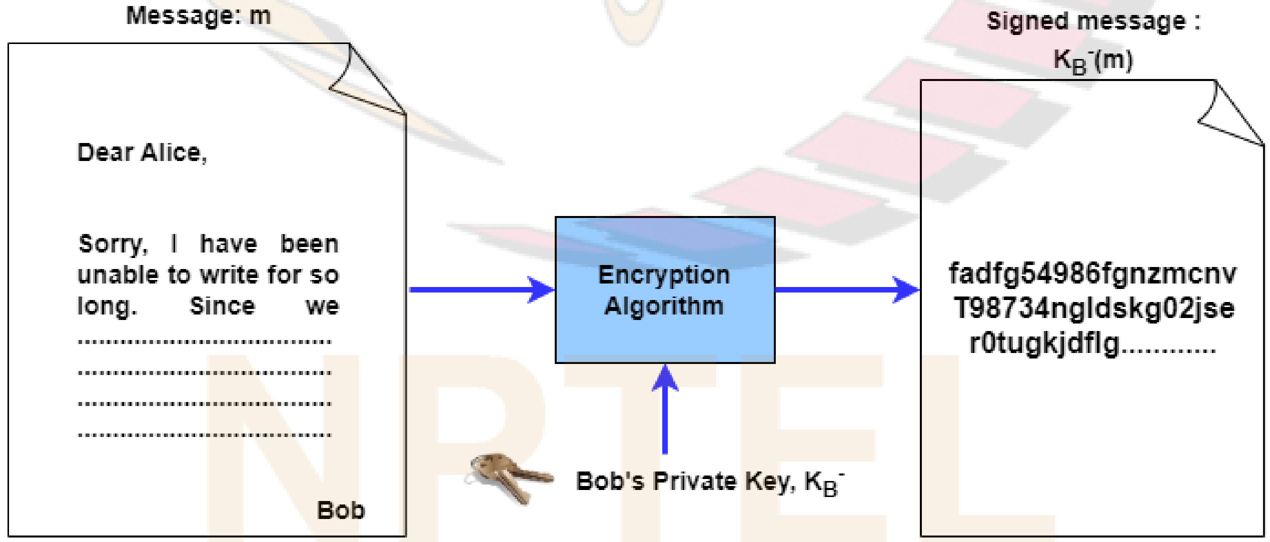
- Recall:
  - ❑manual signatures extensively used on checks, credit card receipts, legal documents, letters, etc.
  - ❑made by a person to indicate that he/ she created a document, agrees with or acknowledges its contents
- Digital signature used to achieve the same objectives for documents in digital form
- Similar to a manual signature, a digital signature must be *verifiable* and *nonforgeable, i.e.*:
  - ❑must be possible to prove that a person’s signature on a document is indeed that person’s signature (verifiability) and
  - ❑no one should be able to create a person’s digital signature except the person himself/ herself (nonforgeability)

Attempt

- To sign a message m, Bob appends a field similar to a MAC to it, *i.e.*:
  - ❑concatenates m and s, where s is a secret bit string *that only Bob knows*, to get (m, s); computes $H(m, s)$
  - ❑$(m, H(m, s))$ is the signed document
- Does this scheme achieve the objectives of a digital signature?
  - ❑No; the signature is nonforgeable, but is not verifiable
- Modified version: another user, say Alice, knows s
- Does the modified version achieve the objectives of a digital signature?
  - ❑No; the signature is verifiable only by Alice; also, it is forgeable
- Want an alternative scheme for implementing a digital signature

Implementation of a Digital Signature: Scheme 1

- Recall: if K_B^+ (respectively, K_B^-) denotes Bob's public key (respectively, private key), then:
  - ❑ $K_B^+(K_B^-(m)) = m$
- To sign a message m, Bob computes $K_B^-(m)$ and appends it to m
  - ❑ $(m, K_B^-(m))$ is the digitally signed message

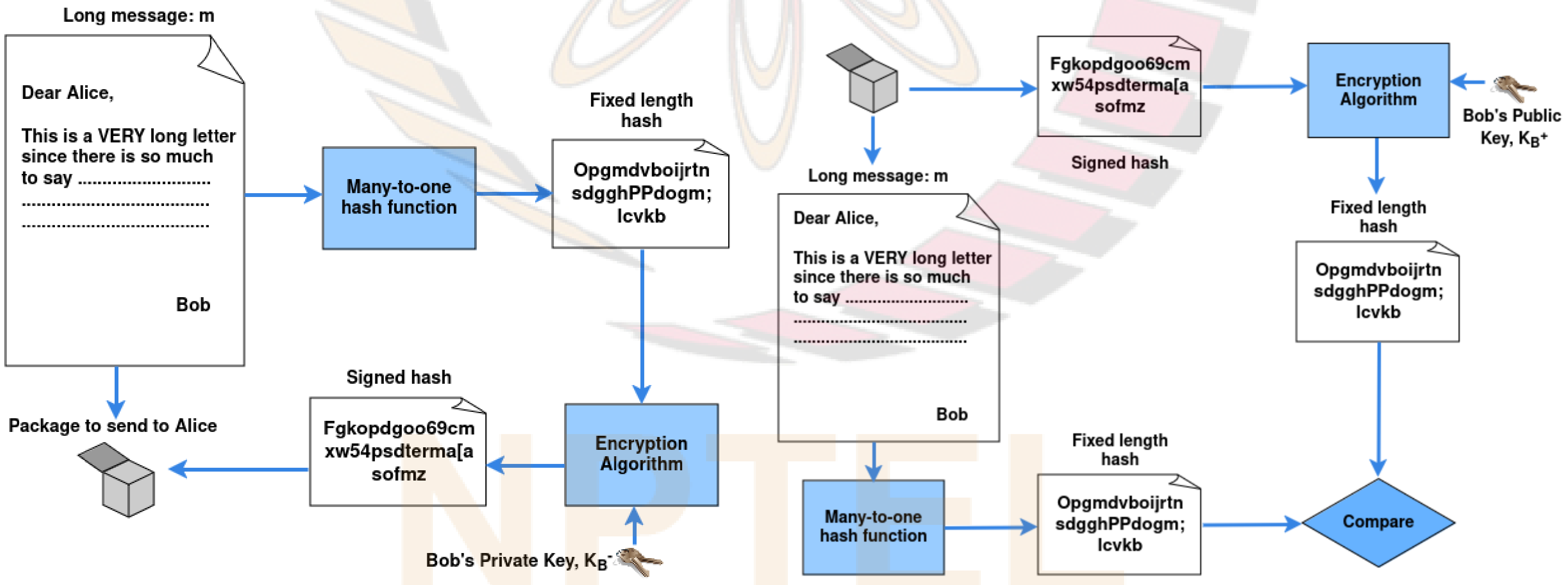


Implementation of a Digital Signature: Scheme 1 (contd.)

- Is the signature $K_B^-(m)$ verifiable and nonforgeable?
- Yes:
  - ❑ Anyone can use K_B^+ to compute $K_B^+(K_B^-(m)) = m$, which is the plaintext message; hence, verifiable
  - ❑ Knowledge of K_B^- is required to compute $K_B^-(m)$; hence, nonforgeable
- **Note**: Above argument assumes that Bob has not shared K_B^- with anyone and it has not been stolen from him
- Shortcoming of the above scheme for implementing a digital signature:
  - ❑ computationally expensive when m is long, since public key encryption/ decryption is time-consuming
- Want a more computationally efficient scheme for creating digital signature

Implementation of a Digital Signature: Scheme 2

- To sign a message m, Bob computes its hash $H(m)$, encrypts it with his private key to get $K_B^-(H(m))$ and appends $K_B^-(H(m))$ to m
  - ❑ $(m, K_B^-(H(m)))$ is the digitally signed message
- Scheme 2 also works and is computationally more efficient
- Consider the alternative scheme, where $c(m)$ is a checksum and $(m, K_B^-(c(m)))$ is digitally signed message. Is this a secure digital signature scheme?
  - ❑ No



Message Integrity

- Recall:
  - in scheme 1, $(m, K_B^-(m))$ is the digitally signed message
  - in scheme 2, $(m, K_B^-(H(m)))$ is the digitally signed message
- Which of these schemes, if any, achieves message integrity?
  - Both; due to verifiability, the fact that $K_B^+(K_B^-(m)) \neq m'$ for $m \neq m'$ and computational infeasibility of finding $m' \neq m$ such that $H(m') = H(m)$

MAC vs Digital Signature for Achieving Message Integrity

- Recall: message integrity of a message m can be achieved using a MAC or a digital signature
- Pros and cons:
  - ❑Digital signature requires encryption, which is time consuming; MAC does not
  - ❑MAC requires sender and receiver to have a shared secret (authentication key); digital signature does not