Assignment #3

Symmetric Cryptography

Due: October 25, 2016

1. Suppose that a long message has been encrypted in CBC mode with a block cipher. You know the key, and want to decrypt a middle section where the beginning and end bit positions of the plaintext don’t align at the cipher’s block length as depicted below. The block length is k bits and . The plaintext to recover starts at *bit i* of the second block, and ends at *bit j* of the fifth block, inclusive. Which blocks would you need to decrypt and what other operations would you need to recover the plaintext? Write expressions to recover the plaintext.

C0

C1

C2

C3

C4

C5

C6

C7

Bit j

Bit i

1. Suppose Alice and Bob have accounts at the same bank. Alice shares a secret key K with the bank, wants to send $100 to Bob, and knows Bob’s identity at the bank (something like Chase Quick Pay at <https://www.chase.com/online/digital/quickpay.html>). Goal: Prevent an adversary from discovering the amount of money, and the source and destination account identifiers. Using symmetric cryptographic operations,
   1. Specify a valid request from Alice to the bank to initiate the money transfer. *Hint: The message must include Alice’s identity, Bob’s identity, and the amount to send.*
   2. Specify a valid response from the bank to Alice about the status (sent with transaction number and remaining balance, not sent).
   3. In this context, a “replay attack” allows an adversary to successfully send a previous money transfer message without breaking any cryptography to initiate the same money transfer one or more times. Update your request and reply messages to thwart replay attacks.
2. Counter mode (CTR) is a mode of operation for block ciphers where an encrypted counter is XORed with the plaintext M to produce a ciphertext C. Initialize: Counter = IV; Encrypt/Decrypt: { C = M XOR E(Counter); Counter = Counter + 1 }. Describe what might go wrong if the counter is reused with the same key. Describe a potential solution.
3. Suppose that there is a database of encrypted Social Security Numbers (SSN). Encryption uses AES-CBC with a 256-bit key and IV = 0xa5a5a5a5a5a5a5a5, and the key is not compromised. There is a function that accepts a SSN, encrypts it, and writes the encrypted SSN to the database: *AddSSN(ssn)*. Describe an attack to reveal a SSN in the database if an attacker has read-only access to the database. Describe a countermeasure.
4. Build a one-way hash function with RSA. *Hint: You may want to think about RSA as a block cipher.* Describe if you prefer that hash function over SHA-256 in a practical application.