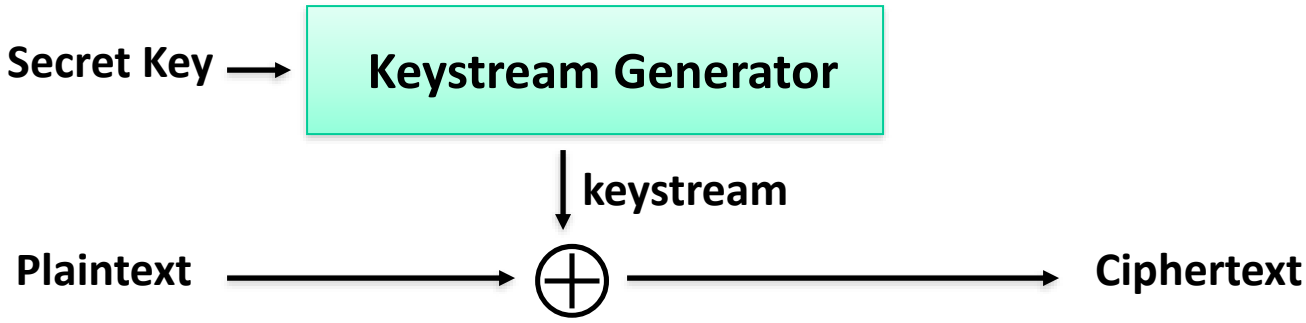
CS5285
Tutorial 3

Question 1

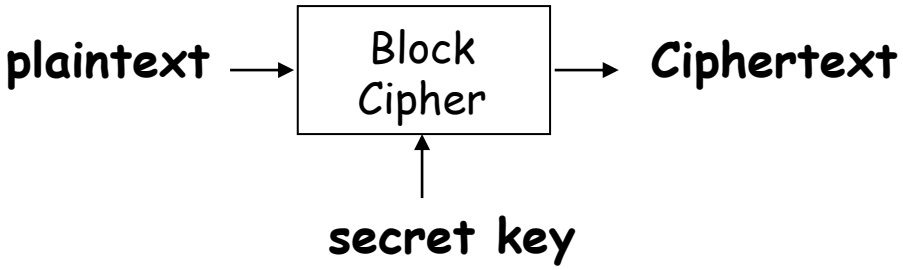
What are the advantage and disadvantage of stream cipher vs. block cipher?

Stream Ciphers



- Secret key length: 128 bits, 256 bits, etc.
- Maximum plaintext length: usually can be arbitrarily long.
- **Security:** Given a "long" segment of keystream (e.g. 2^{40} bits), the secret key cannot be derived AND the subsequent segment of the keystream cannot be deducted.

Block Ciphers



- A block cipher takes a *block* of **plaintext** and a **secret key**, produces a *block* of **ciphertext**.
- The key is **reused** for different plaintext blocks
- Typical block sizes: 64 bits, 128 bits, 192 bits, 256 bits
- Key sizes: 56 bits (DES), 128/192/256 bits (AES)
- Popular block ciphers: DES, 3DES, AES, Twofish, Serpent

Question 1

Advantage and disadvantage of stream cipher vs. block cipher?

- Stream Cipher
  - Advantage :
    - Said to be faster than block cipher (generate pseudo-random string, XOR).
    - Keystream function does not need to be reversible…
  - Disadvantage:
    - Keystream cannot be reused, same plaintext/keystream always yields same ciphertext (independent of previous plaintext).
    - Additional integrity check required, otherwise simple to modify bits in message.
- Block Cipher
  - Advantage:
    - If we use the right mode of operation ciphertext depends on prior plaintext even if key remains the same (e.g. in CBC mode just change first block of plaintext)
  - Disadvantage:
    - Needs to be reversible (PT> CT, CT>PT)
    - Needs padding to block size

Question 2

3DES: Consider 3DES:

$$C = \mathrm{DES}_{K_1}(\mathrm{DES}^{-1}_{K_2}(\mathrm{DES}_{K_1}(M)))$$

where C, M are the ciphertext and plaintext, respectively, and $K = (K_1, K_2)$ is the key. How many keys on average do we have to try in a bruteforce attack?

Bruteforce Attack | Exhaustive Key Search

- **An algorithm is secure when the easiest way of attacking it is by bruteforce attack.**
  - i.e. check all possible key combinations one by one (could be done in parallel)
  - For a key of *n* bits, the total number of possible keys (or the entire key space) is 2^n.
  - An average of half the combinations should be tried in order to find the key, i.e. 2^{n-1}.

Question 2

3DES: Consider 3DES:

$$C = \mathrm{DES}_{K_1}(\mathrm{DES}^{-1}_{K_2}(\mathrm{DES}_{K_1}(M)))$$

where C, M are the ciphertext and plaintext, respectively, and $K = (K_1, K_2)$ is the key. How many keys on average do we have to try in a bruteforce attack?

What is the key space?

Key length 56+56= 112 bits, key space is 2^{112}

How much brute force attempts?

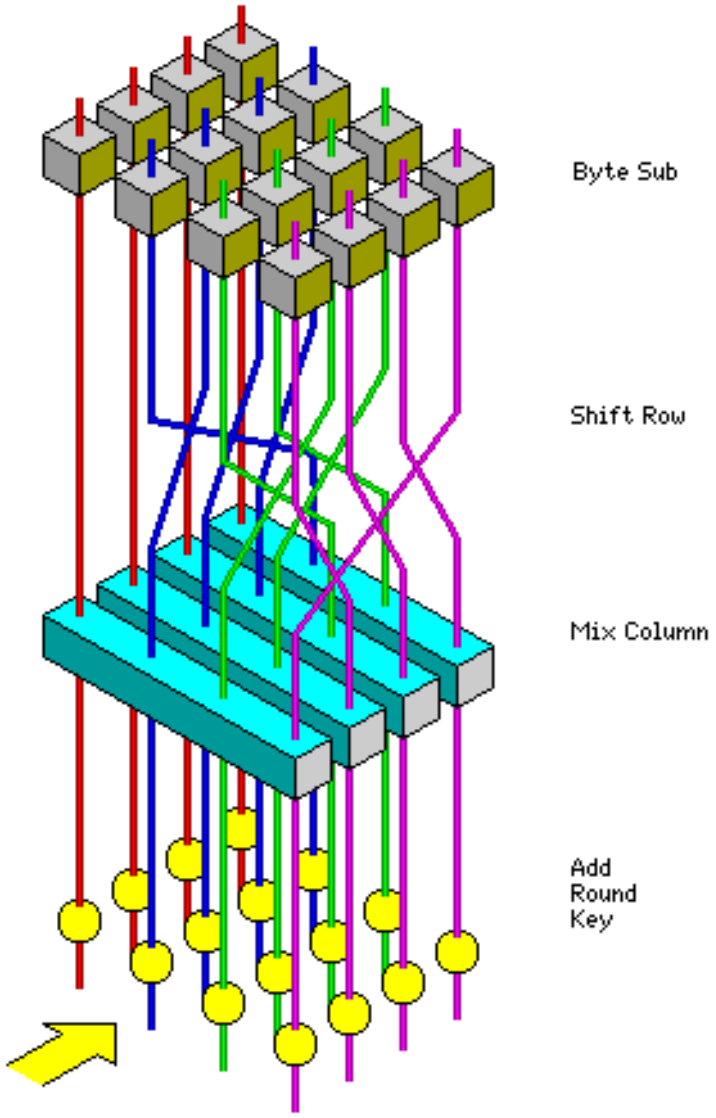
Keyspace/2 = $2^{(112-1)}=2^{111}$

Question 3

DESX: Let AESX-192 be a block cipher which is similar to DESX ($DESX(M) = K_3 \oplus DES_{K_2}(M \oplus K_1)$) but the DES ha been replaced by AES and the AES key size is 192 bits. Compute the keyspace of the AESX-192.

AES (Advanced Encryption Standard)

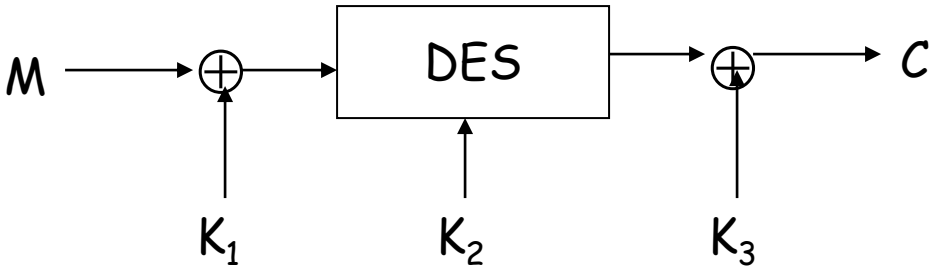
- **Replacement of DES**
- **Block size:** 128 bits
- **Key length:** 16, 24, or 32 bytes (128, 192, or 256 bits) – independent of block size
- 10 to 14 rounds (depends on key length)
- Each round has 4 transformations (except the last round)
  - **ByteSub**
  - **ShiftRow**
  - **MixColumn**
  - **AddRoundKey**



DESX

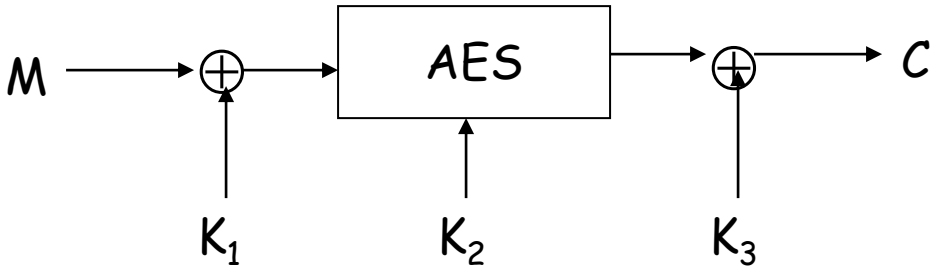
- DESX: three keys

$$C = K_3 \oplus DES(K_2, M \oplus K_1)$$



- If we made an AESX

$$C = K_3 \oplus AES(K_2, M \oplus K_1)$$



Question 3

DESX: Let AESX-192 be a block cipher which is similar to DESX ($DESX(M) = K_3 \oplus DES_{K_2}(M \oplus K_1)$) but the DES ha been replaced by AES and the AES key size is 192 bits. Compute the keyspace of the AESX-192.

- What is total keyspace? First how many keys?
- Three keys K1, K2 + K3, with key space being |K1|+|K2|+|K3|
- Size of K2?
  - K2= AES keys size
  - 192 bits
- Size of K1 and K3?
  - |K1|=|K3|=|M|=?
  - 128 bits
- Total keyspace = 2^{448} total (key length 448)

Question 4

Comment on security/efficiency of 2-key 3AES and AES-256

- Keyspace? Brute force search?
  - AES-256 has 256 bit key
  - 3AES has 128+128= 256 bit key
  - Same keyspace! Same complexity for key search.
- Efficiency? How long to compute?
  - AES-256 has 14 rounds
  - How many for 3AES?
    - 3x10=30
  - Which is one has shorter (time) brute force key search?
  - AES-256, less work to search as each attempt shorter

Question 5

CBC Mode: Consider a block cipher with CBC mode.

CBC Encryption

$C_0 = E(K; IV \oplus P_0)$

$C_1 = E(K; C_0 \oplus P1)$

$C_2 = E(K; C_1 \oplus P2)$

...

CBC Decryption

$P_0 = IV \oplus D(K; C_0)$

$P_1 = C_0 \oplus D(K; C_1)$

$P_2 = C_1 \oplus D(K; C_2)$

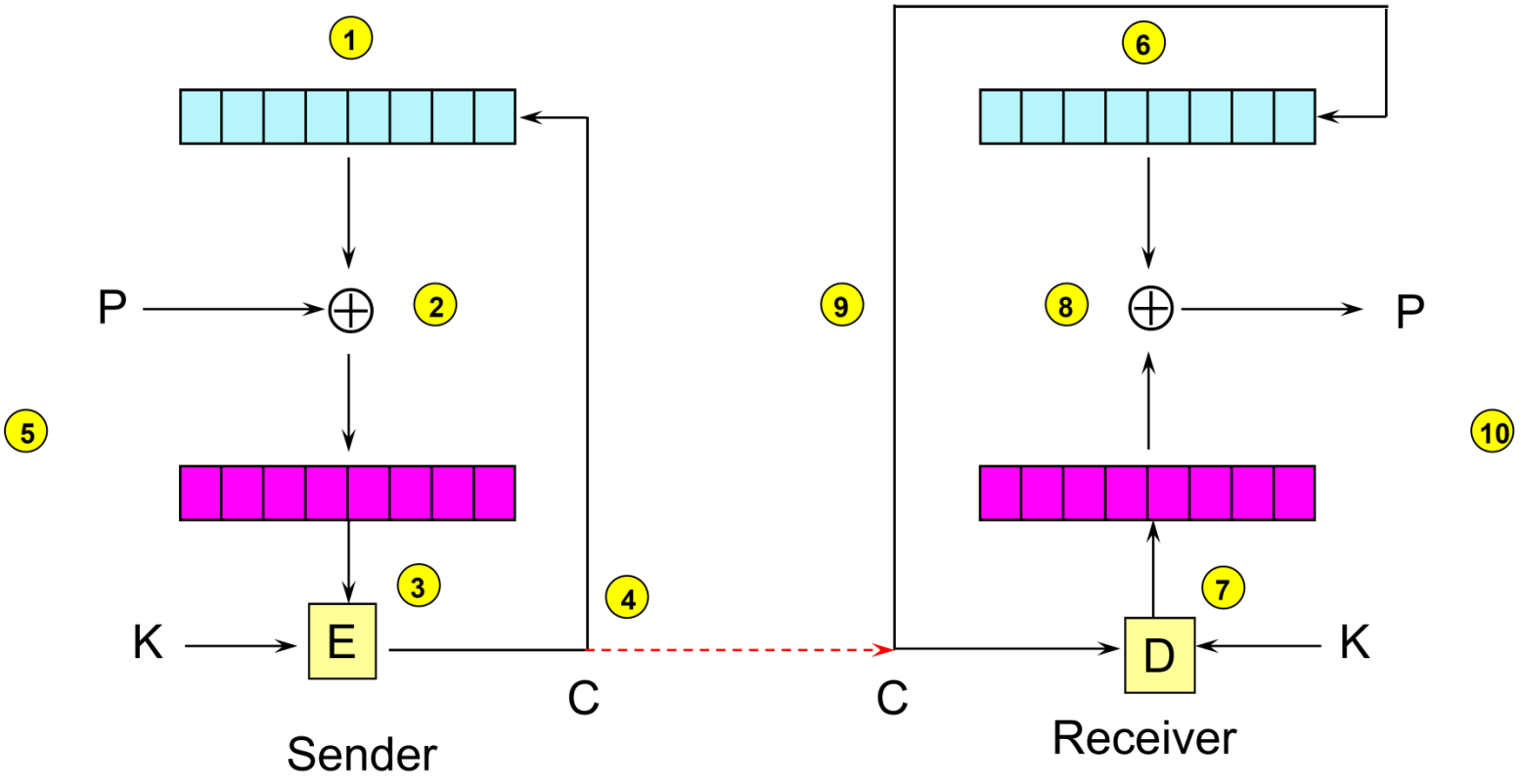
...

(a) During *encryption*, if one block of the *plaintext* input is different (two indentical messages, except for 1 bit), how many blocks of the corresponding ciphertext will be effected?

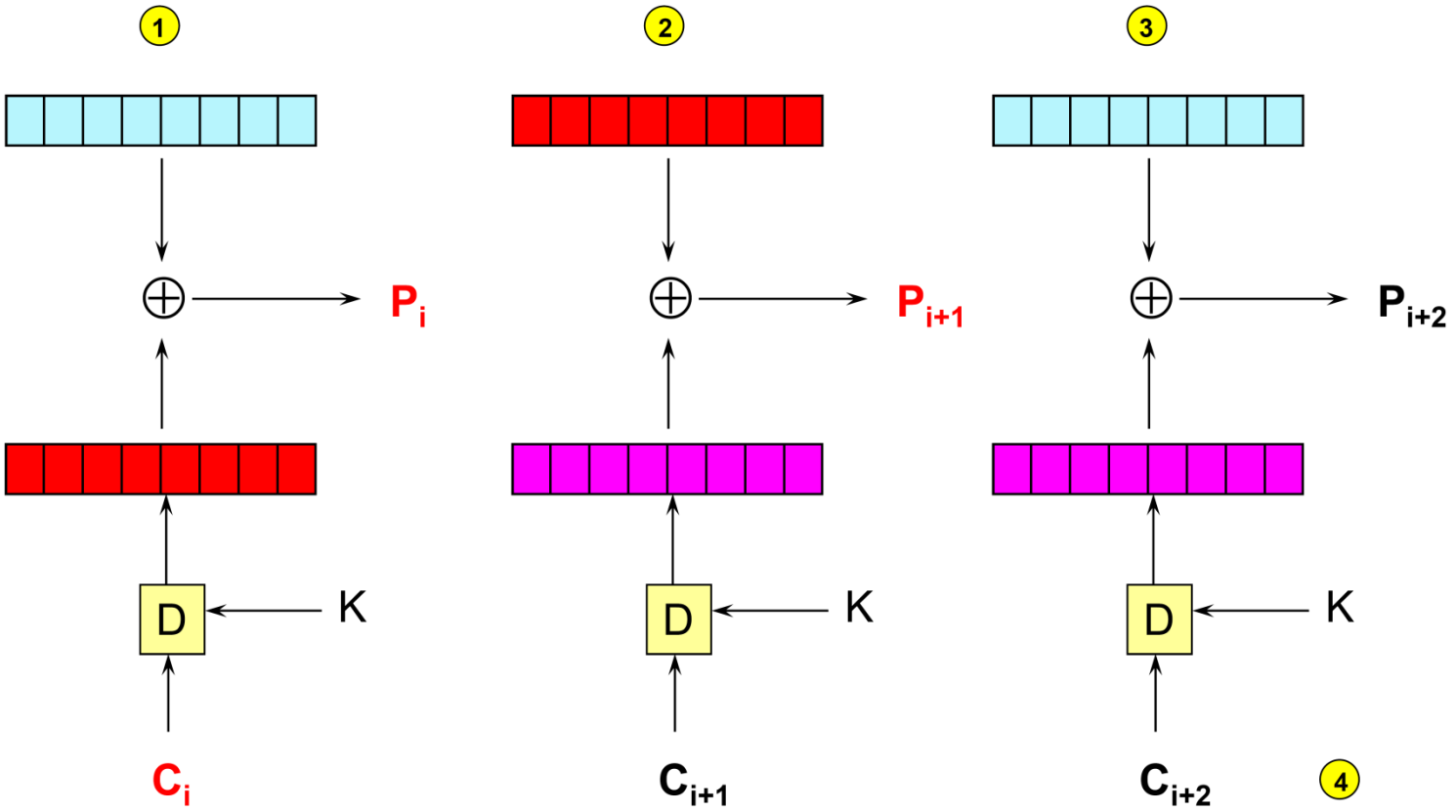
(b) During *decryption*, if one block of the *ciphertext* input is incorrect (an error occurs during transmission), how many blocks of the corresponding plaintext will be effected?

(c) What happens if the receiver has an incorrect IV when decrypting blocks of ciphertext in CBC mode?

CBC Mode



CBC Error



Question 5

(a) During *encryption*, if one block of the *plaintext* input is different (two indentical messages, except for 1 bit), how many blocks of the corresponding ciphertext will be effected?

(a) If one block of the plaintext is modified/updated when encrypting under CBC mode, then the current ciphertext block C_i as well as all subsequent ciphertext blocks C_{i+1}, C_{i+2}, $\cdots$ will be effected, because each ciphertext block depends on the previous ciphertext block, and thus an error in one of the plaintext blocks propagates indefinitely.

(b) During *decryption*, if one block of the *ciphertext* input is incorrect, how many blocks of the corresponding plaintext will be effected?

(b) If one ciphertext block C_i is modified/updated, then the corresponding plaintext block P_i as well as the next plaintext block P_{i+1} will be effected. Subsequent plaintext blocks P_{i+2}, P_{i+3}, $\cdots$ will be unaffected. CBC decryption is in this sense self-synchronising in that it recovers from a modified ciphertext block, although two blocks of plaintext will be modified.

(c) What happens if the receiver has an incorrect IV when decrypting blocks of ciphertext in CBC mode?

(c) If the IV is incorrect when decrypting blocks of ciphertext, then the first plaintext block P_0 will be updated. Subsequent plaintext blocks P_1, P_2, $\cdots$ will be correct, because they do not depend on the IV or previous plaintext blocks.

The end!

Any questions…