

Computer Security

CS 426

Lecture 28

Message Authentication Code, Review of Homework 4, and Dynamic Credit Card Numbers

Announcements

- Plan for December 6:
 - DBMS Security
 - Topics cover in final exam

Encryption vs. Message Authentication Code (MAC)

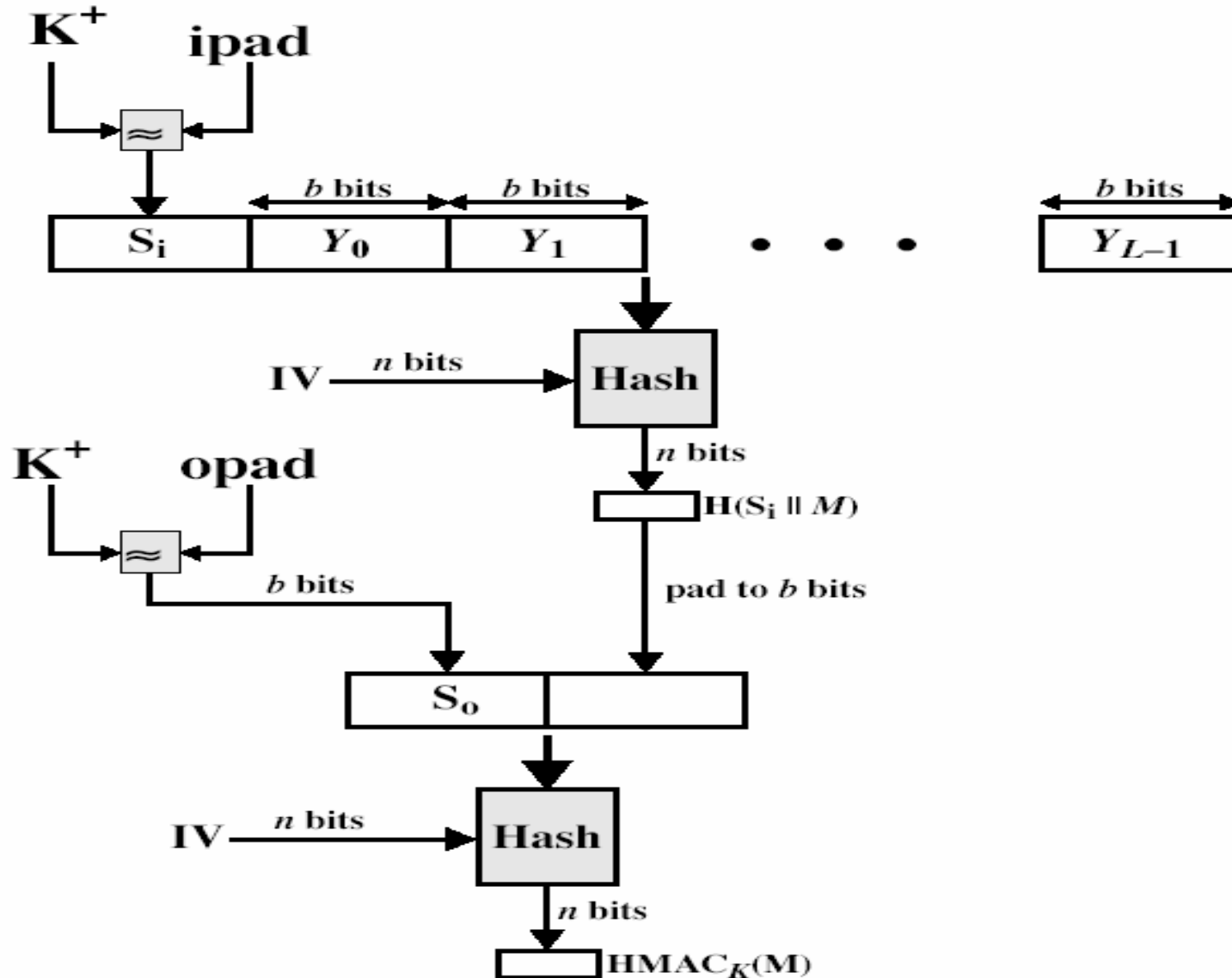
- Encryption protects confidentiality and MAC protects authenticity and integrity
 - From $C = E_K(M)$, one cannot recover M , but one may be able to find C' , M' such that $C' = E_K(M')$
 - From $T = \text{MAC}_K(M)$, one may be able to recover M , but one must not be able to find M' , T' such that $T' = \text{MAC}_K(M')$
- Both require the sender and the receiver share K
- A simple, insecure MAC
 - $\text{MAC}_K(M) = H(K \parallel M)$, where H is a cryptography hash function

HMAC

$$\text{HMAC}_K[M] = \text{Hash}[(K^+ \oplus \text{opad}) \parallel \text{Hash}[(K^+ \oplus \text{ipad}) \parallel M]]$$

- K^+ is the key padded (with 0) to B bytes, the input block size of the hash function
- ipad = the byte $0x36$ repeated B times
- opad = the byte $0x5C$ repeated B times.

HMAC Overview



CBC-MAC

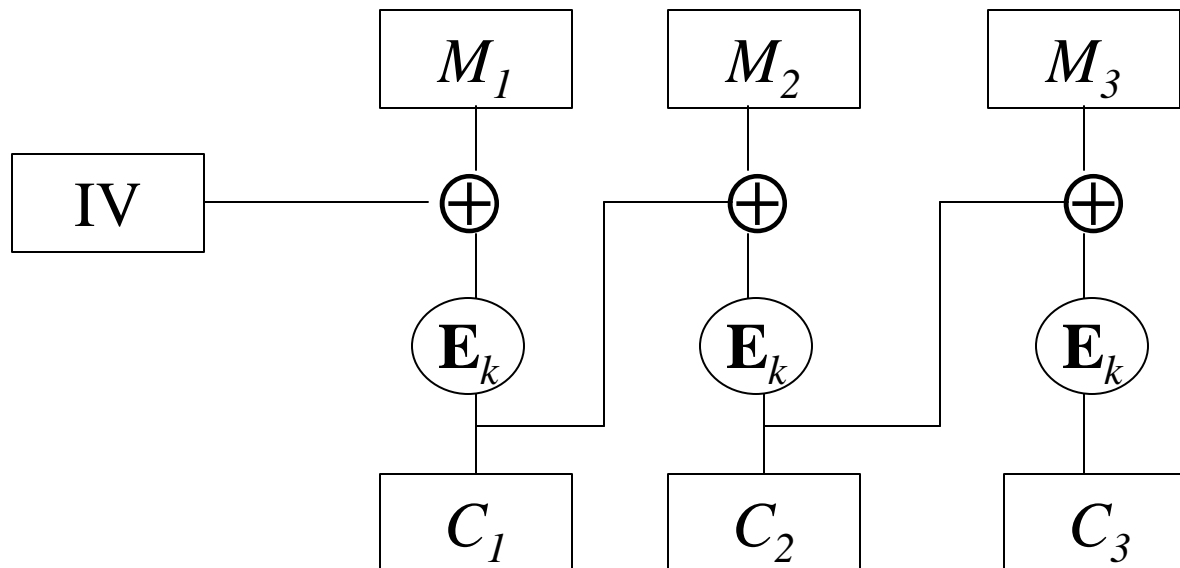
- Given a block cipher \mathbf{E} with block size m
- Given message $M = M_1 \parallel M_2 \parallel \dots \parallel M_n$
- MAC of M is the last ciphertext block of $\mathbf{E}_k(M)$
 - $z_0 = IV = 0^m$
 - $z_i = \mathbf{E}_k(z_{i-1} \oplus M_i)$ for $1 \leq i \leq n$
 - $MAC = z_n$
- No need to use random IV here.

Encryption Modes: CBC

- **Cipher Block Chaining (CBC):** next input depends of previous output

- Plaintext is $M_1, M_2, M_3, M_4,$

- Ciphertext is: $C_1 = IV \oplus \mathbf{E}_k(M_1)$ $C_2 = C_1 \oplus \mathbf{E}_k(M_2)$
 $C_3 = C_2 \oplus \mathbf{E}_k(M_3)$ $C_4 = C_3 \oplus \mathbf{E}_k(M_4)$



Properties of CBC-MAC

- Secure for messages of a fixed number of blocks assuming the block cipher is secure
- Not secure with variable lengths
- Slower than HMAC

Design of ATM

- What are the information
 - account number: routing number + account number
 - PIN
 - salt
 - balance
- Trusting relationship among bank and ATM providers
 - who can share keys with whom
- Encryption vs. Message Authentication

Bank Storage & ATM Design

- Bank Storage
 - account number, balance in clear
 - PIN is not stored in clear,
 - stores $v = \text{MAC}_K(\text{acc}) \oplus H(\text{acc} \parallel \text{PIN})$
- ATM (no offline operation requirement)
 - card stores acc
 - send acc, $p = H(\text{acc} \parallel \text{PIN})$ to bank
 - bank checks whether $p \oplus \text{MAC}_K(\text{acc}) = v$ and report to ATM

ATM Design (online operation)

- Design 1 (no balance checking)
 - ATM must be able to validate PIN
 - Card stores: acc , $E_K(H(\text{acc} \parallel \text{PIN}))$, $E_{K_1}[K]$, $E_{K_2}[K], \dots$
 - ATM is able to recover K , can recover $H(\text{acc} \parallel \text{PIN})$
 - how to remedy weakness
- Design 2 (maintaining balance on card)
 - what are the security issues?
- Design 3 (ask card to check PIN)
 - why this does not make sense?

Code That Reproduce Itself

- Key:
 - contains two parts that are essentially the same (one part is data, the other part is code)

See Slides on
Dynamic Credit
Card Numbers

Coming Attractions ...

- December 6:
 - DBMS Security
 - Review of topics cover in final exam

