Message Authentication Codes (MAC)

- A keyed checksum of the message.
- Sender of a message $M$ computes $c = \text{MAC}_K(M)$ and sends $(M, c)$ to the receiver.
- Receiver also computes $c' = \text{MAC}_K(M)$. If $c' = c$, the message is accepted.
- Example applications:
  - protecting files on an OS against modification
  - authentication of routing messages

MACs (cont’d)

- A MACed message is not necessarily encrypted.
- MAC function doesn’t need to be invertible.
- MAC keys are symmetric. Hence, doesn’t provide non-repudiation. (unlike digital signatures)
- Security of a MAC: An attacker shouldn’t be able to generate a valid $(M', c')$ pair, even after seeing many valid message-MAC pairs possibly of his choice (i.e. by a chosen message attack).
MAC from a Block Cipher

How to obtain a MAC from a block cipher?

Suggestion:
– divide message into blocks
– compute a checksum by adding (or xoring) them
– encrypt the checksum with the block cipher

Is this construction secure?
– If the message is not encrypted?
– If the message is encrypted?

CBC-MAC

• Simple CBC-MAC:
  – Compute the CBC over the message with IV = 0.
  (Q: Why not a random IV?)
  – The last output block is the MAC

Other alternatives:
– ECB?
– OFB/CTR?
– CFB?

CBC-MAC in Practice

Simple CBC-MAC is not exactly secure as a MAC. It has two popular flavors:

• CMAC (authentication only)
  – CBC-MAC with some extra processing at the end
  – Recommended by NIST SP 800-38B

• CCM (both encryption & authentication)
  – Counter mode encryption with CBC-MAC
  – Recommended by NIST SP 800-38C
  – Used in WPA2