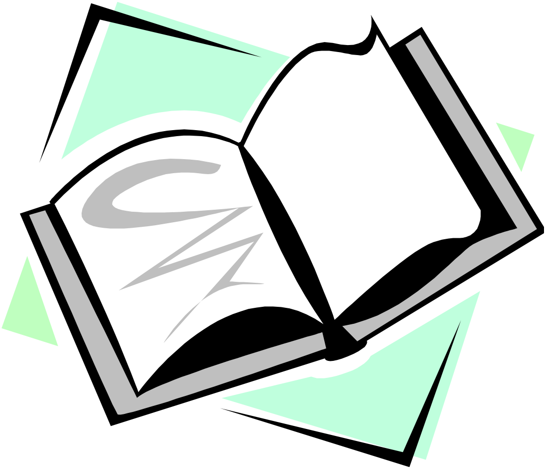
Cryptography
CS 555

Topic 11: Encryption Modes and CCA Security

Outline and Readings

- Outline
  - Encryption modes
  - CCA security
- Readings:
  - Katz and Lindell: 3.6.4, 3.7



Review: IND Security

- An encryption scheme Π = (Gen, Enc, Dec) has indistinguishable encryptions in the presence of an eavesdropper if for all PPT adversary *A*, there exists a negligible function negl such that
  - $\Pr[\mathbf{PrivK}^{eav}_{A,\Pi}=1] \leq \frac{1}{2} + \mathrm{negl}(n)$

  – *A* outputs a pair of equal-length messages m_0 and m_1

  – *A* is given the challenge ciphertext $\mathrm{Enc}_k(m_b)$

  - Where b is chosen at uniform random from {0,1}

  – *A* outputs b'

  – $\mathbf{PrivK}^{eav}_{A,\Pi}=1$ when b=b'

Review: CPA-secure (aka IND-CPA security)

- Π has indistinguishable encryption under a chosen-plaintext attack iff. for all PPT adversary *A*, there exists a negligible function negl
  - $\Pr[\mathbf{PrivK}^{\mathbf{cpa}}_{A,\Pi}=1] \leq ½ + \text{negl}(n)$
  - – *A* is given oracle access to $\text{Enc}_k(\cdot)$, and outputs a pair of equal-length messages m_0 and m_1
  - – *A* is given the challenge ciphertext $\text{Enc}_k(m_b)$
    - Where b is chosen at uniform random from {0,1}
  - – *A* still has oracle access to $\text{Enc}_k(\cdot)$, and (after some time) outputs b’
  - – $\mathbf{PrivK}^{\mathbf{cpa}}_{A,\Pi}=1$ when b=b’

Review: Pseudorandom Permutations (PRP)

- We say that a length-preserving keyed function F: $\{0,1\}^k \times \{0,1\}^* \rightarrow \{0,1\}^*$, is a keyed permutation if and only if each F_k is a bijection
- A Pseudorandom Permutation (PRP) is a keyed permutation that is indistinguishable from a random permutation
- A Strong PRP is a keyed permutation is indistinguishable from a random permutation when the distinguisher is given access to both the function and its inverse
- We assume block ciphers are PRP.

Need for Encryption Modes

- A block cipher encrypts only one block
- Needs a way to extend it to encrypt an arbitrarily long message
- Want to ensure that if the block cipher is secure, then the encryption is secure
- Aims at providing **CPA security** assuming that the underlying block ciphers are strong

Block Cipher Encryption Modes: ECB

- Message is broken into independent blocks;
- Electronic Code Book (ECB): each block encrypted separately.
- **Encryption: $c_i = E_k(x_i)$**
- **Decrytion: $x_i = D_k(c_i)$**

Properties of ECB

- Deterministic:
  - the same data block gets encrypted the same way,
    - reveals patterns of data when a data block repeats
  - when the same key is used, the same message is encrypted the same way
- How to show that ECB is not CPA-secure?
- How to show that ECB is not IND-secure (even in a ciphertext only attack)?

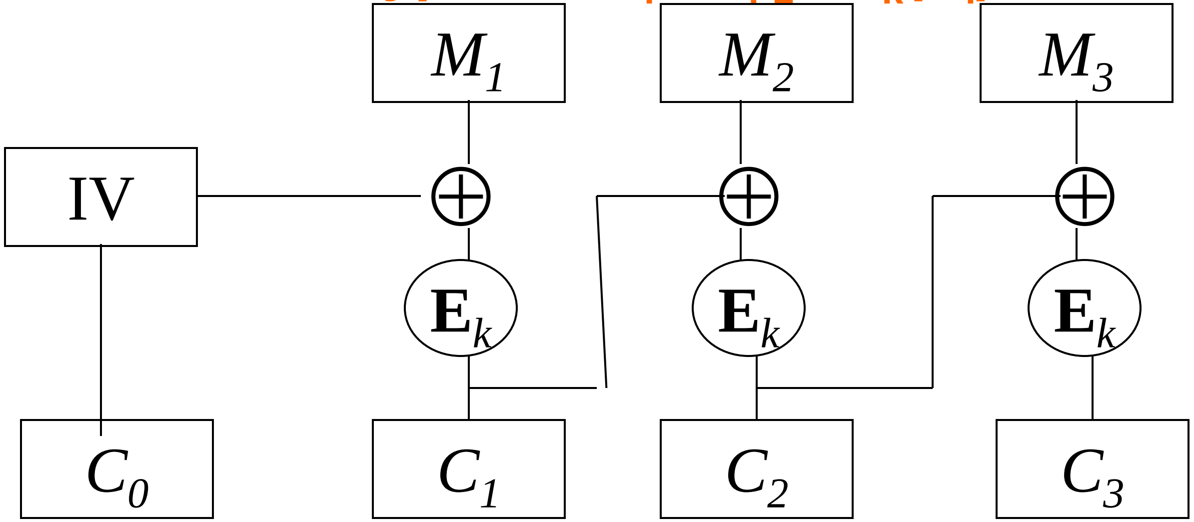
- Usage: Should not be used.

Encryption Modes: CBC

- Cipher Block Chaining (CBC):
  - Uses a random Initial Vector (IV)
  - Next input depends upon previous output

Encryption: $C_i = E_k(M_i \oplus C_{i-1})$, **with** $C_0 = IV$

Decryption: $M_i = C_{i-1} \oplus D_k(C_i)$, **with** $C_0 = IV$



Properties of CBC

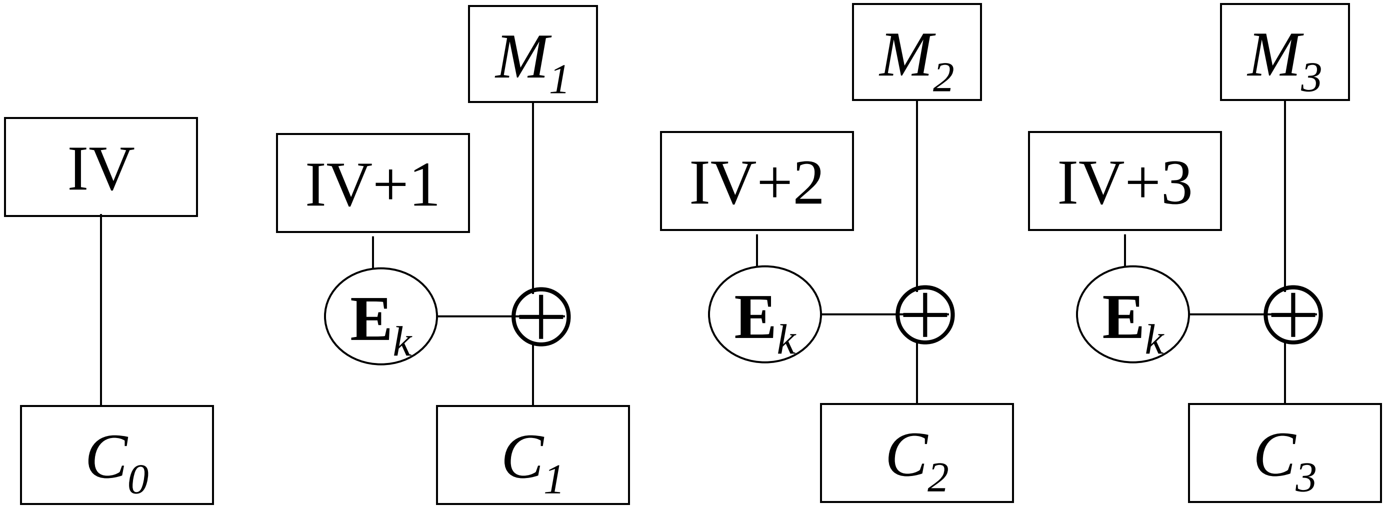
- Randomized encryption: repeated text gets mapped to different encrypted data.
  - Is CPA secure assuming that the block cipher is secure (i.e., it is a Pseudo Random Permutation (PRP))
- Each ciphertext block depends on all preceding plaintext blocks.
- Usage: chooses random IV and protects the **integrity** of IV
  - The IV is not secret (it is part of ciphertext)
  - The adversary cannot control the IV

Encryption Modes: OFB

- Output feedback (OFB):
  - construct a PRNG using a Block Cipher
  - IV is randomly chosen
  - y_0=IV $y_i = E_k[y_{i-1}]$
  - Use the stream y_1, y_2,… to XOR with message
  - Randomized encryption
  - Provides CPA-secure encryption with a PRF
  - Sequential encryption, can preprocess

Encryption Modes: CTR

- Counter Mode (CTR): Defines a stream cipher using a block cipher
  - Uses a random IV, known as the counter
  - Encryption: C_0=IV, $C_i = M_i \oplus E_k[IV+i]$
  - Decryption: IV=C_0, $M_i = C_i \oplus E_k[IV+i]$



Properties of CTR

- Gives a stream cipher from a block cipher
- Randomized encryption:
  - when starting counter is chosen randomly
- Random Access: encryption and decryption of a block can be done in random order, very useful for hard-disk encryption.
  - E.g., when one block changes, re-encryption only needs to encrypt that block. In CBC, all later blocks also need to change

Theorem 3.29:

- CTR mode provides CPA-secure encryption with a block cipher that is a PRF.
- Proof.
  - When a true random function is used, ciphertext leaks no information about plaintext unless some string in the sequence IV+1,…IV+ℓ overlaps with some sequence used for encrypting other messages.
  - Let q(n) be the bound on number of messages encrypted, as well as bound on size of messages.
  - Prob that an overlap occurs is less than $2q(n)^2/2^n$, which is negligible

Block Length and Security

- Adversary success probability depends on block size
- For block size 64, this is ½ + $q^2/2^{63}$,
- The advantage $q^2/2^{63}$ can be significant

Stream Cipher vs Block Cipher

- Stream cipher (e.g., RC4)
- Block cipher (AES)

- In software encryption, RC4 is twice as fast as AES
- Security for Block cipher (AES) is much better understood than stream cipher (RC4)
- Use AES unless in really constrained environment

The CCA Indistinguishablility Experiment: **PrivK^{cca}**(n)

- A k is generated by Gen(1^n)
- Adversary is given oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$, and outputs a pair of equal-length messages m_0 and m_1
- A random bit b is chosen, and adversary is given $c \leftarrow Enc_k(m_b)$
  - Called the challenge ciphertext
- Adversary still has oracle access to $Enc_k(\cdot)$ and $Dec_k(\cdot)$; however, Adversary cannot ask for $Dec_k(c)$.
- Adversary outputs b'
- **PrivK^{cca}**(n) = 1 if b=b' (adversary wins) and =0 otherwise

Existing Schemes are not CCA Secure

- How to break CTR mode's CCA security?
- How to break CBC mode's CCA security?
- Non-malleability
  - Cannot change the ciphertext while predicting what changes in decrypted plaintext will be.
- CCA-secure implies non-malleability
- How to build CCA-secure encryption scheme?
  - Make sure that ciphertext cannot be changed.
  - Any change will result in decryption not outputing the message.

Coming Attractions …

- More on Number Theory
- Reading: Katz & Lindell: 7.1.3, 7.1.4, 7.1.5, 7.2