Cryptography
CS 555

Topic 8: Modes of Encryption, The Penguin and CCA security

Reminder: Homework 1

- Due on **Friday** at the **beginning** of class

- Please typeset your solutions

Recap

- Pseudorandom Functions
- CPA-Security

Today's Goals:

- Evaluate several modes of operation for stream-ciphers + block-ciphers
- Introduce Chosen Ciphertext Attacks/CCA-Security
- ~~Construct encryption scheme with CCA-Security~~

Chosen Ciphertext Attacks

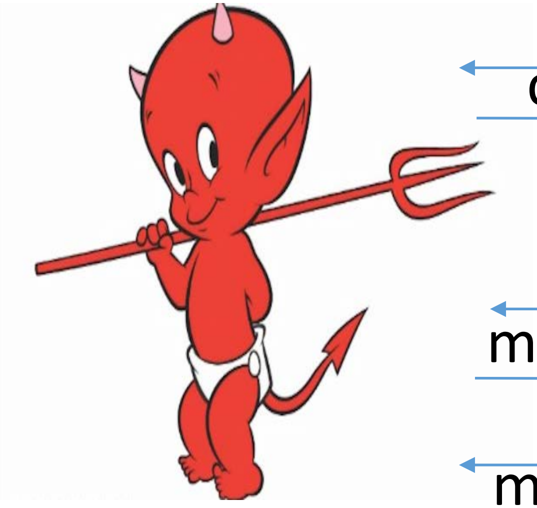
- Sometimes an attacker has ability to obtain (partial) decryptions of ciphertexts of its choice.
- CPA-Security does not model this ability.

Examples:

- An attacker may learn that a ciphertext corresponds to an ill-formed plaintext based on the reaction (e.g., server replies with “invalid message”).
- Monitor enemy behavior after receiving and encrypted message.
- **Authentication Protocol:** Send $Enc_k(r)$ to recipient who authenticates by responding with r.

CCA-Security (Ind-CC

We could set $m_0 = m_{-1}$ or $m_1 = m_{-2}$



m_{-1}

$c_{-1} = \mathrm{Enc}_K(m_{-1})$

c_{-2}

$m_{-2} = \mathrm{Dec}_K(c_{-2})$

⋮

m_0, m_1

$c = \mathrm{Enc}_K(m_b)$

m_3

$c_3 = \mathrm{Enc}_K(m_3)$

c_3

$m_3 = \mathrm{Dec}_K(m_3)$

$c_4 = c$

⋮

"No Way!"

b'

However, we could still flip 1 bit of c and ask challenger to decrypt

Random bit b

K = Gen(.)

CCA-Security $\left(PrivK_{A,\Pi}^{cca}(n)\right)$

1. Challenger generates a secret key k and a bit b
2. Adversary (A) is given oracle access to Enc_k and Dec_k
3. Adversary outputs m_0, m_1
4. Challenger sends the adversary $c = Enc_k(m_b)$.
5. Adversary maintains oracle access to Enc_k and Dec_k ,however the adversary is not allowed to query $Dec_k(c)$.
6. Eventually, Adversary outputs b'.

$$PrivK_{A,\Pi}^{cca}(n) = 1 \text{ if } \mathrm{b} = \mathrm{b}'; \text{otherwise } 0.$$

CCA-Security: For all PPT A exists a negligible function negl(n) s.t.

$$\Pr\left[PrivK_{A,\Pi}^{cca}(n) = 1\right] \leq \frac{1}{2} + negl(n)$$

CCA-Security

Definition 3.33: An encryption scheme Π is CCA-secure if for all PPT A there is a negligible function negl(n) such that

$$\Pr\left[PrivK_{A,\Pi}^{cca}(n) = 1\right] \leq \frac{1}{2} + negl(n)$$

CPA-Security doesn't imply CCA-Security

$$\text{Enc}_k(m) = \langle r, F_k(r) \oplus m \rangle$$

Attacker: Selects $m_0 = 0^n$ and $m_1 = 1^n$

Attacker Receives: $c = \langle r, s \rangle$ where $s = F_k(r) \oplus m_b$

Attacker Queries: $\text{Dec}_k(c')$ for

$$c' = \langle r, s \oplus 10^{n-1} \rangle$$

Attacker Receives: 10^{n-1} (if b=0) or 01^{n-1} (if b=1)

Example Shows: CCA-Security doesn't imply **CCA1** Security (Why?)

Attacks in the Wild

- Padding Oracle Attack
- Length of plaintext message must be multiple of block length
- Popular fix PKCS #5 padding
  - 4 bytes of padding (0x04040404)
  - 3 bytes of padding (0x030303)
- “Bad Padding Error”
  - Adversary submits ciphertext(s) and waits to if this error is produced
  - Attacker can repeatedly modify ciphertext to reveal original plaintext piece by piece!

Example

M="hello…please keep this message secret"+0x030303

$$\mathrm{C} = \langle r, s = F_k(r) \oplus m \rangle$$

- $\mathrm{C}' = \langle r, F_k(r) \oplus m \oplus$0x0000 … .30000 $\rangle$

Ask to decrypt C'

- If we added < 3 bits of padding C' can be decrypted.
- Otherwise, we will get a decryption error.

Once we know we have three bits of padding we can set

$$\mathrm{C}'' = \langle r, s = F_k(r) \oplus \text{0x0000 … .30303} \oplus \text{0x0 …}\, \mathbf{gg}040404 \rangle$$

If C'' decrypts then we can infer the last byte "t" from $\mathbf{gg} \oplus 0\mathrm{x}04$.

CCA-Security

- **Gold Standard**: CCA-Security is strictly stronger than CPA-Security
- If a scheme has indistinguishable encryptions under one chosen-ciphertext attack then it has indistinguishable multiple encryptions under chosen-ciphertext attacks.
- None of the encryption schemes we have considered so far are CCA-Secure ☹
- CCA-Security implies non-malleability (message integrity)
  - An attacker who modifies a ciphertext c produces c’ which is either
    - Invalid, or
    - Decryptions to unrelated message

Back to CPA-Security

- We will build a CCA-Secure Encryption scheme later in the course
  - We will need to introduce additional tools (Message Authentication Codes)
- Remaining Lecture: Modes of Operation for Stream-Ciphers and Block-Ciphers

CPA-Secure Encryption

$$\text{Enc}_k(\text{m}) = \langle r, F_k(r) \oplus m \rangle$$

$$\text{Dec}_k(\langle r, s \rangle) = F_k(r) \oplus s$$

Drawbacks:

- Encryption is for fixed length messages only
- Length of ciphertext is twice as long as message

Stream Ciphers/Block Ciphers

- Attacker can still tamper with ciphertexts to flip bits of plaintext

CCA Security

Stream Ciphers Modes

- What if we don't know the length of the message to be encrypted a priori?
  - Stream Cipher: $G_\infty(s, 1^n)$ outputs n pseudorandom bits as follows
  - **Initial State**: st_0 = **Initialize**(s)
  - **Repeat**
    - (y_i, st_i)=GetBits(st_{i-1})
    - Output y_i
- **Synchronized Mode**
  - Message sequence: $m_1, m_2, \ldots$
  - Ciphertext sequence: $c_i = m_i \oplus y_i$ (same length as ciphertext!)
  - "CPA-like" security follows from cipher security (must stop after n-bits)
  - Deterministic encryption, what gives???
  - Requires both parties to maintain state (not good for sporadic communication)

Stream Ciphers Modes

- What if we don't want to keep state?
- **Unsynchronized Mode**
  - Message sequence: $m_1, m_2, \ldots$
  - Ciphertext sequence: $c_i = \langle \mathrm{IV}, m_i \oplus G_\infty(s, IV, 1^{|m_i|}) \rangle$
  - CPA-Secure if $F_k(\mathrm{IV}) = G_\infty(k, IV, 1^n)$ is a (weak) PRF.
  - No shared state, but longer ciphertexts….

Pseudorandom Permutation

A keyed function $\mathrm{F}: \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$, which is invertible and "looks random" without the secret key k.

- Similar to a PRF, but
- Computing $F_k(x)$ *and* $F_k^{-1}(x)$ is efficient (polynomial-time)

Definition 3.28: A keyed function $\mathrm{F}: \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ is a **strong pseudorandom permutation** if for all PPT distinguishers D there is a negligible function μ s.t.

$$\left| Pr\left[D^{F_k(.),F_k^{-1}(.)}(1^n)\right] - Pr\left[D^{f(.),f^{-1}(.)}(1^n)\right]\right| \leq \mu(n)$$

Pseudorandom Permutation

Definition 3.28: A keyed function $\mathrm{F}: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a **strong pseudorandom permutation** if for all PPT distinguishers D there is a negligible function μ s.t.

$$\left|Pr\left[D^{\boxed{F_k(.),F_k^{-1}(.)}}(1^n)\right] - Pr\left[D^{\boxed{f(.),f^{-1}(.)}}(1^n)\right]\right| \le \mu(n)$$

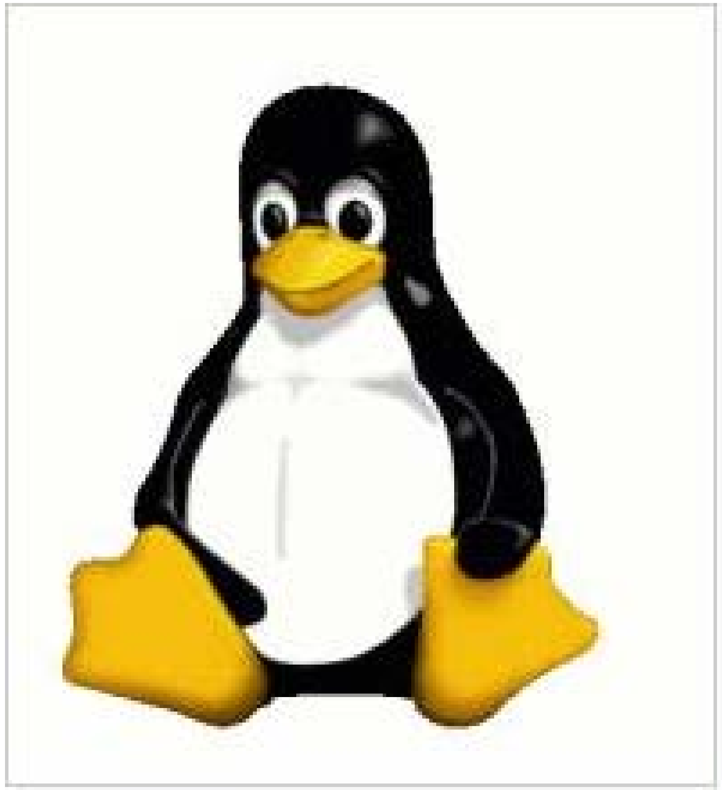
Notes:

- the first probability is taken over the uniform choice of $k \in \{0,1\}^n$ as well as the randomness of D.
- the second probability is taken over uniform choice of f $\in$**$\mathbf{Perm}_n$**as well as the randomness of D.
- D is *never* given the secret k
- However, D is given oracle access to keyed permutation and inverse

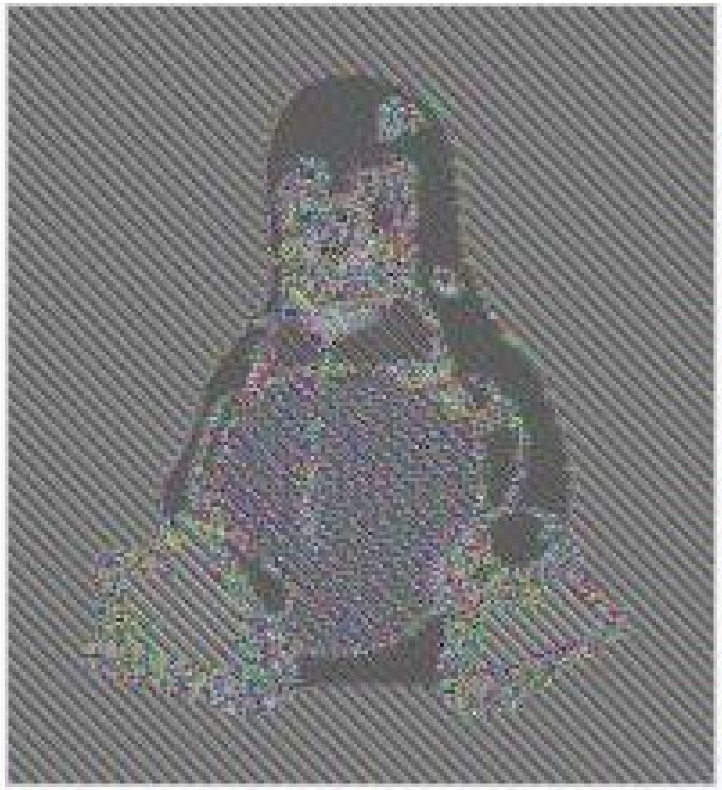
Electronic Code Book (ECB) Mode

- Uses strong PRP $F_k(x)$ and $F_k^{-1}(x)$
- Enc_k
  - **Input**: $m_1,\ldots,m_\ell$
  - **Output**: $\langle F_k(m_1), \ldots, F_k(m_\ell) \rangle$
- How to decrypt?
- Is this secure?
- **Hint**: Encryption is deterministic.
  - **Implication**: Not CPA-Secure
  - But, it gets even worse

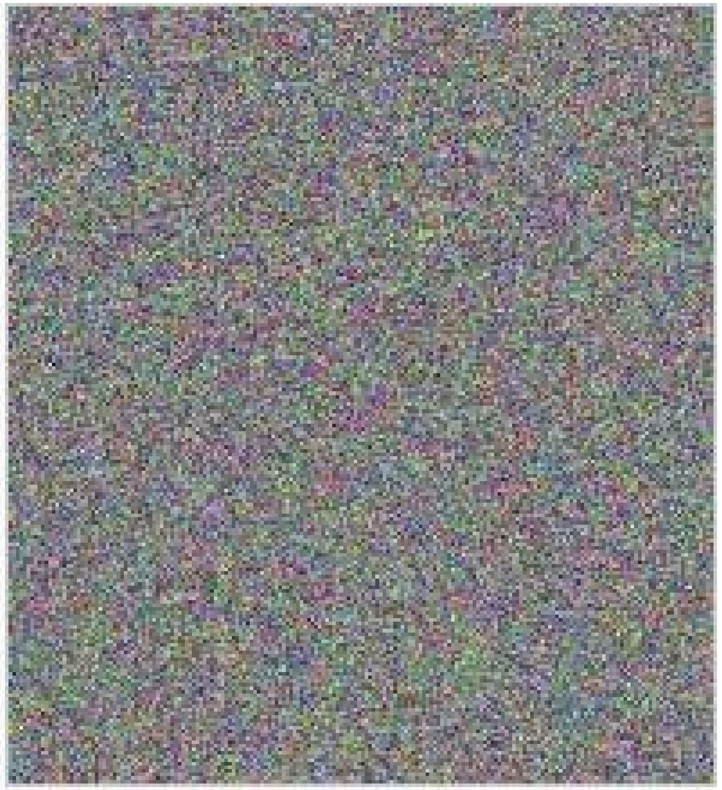
ECB Mode (A Failed Approach)



Original image



Encrypted using ECB mode



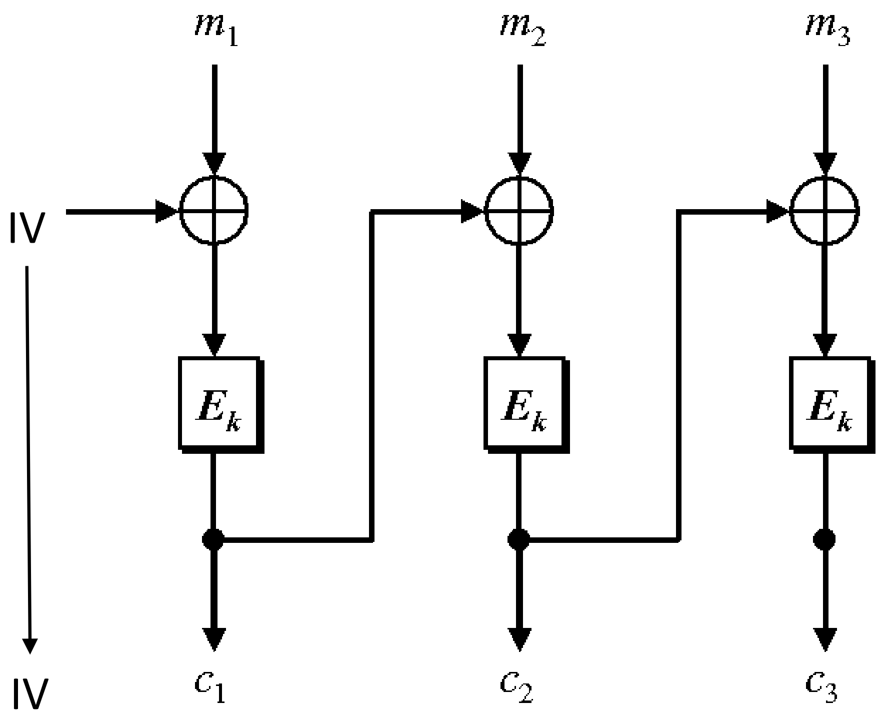
Modes other than ECB result in pseudo-randomness

The Penguin Principle

If you can still see the penguin after "encrypting" the image something is very very wrong with the encryption scheme.

Cipher Block Chaining

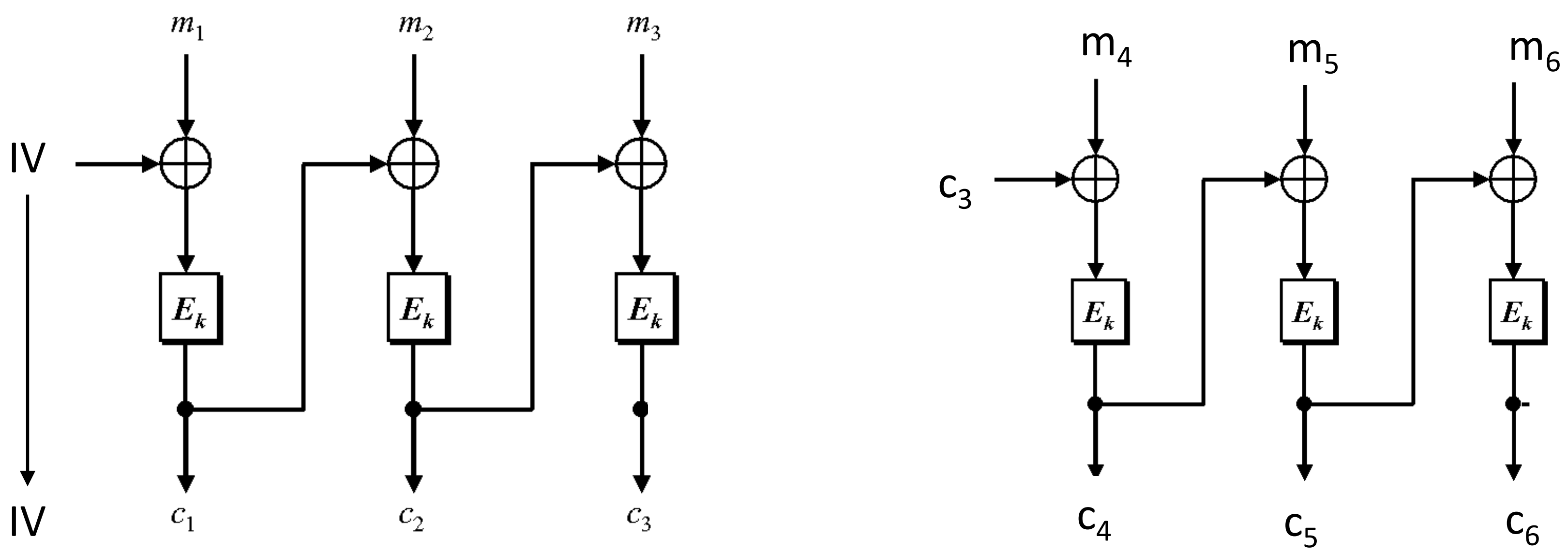
- CBC-Mode (below) is CPA-secure if E_k is a PRP



Reduces bandwidth!

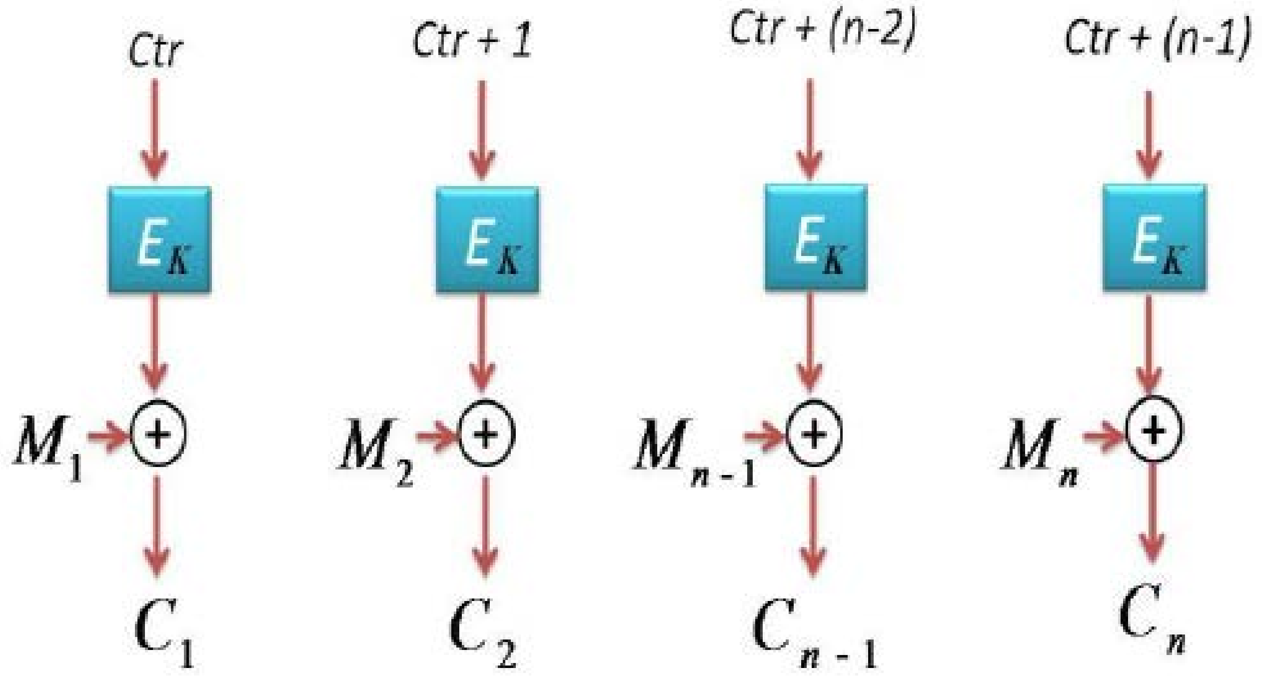
Message: 3n bits
Ciphertext: 4n bits

Chained CBC-Mode



- First glance: seems similar to CBC-Mode and reduces bandwidth
- Vulnerable to CPA-Attack! (Set $\mathrm{m}_4 = \mathrm{IV} \oplus c_3 \oplus m_1'$ and c_4=c_1 iff m_1=m_1')
- **Moral**: Be careful when tweaking encryption scheme!

Counter Mode



- Input: $m_1,\ldots,m_n$
- Output: $c = (ctr, c_1,c_2,\ldots,c_n)$ where ctr is chosen uniformly at random
- **Theorem**: If E_k is PRF then counter mode is CPA-Secure
- **Advantages**: Parallelizable encryption/decryption

Next Class

- Read Katz and Lindell 4.1-4.2
- Message Authentication Codes (MACs) Part 1