CS458: Introduction to Information Security

Notes 4: Symmetric Cryptography - More About Block Ciphers

Yousef M. Elmehdwi

Department of Computer Science

Illinois Institute of Technology

yelmehdwi@iit.edu

September 20, 2018

Slides: Modified from Christof Paar and Jan Pelzl

Outline

- Encryption with Block Ciphers: Modes of Operation
- Exhaustive Key Search Revisited
- Increasing the Security of Block Ciphers

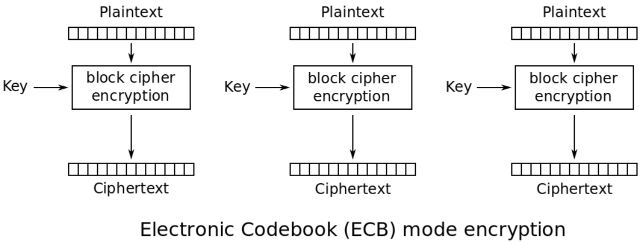
modes of operation

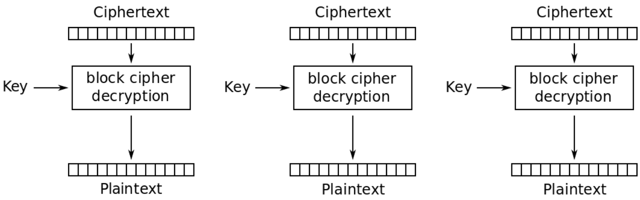
- There are several ways of encrypting long plaintexts, e.g., an e-mail or a computer file, with a block cipher (“modes of operation”)
- Many modes - we discuss 3 most popular.
- Electronic Codebook (ECB) mode
  - Encrypt each block independently.
  - Most obvious approach, but a bad idea.
- Cipher Block Chaining (CBC) mode
  - Chain the blocks together.
  - More secure than ECB, virtually no extra work.
- Counter (CTR) mode
  - Block ciphers acts like a stream cipher.
  - Popular for random access.

Electronic Codebook (ECB) mode

- Most obvious way to use a block cipher.
    - Encrypt all plaintext blocks.
    - Concatenate all resulting ciphertext blocks.
    - Output ciphertext

ECB Mode[1]





[1] Image Source Wikipedia:Block cipher mode of operation

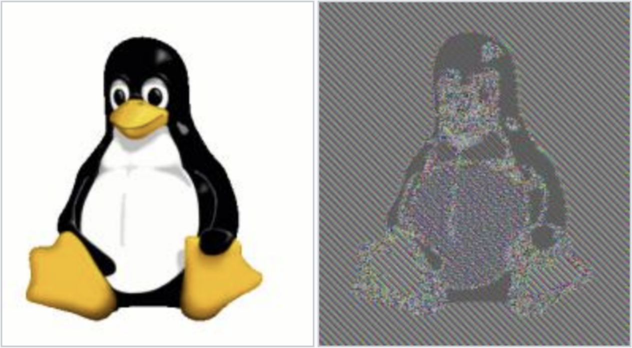
ECB Cut and Paste

- Suppose plaintext is:
  - `Alice digs Bob.  Trudy digs Tom.`
- Assuming 64-bit blocks and 8-bit ASCII:
  - x_0 = "Alice di", x_1 = "gs Bob."
  - x_2 = "Trudy di", x_3 = "gs Tom."
- Attack:
  - Ciphertext: y_0, y_1, y_2, y_3
  - Eve (here Trudy) cuts and pastes: y_0, y_3, y_2, y_1
  - Decrypts as
    `Alice digs Tom.  Trudy digs Bob.`

ECB Weakness

- Suppose $x_i = x_j$
  - Then $y_i = y_j$ and Eve knows $x_i = x_j$
  - This gives Eve some information, even if she does not know x_i or x_j
  - Eve might know x_i
- Q: Is this a serious issue?
  - The disadvantage of this method is a lack of diffusion.
  - Because ECB encrypts identical plaintext blocks into identical ciphertext blocks, it does not hide data patterns well.
  - In some senses, it doesn't provide serious message confidentiality, and it is not recommended for use in cryptographic protocols at all.

Alice hates ECB mode[2]



Original image Encrypted using ECB mode

- Q: Why does it happen?

[2] Image Source Wikipedia:Block cipher mode of operation

ECB: advantages/disadvantages

- Advantages
  - bit errors caused by noisy channels only affect the corresponding block but not succeeding blocks
  - Block cipher operating can be parallelized
    - advantage for high-speed implementations
- Disadvantages
  - ECB encrypts highly deterministically
    - identical plaintexts result in identical ciphertexts
    - an attacker recognizes if the same message has been sent twice
    - plaintext blocks are encrypted independently of previous blocks
    - an attacker may reorder ciphertext blocks which results in valid plaintext

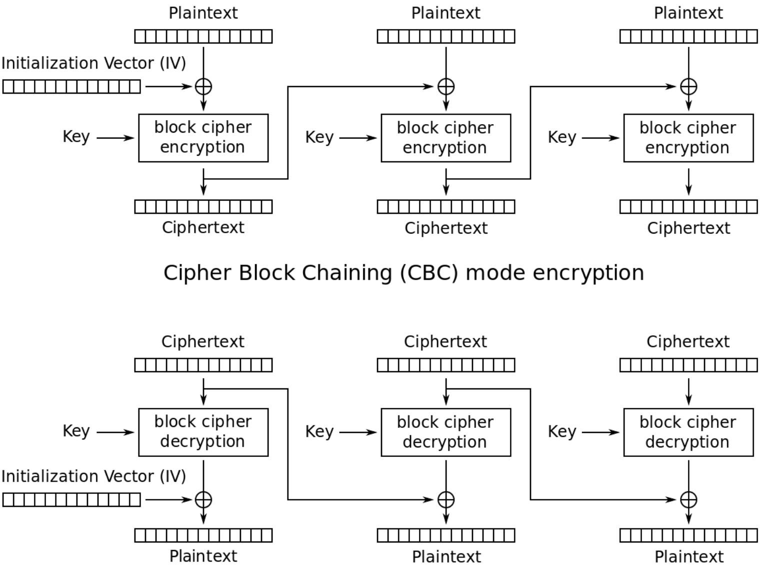
Cipher Block Chaining (CBC) mode

- We want to solve two problems
    1. Make encryption probabilistic
    2. Combine encryption of all blocks
- An encryption scheme is "deterministic" if a particular plaintext is mapped to a fixed ciphertext if the key is unchanged
- A "probabilistic" encryption scheme uses randomness to achieve a non-deterministic generation of y_i

Cipher Block Chaining (CBC) mode

- Blocks are "chained" together in a special way that introduces dependance between them.
- A random initialization vector, or IV, is required to initialize CBC mode.
  - Nothing to chain the first block with.
  - IV is random, but not secret

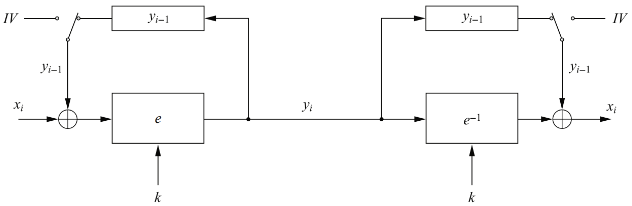
CBC Mode[3]



[3] Image Source Wikipedia:Block cipher mode of operation

CBC Mode: Encryption and decryption

- For the first plaintext block x_1 there is no previous ciphertext
  - an IV is added to the first plaintext to make each CBC encryption nondeterministic
  - the first ciphertext y_1 depends on plaintext x_1 and the IV
- The second ciphertext y_2 depends on the IV, x_1 and x_2
- The third ciphertext y_3 depends on the IV and x_1, x_2 and x_3, and so on

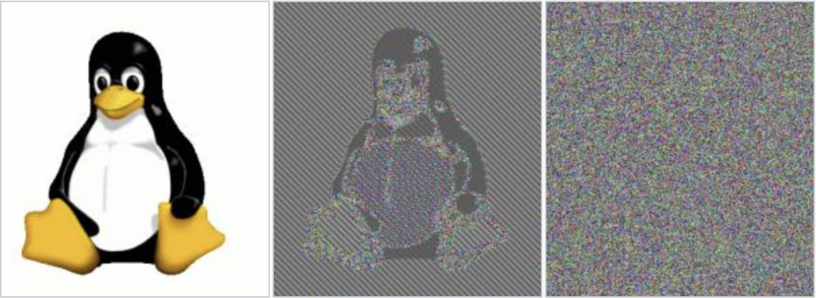


- **Encryption**
  - $y_1 = E_K(x_1 \oplus IV)$
  - $y_i = E_K(x_i \oplus y_{i-1}),\ i \geq 2$
- **Decryption**
  - $x_1 = D_K(y_1) \oplus IV$
  - $x_i = D_K(y_i) \oplus y_{i-1},\ i \geq 2$

CBC Mode

- Identical plaintext blocks yield different ciphertext blocks - this is very good!
- But what about errors in transmission?
  - If y_j is garbled to, say, G then
    - $x_j \neq D_K(G) \oplus y_{j-1}$
    - $x_{j+1} \neq D_K(y_{j+1}) \oplus G$
  - But
    - $x_{j+2} = D_K(y_{j+2}) \oplus y_{j+1}$
    - $x_{j+3} = D_K(y_{j+3}) \oplus y_{j+2}$
    - $\ldots$
  - Automatically recovers from errors!
  - One damaged block propagates to two blocks.
- Cut and paste is still possible, but more complex (and will cause garbles)

Alice likes CBC mode[4]



Original image | Encrypted using ECB mode | Modes other than ECB result in pseudo-randomness

- Q: Why does it happen?

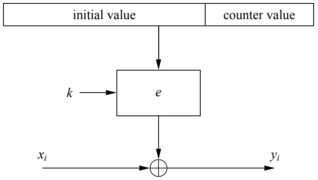
[4] Image Source Wikipedia:Block cipher mode of operation

IV: Initial Vector

- Does not have to be secret
- Should be a non-secret nonce (value used only once)

Counter Mode (CTR) mode

- Use block cipher like a stream cipher.
    - i.e., use the block cipher as a key stream generator
- The key stream is computed in a block wise fashion
- The input to the block cipher is a counter which assumes a different value every time the block cipher computes a new key stream block

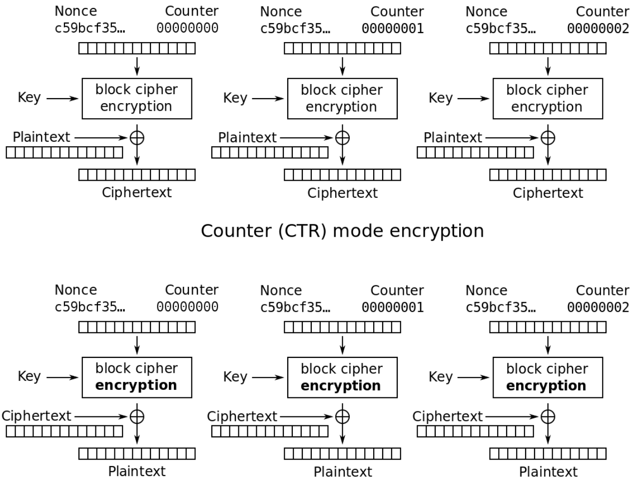


- CTR is popular for random access.
- Preprocessing can greatly improve efficiency.
- Never recovers from IV errors.
- Critical not to reuse IV.
- No error propagation in case of loss or damage.

Counter Mode (CTR) mode

- **Encryption**
  - $y_i = E_k(IV \; // \; CTR_i) \oplus x_i, \; i \geq 1$
- **Decryption**
  - $x_i = E_k(IV \; // \; CTR_i) \oplus y_i, \; i \geq 1$

CTR Mode[5]



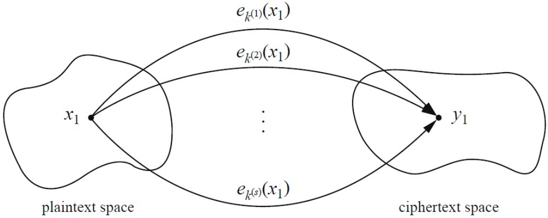
5 Image Source Wikipedia:Block cipher mode of operation

Outline

- Encryption with Block Ciphers: Modes of Operation
- Exhaustive Key Search Revisited
- Increasing the Security of Block Ciphers

Exhaustive Key Search Revisited

- A simple exhaustive search for a DES key knowing one pair (x_1, y_1):
- $DES_{k^{(i)}}(x_1) \stackrel{?}{=} y_1$, $i = 0,1,\ldots,2^{56}-1$
- However, for most other block ciphers a key search is somewhat more complicated
- A brute-force attack can produce false positive results
  - keys k^i that are found are not the one used for the encryption



  - The likelihood of this is related to the relative size of the key space and the plaintext space
  - A brute-force attack is still possible, but several pairs of plaintext–ciphertext are needed

Exhaustive Key Search Revisited

- Assume a cipher with a block width of $64\ bit$ and a key size of $80\ bit$
- If we encrypt x_1 under all possible $2^{80}\ keys$, we obtain 2^{80} ciphertexts
  - However, there exist only 2^{64} different ones
- If we run through all keys for a given plaintext-ciphertext pair, we find on average $2^{80}/2^{64} = 2^{16}$ keys that perform the mapping $e_k(x_1)=y_1$
  - Given a block cipher with a key length of $k\ bits$ and block size of $n\ bits$, as well as t plaintext–ciphertext pairs $(x_1,\ y_1),\ldots,(x_t,y_t)$, the expected number of false keys which encrypt all plaintexts to the corresponding ciphertexts is: 2^{k-tn}
- In this example assuming two plaintext-ciphertext pairs, the likelihood is $2^{80-2\times 64}=2^{-48}$
  - for almost all practical purposes two plaintext-ciphertext pairs are sufficient

Outline

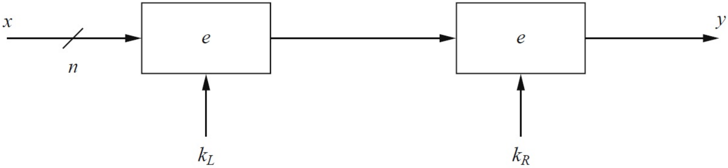
- Encryption with Block Ciphers: Modes of Operation
- Exhaustive Key Search Revisited
- Increasing the Security of Block Ciphers
  - Double Encryption and Meet-in-the-Middle Attack
  - Triple Encryption
  - Key Whitening

Increasing the Security of Block Ciphers

- In some situations we wish to increase the security of block ciphers
- Two approaches are possible
    - Multiple encryption
        - theoretically much more secure, but **sometimes** in practice increases the security very little
    - Key whitening

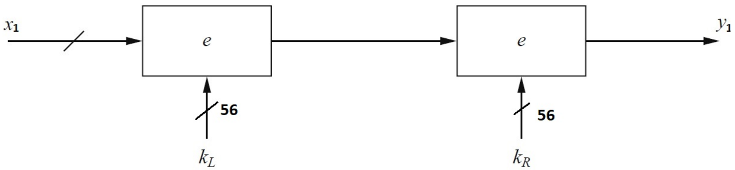
Double Encryption

- A plaintext x is first encrypted with a key k_L, and the resulting ciphertext is encrypted again using a second key k_R



- Assuming a key length of $k\ bits$, an exhaustive key search would require $2^k \times 2^k = 2^{2k}$ encryptions or decryptions

Complexity of brute-force attack?



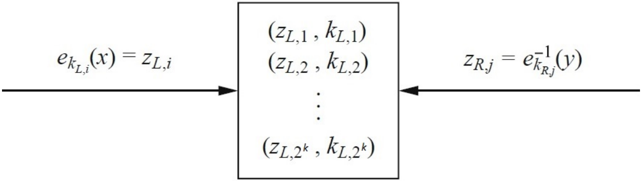
- **Naïve Approach**
  - $x_1 \stackrel{?}{=} e^{-1}{}_{k_{L,i}}(e^{-1}{}_{k_{R,j}}(y_1))$
  - $2^{56} \times 2^{56} = 2^{112}$ key tests $\Rightarrow$ lifetime of universe :)
- Can we find a better attack?

Meet-in-the-Middle Attack

- Can we search for K_L and K_R separately?
- $\underbrace{2^{56}}_{search\ for\ K_L} + \underbrace{2^{56}}_{search\ for\ K_R} = 2 \times 2^{56} = 2^{57}$
- Meet-in-the-Middle Attack

Meet-in-the-Middle Attack

- A Meet-in-the-Middle attack requires $2^k + 2^k = 2^{k+1}$ operations!



- **Phase I: Table Computation**: for the given (x_1, y_1) the left encryption is brute-forced for all $k_{L,i}$, $i = 1, 2, \ldots, 2^k$ and a lookup table with 2^k entry (each $n+k$ bits wide) is computed
  - the lookup table should be ordered by the result of the encryption $(z_{L,i})$
- **Phase II: Key Matching**: the right encryption is brute-forced (using decryption) and for each $z_{R,i}$ it is checked whether $z_{R,i}$ is equal to any $z_{L,i}$ value in the table of the first phase
  - $\Rightarrow (K_{L,i}, K_{R,j})$ are possible keys (K_L, K_R)
  - Note: sometimes we have to use a second pair (x_2, y_2):

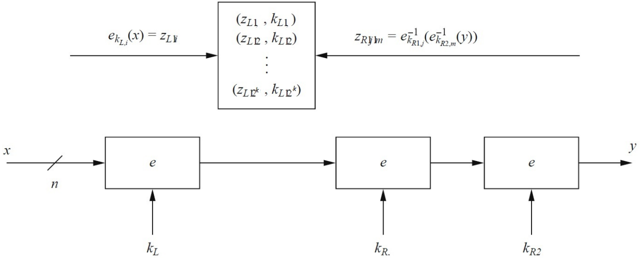
$$x_2 \stackrel{?}{=} e^{-1}{}_{k_{L,i}}(e^{-1}{}_{k_{R,j}}(y_2))$$

Meet-in-the-Middle Attack

- Computational Complexity
  - number of encryptions and decryptions $= 2^k + 2^k = 2^{k+1}$
  - number of storage locations $= 2^k$
- **Double encryption is not much more secure then single encryption!**

Triple Encryption

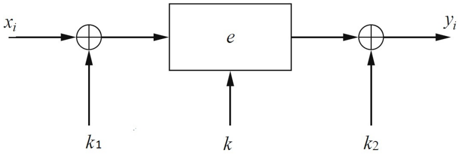
- The encryption of a block three times $y = e_{k_3}(e_{k_2}(e_{k_1}(x)))$
- In practice a variant scheme is often used EDE (encryption-decryption-encryption) $y = e_{k_3}(e^{-1}_{k_2}(e_{k_1}(x)))$
  - Advantage: choosing $k_1 = k_2 = k_3$ performs single DES encryption
- Still we can perform a meet-in-the middle attack, and it reduces the effective key length of triple encryption from 3K to 2K
  - The attacker must run 2^{112} tests in the case of 3DES



- **Triple encryption effectively doubles the key length**

Key Whitening

- Makes block ciphers such as DES much more resistant against brute-force attacks
- In addition to the regular cipher key k, two whitening keys k_1 and k_2 are used to XOR-mask the plaintext and ciphertext



- It does not strengthen block ciphers against most analytical attacks such as linear and differential cryptanalysis
- It is not a "cure" for inherently weak ciphers
- Its main application is ciphers that are relatively strong against analytical attacks but possess too short a key space especially DES
- Most modern block ciphers such as AES already apply key whitening internally by adding a subkey prior to the first round and after the last round.

Lessons Learned

- There are many different ways to encrypt with a block cipher. Each mode of operation has some advantages and disadvantages
- Several modes turn a block cipher into a stream cipher
- There are modes that perform encryption together with authentication, i.e., a cryptographic checksum protects against message manipulation
- The straightforward ECB mode has security weaknesses, independent of the underlying block cipher
- The counter mode allows parallelization of encryption and is thus suited for high speed implementations
- Double encryption with a given block cipher only marginally improves the resistance against brute-force attacks
- Triple encryption with a given block cipher roughly doubles the key length
- Triple DES (3DES) has an effective key length of *112 bits*
- Key whitening enlarges the DES key length without much computational overhead.