Cryptography III: Symmetric Ciphers Cont'

CSE 565: Fall 2024
Computer Security

Xiangyu Guo (xiangyug@buffalo.edu)

University at Buffalo

Announcement

- Please sign-up at course Piazza.
- Reminder of Quiz 0 (**Due 09/19**).

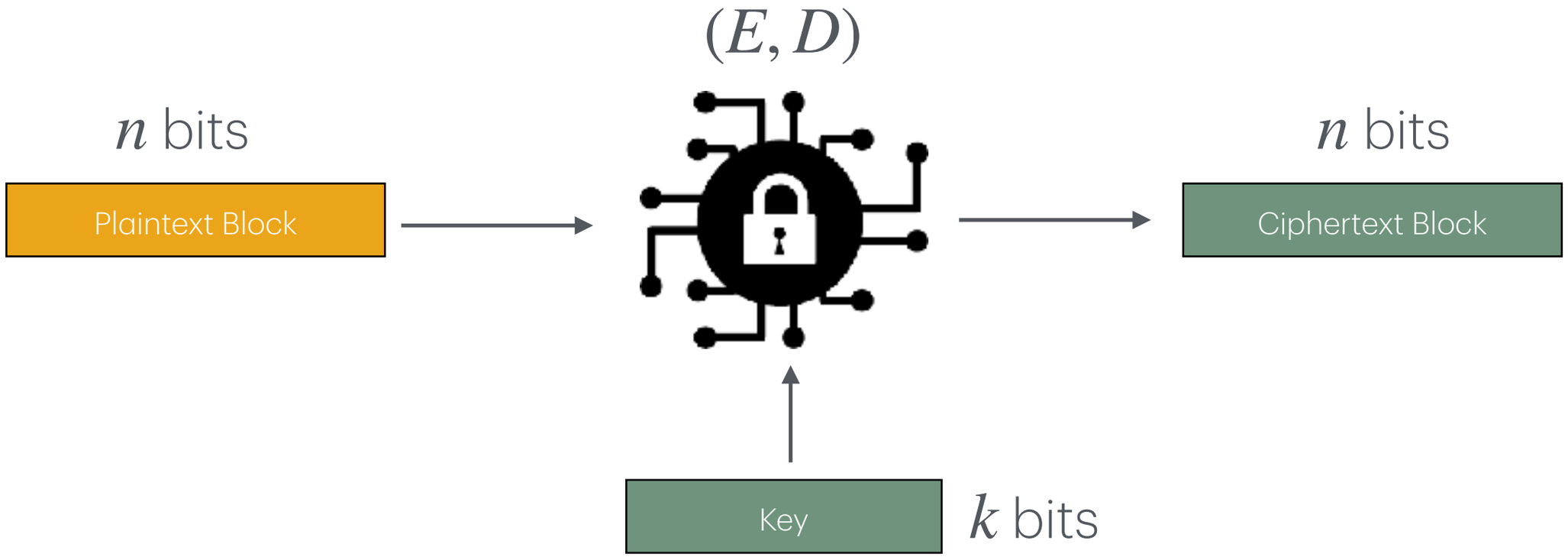
Review of Last Lecture

- Stream Ciphers
  - OTP, PRG
  - Stream cipher ≈ PRG + OTP
- Block Ciphers
  - Design principles
  - Common structure: SPN & Feistel Network

Today's Topic

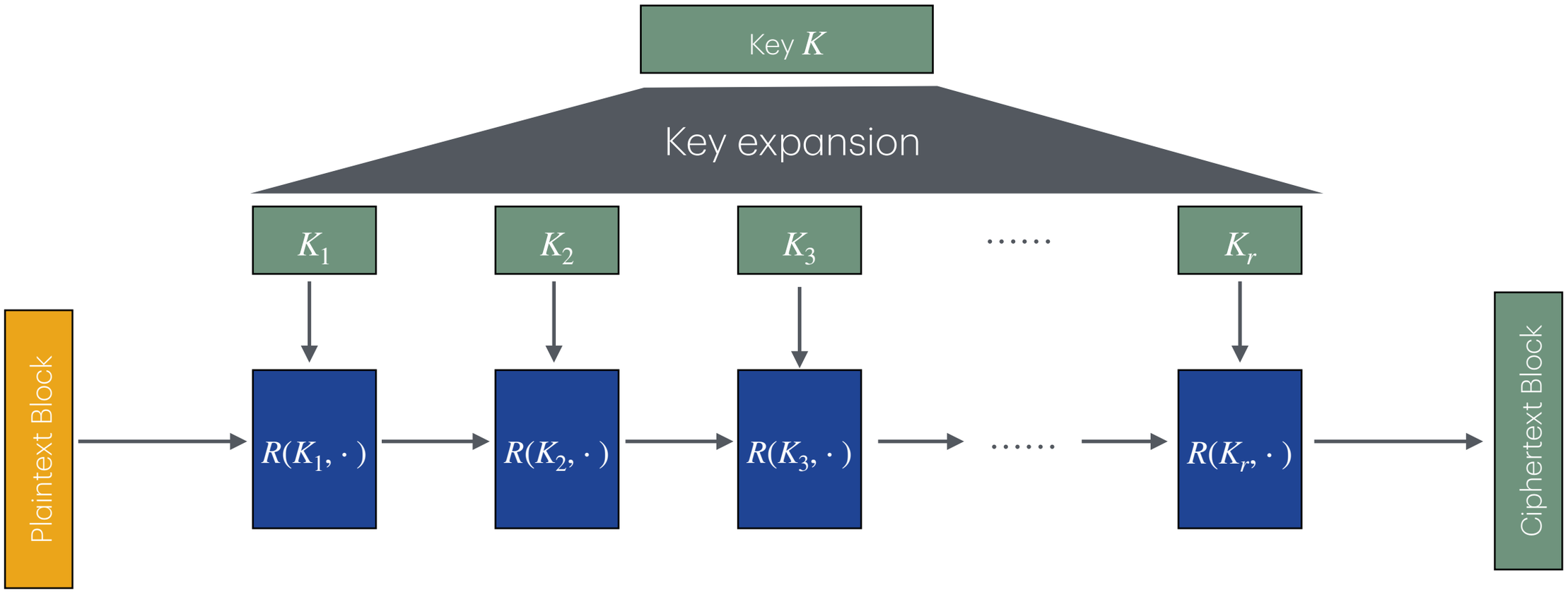
- Digital Encryption Standard (DES)
  - Construction; Attack; Strengthening
- Advanced Encryption Standard (AES)
  - Construction
- Block Cipher Encryption Modes
  - ECB: never use
  - CBC mode: random IV / nounce
  - CTR mode: essentially used as a PRG for stream cipher

Recall: What is a block cipher?



- Canonical examples:
  - ~~DES: $n = 64$ bits, $k = 56$ bits~~
  - 3DES: $n = 64$ bits, $k = 168$ bits
  - AES: $n = 128$ bits, $k = 128,192,256$ bits

Recall: What is a block cipher?



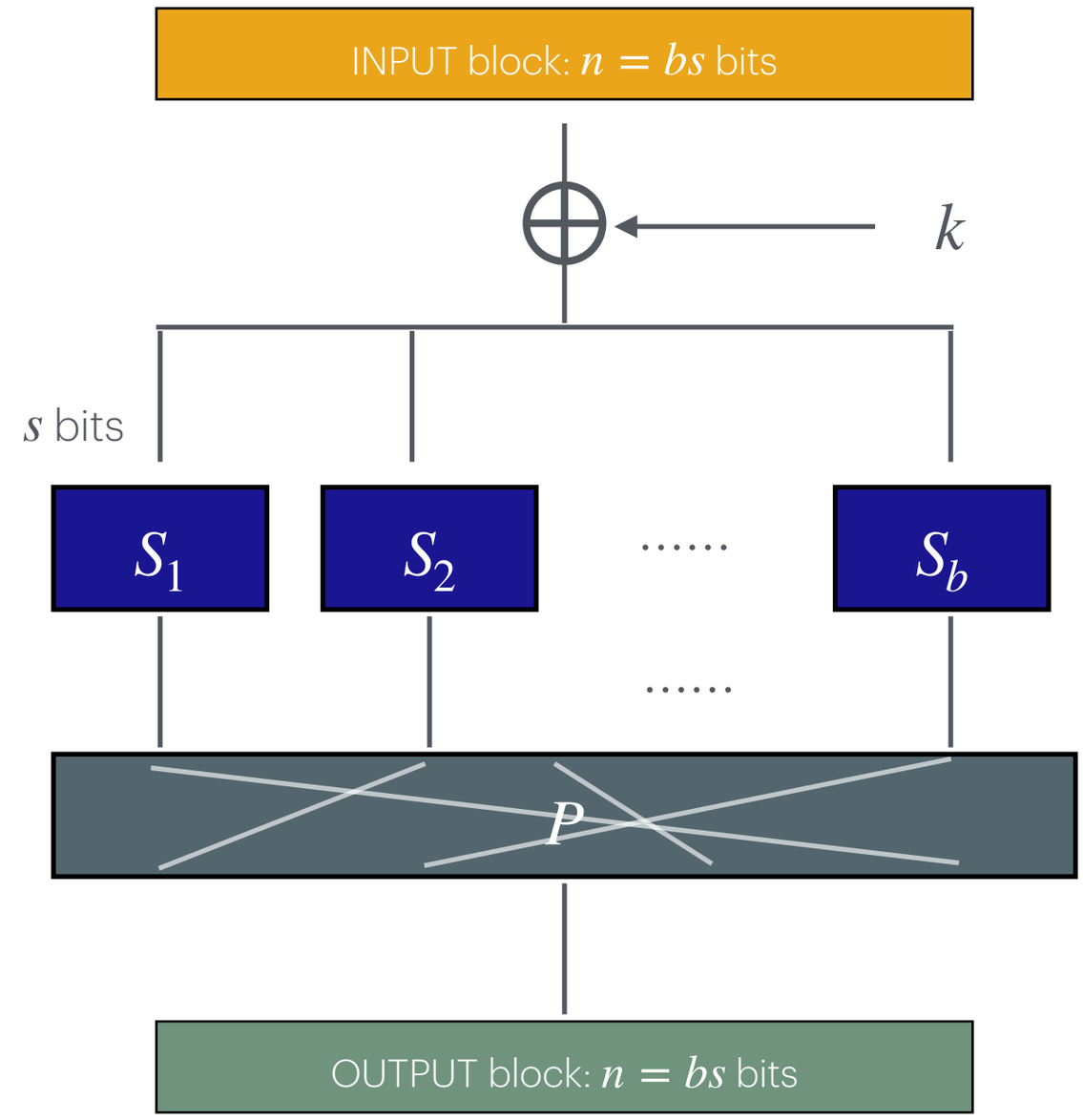
- $R(K_i, m)$: **round function**
  - DES: round $r = 16$, 3DES: round $r = 48$
  - AES: round $r = 10/12/14$

Recall: Substitution-Permutation Network (SPN)

One SPN *round*:

1. Split a block into b chunks
2. S-Box: substitute each block with another block
3. P-Box: Mix outputs from different chunks by permuting bits

- Every step is **reversible**.
- Decryption: run backwards.

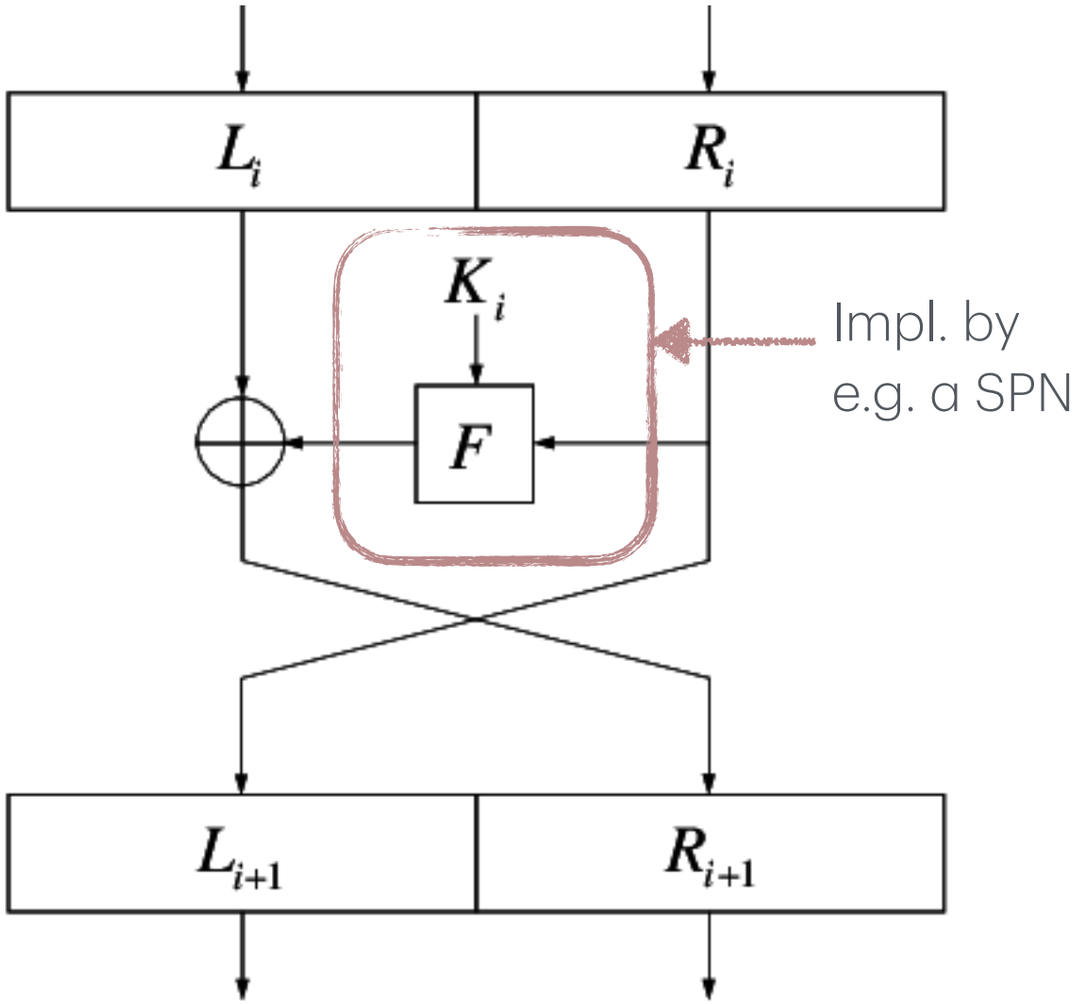


Recall: Feistel Network

One Feistel round:

- Only encrypt half of the Input block
  - So *one round alone* does not provide security
- Security provided by a *Pseudorandom Function* F
  - "Like" PRG used in stream cipher
- Lastly, swap the two half

Decrypt: run again with L, R swapped



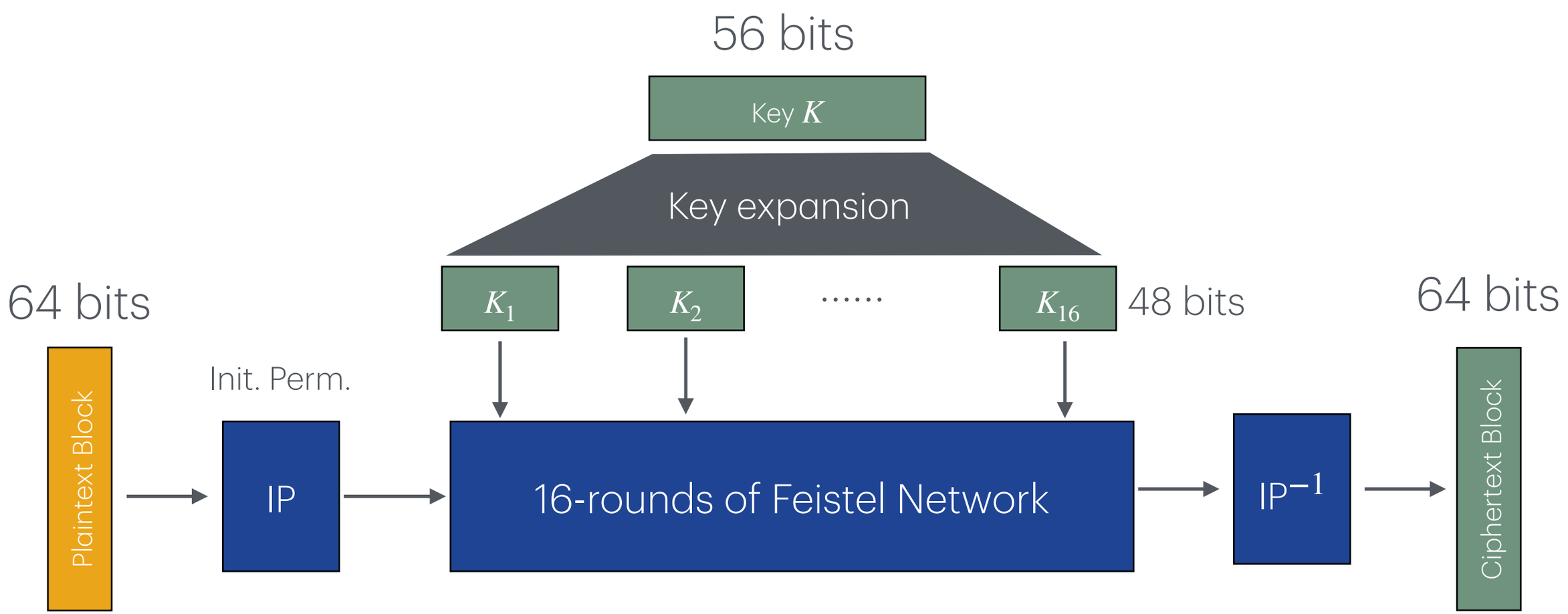
Data Encryption Standard (DES)

Data Encryption Standard (DES)

- Early 1970s: Horst Feistel designs Lucifer at IBM
  - key-len = **128** bits ; block-len = 128 bits
- 1973: NBS asks for block cipher proposals; IBM submits variant of Lucifer.
- 1976: NBS adopts DES as a federal standard; key-len = **56** bits ; block-len = 64 bits
- 1997: DES broken by exhaustive search
- 2000: NIST adopts Rijndael as AES to replace DES
- Widely deployed in banking (ACH) and commerce

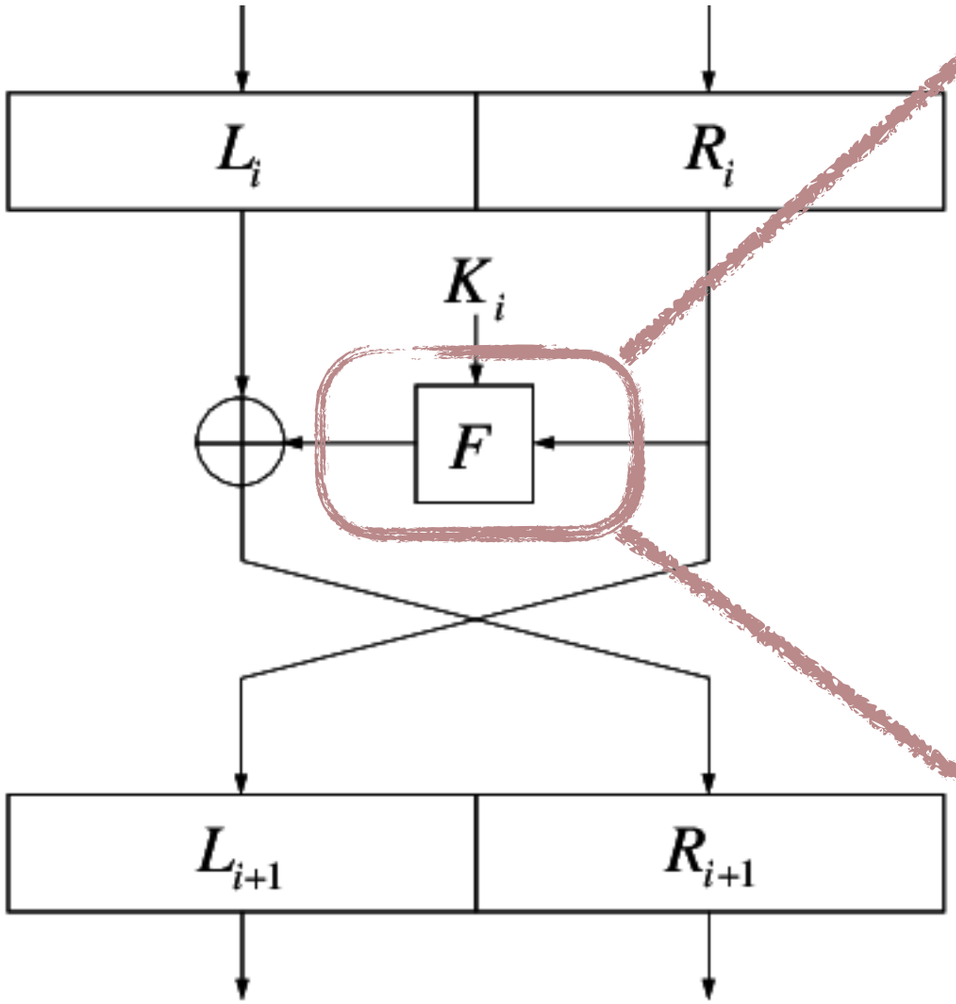
Data Encryption Standard (DES)

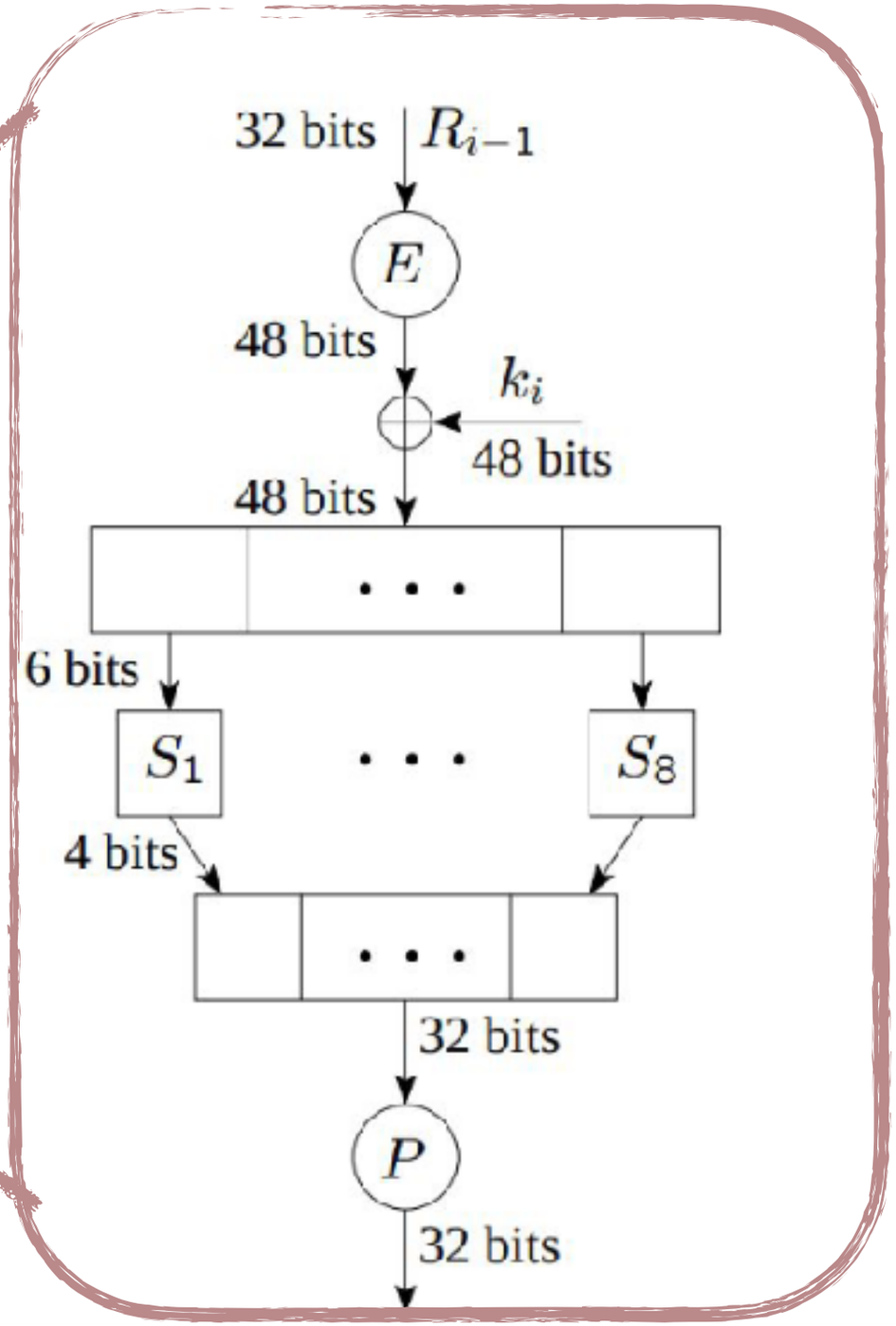
- Essentially just 16-rounds of Feistel Network



Data Encryption Standard (DES)

One round of Feistel:

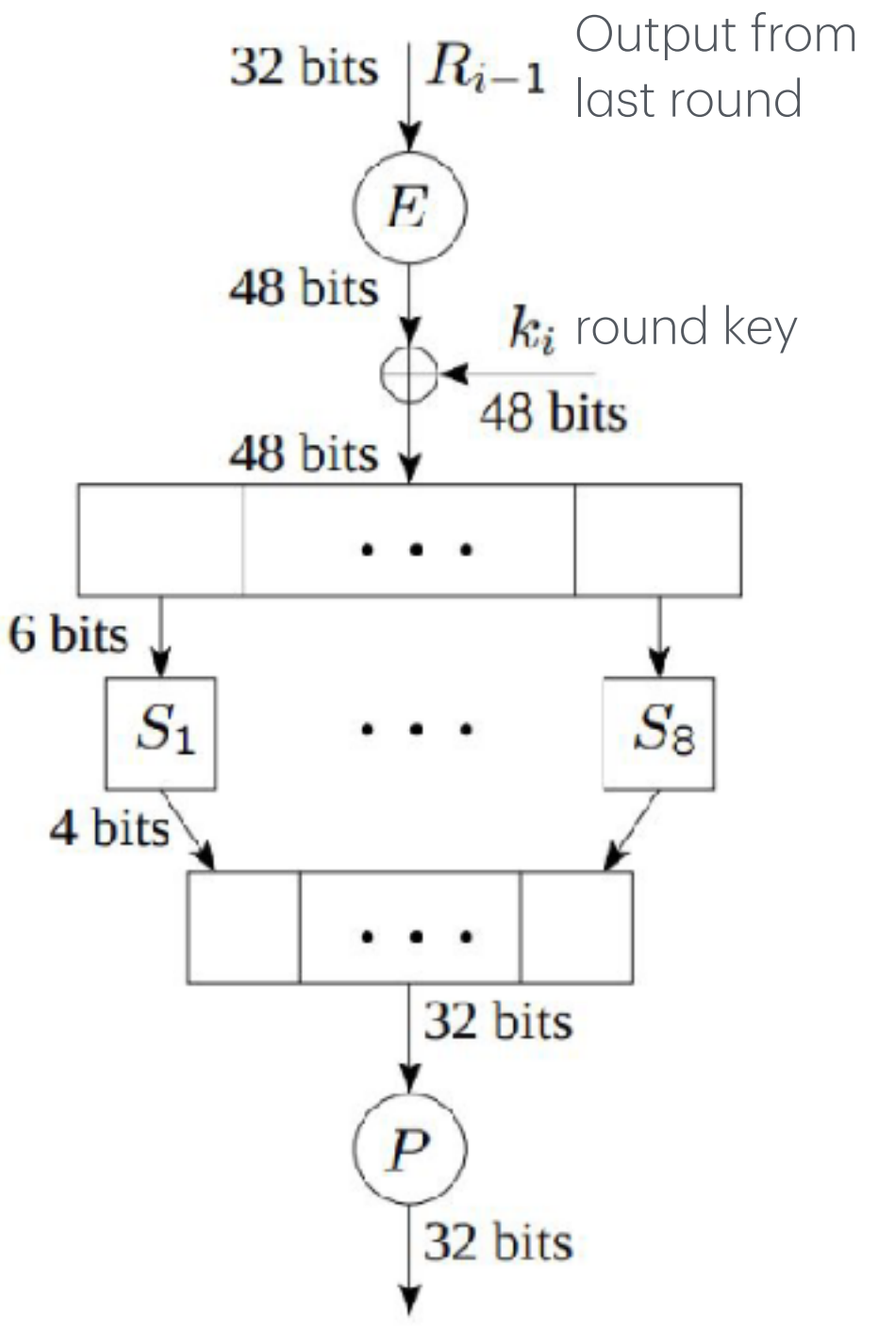




Data Encryption Standard (DES)

- The pseudorandom func $F(k_i, R_{i-1})$ is "essentially" a SPN

1. Input expansion: 32 →48 bits
2. XOR with round key
3. Substitution: S-boxes
   mapping 6 bits to 4 bits
   - Not reversible
4. Simple permutation



S-Box in DES

- Just a look-up table $S_i : \{0,1\}^6 \mapsto \{0,1\}^4$
  - E.g. **011011** ↦ **1001**

S_5		Middle 4 bits of input															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Outer bits	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

S-Box in DES

- S-Boxes are the only source of non-linearity in DES
  - If linear, then the entire DES is a linear function:

$$\mathrm{DES}(k, m) = \boxed{B} \begin{matrix} m \\ k_1 \\ \vdots \\ k_{16} \end{matrix} = \boxed{c} \mod 2$$

- Totally broken!
- If not *far from* linear, still susceptible to linear cryptanalysis [Matsui'1994]
  - (though not very practical, needs 2^{42} random input/output pairs)

S-Box in DES

- *Differential cryptanalysis* was rediscovered in the late 1980s [Biham-Shamir];
- The technique was known 20 years earlier to both IBM and the NSA and kept secret.
- To break the full 16 rounds, differential cryptanalysis requires 2^{47} chosen plaintexts.
- DES S-Box was designed to be resistant to such attacks.

Exhaustive search attack on DES

- Problem: Given some ptext/ctext pairs $(m_i, c_i), i = 1,\ldots, q$, find the corresponding DES $k \in \{0,1\}^{56}$
  - Fact: Only need $q = 3$ pairs to uniquely determine a key

- 1997: Internet search — 3 months
- 1998: EFF machine (“Deep Crack”) — 3 days (250K $)
- 1999: Combined search — 22 hours
- 2006: COPACOBANA (120 FPGAs) — 7 days (10K $)
- 2012: Online service with a fee - 26 hrs
- **56-bit ciphers should NOT be used**

Strengthening DES

- Example: Just apply DES multiple times
- **3DES**: let (E, D) be the DES cipher, the 3DES (E_3, D_3) is
  - $E_3((k_1, k_2, k_3), m) := E(k_1, D(k_2, E(k_3, m)))$
  - $D_3((k_1, k_2, k_3), c) := D(k_3, E(k_2, D(k_1, c)))$
- Key size=3*56=168 bits; 3x slower than DES
- *Meet-in-the-middle* attack: 2^{118} time. (explained in next slide)

Strengthening DES

- Why *not* 2DES?
- **2DES**: let (E, D) be the DES cipher, the 2DES (E_2, D_2) is $E_2((k_1, k_2), m) := E(k_1, E(k_2, m))$
- Key size=2*56=112 bits
- Meet-in-the-middle attack: 112×2^{56} time.
  - Need to find k_1, k_2 s.t. $E(k_1, m) = D(k_2, c)$
  - Cache and sort all possible $E(k_1, m)$ value ($O(n \log n) = 2^{56} \times 56$ time),
  - then check $D(k_2, c)$ for each possible k_2 (same $O(n \log n) = 2^{56} \times 56$ time using binary search)

Advanced Encryption Standard (AES)

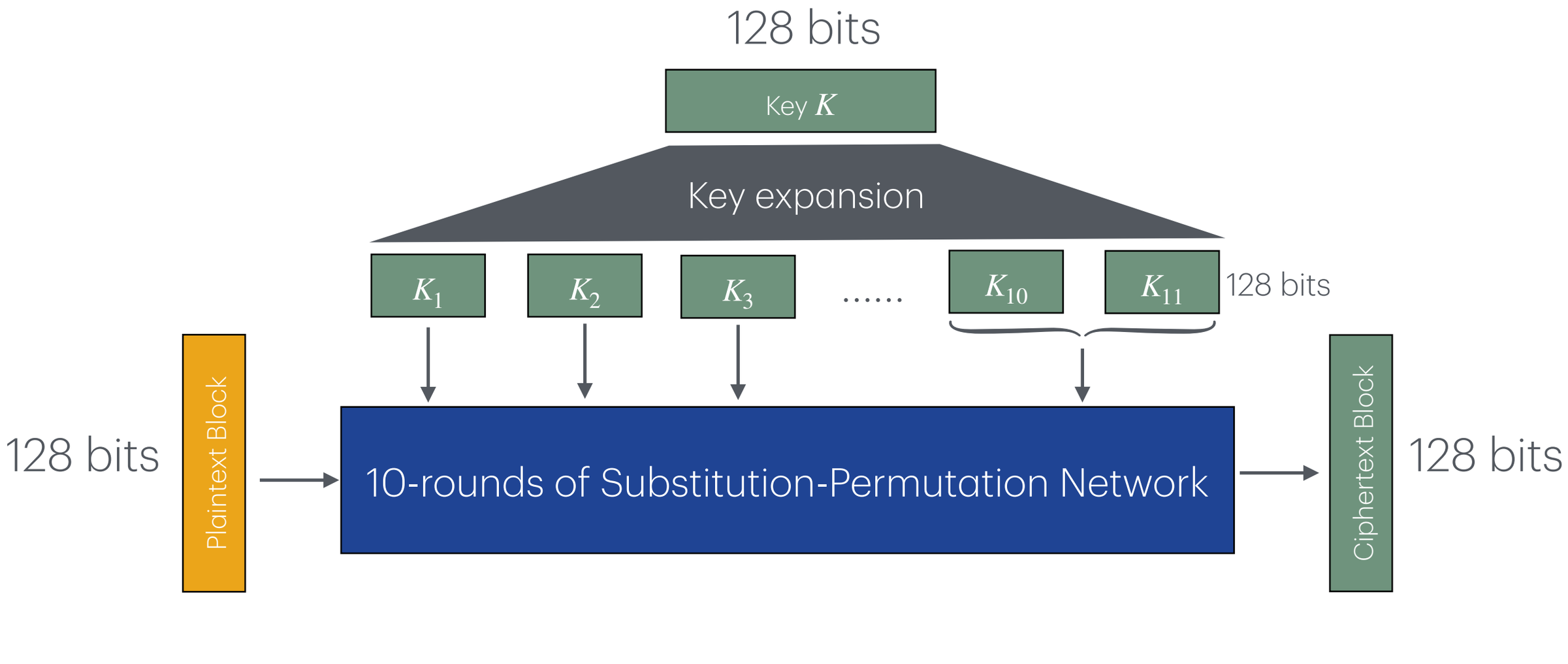
Advanced Encryption Standard (AES)

History

- 1997: NIST publishes request for proposal
- 1998: 15 submissions. Five claimed attacks.
- 1999: NIST chooses 5 finalists
- 2000: NIST chooses **Rijndael** as AES (designed in Belgium)
- Key sizes: 128 / 192 / 256 bits.
  - #Rounds: 10 / 12 / 14
- Block size: 128 bits

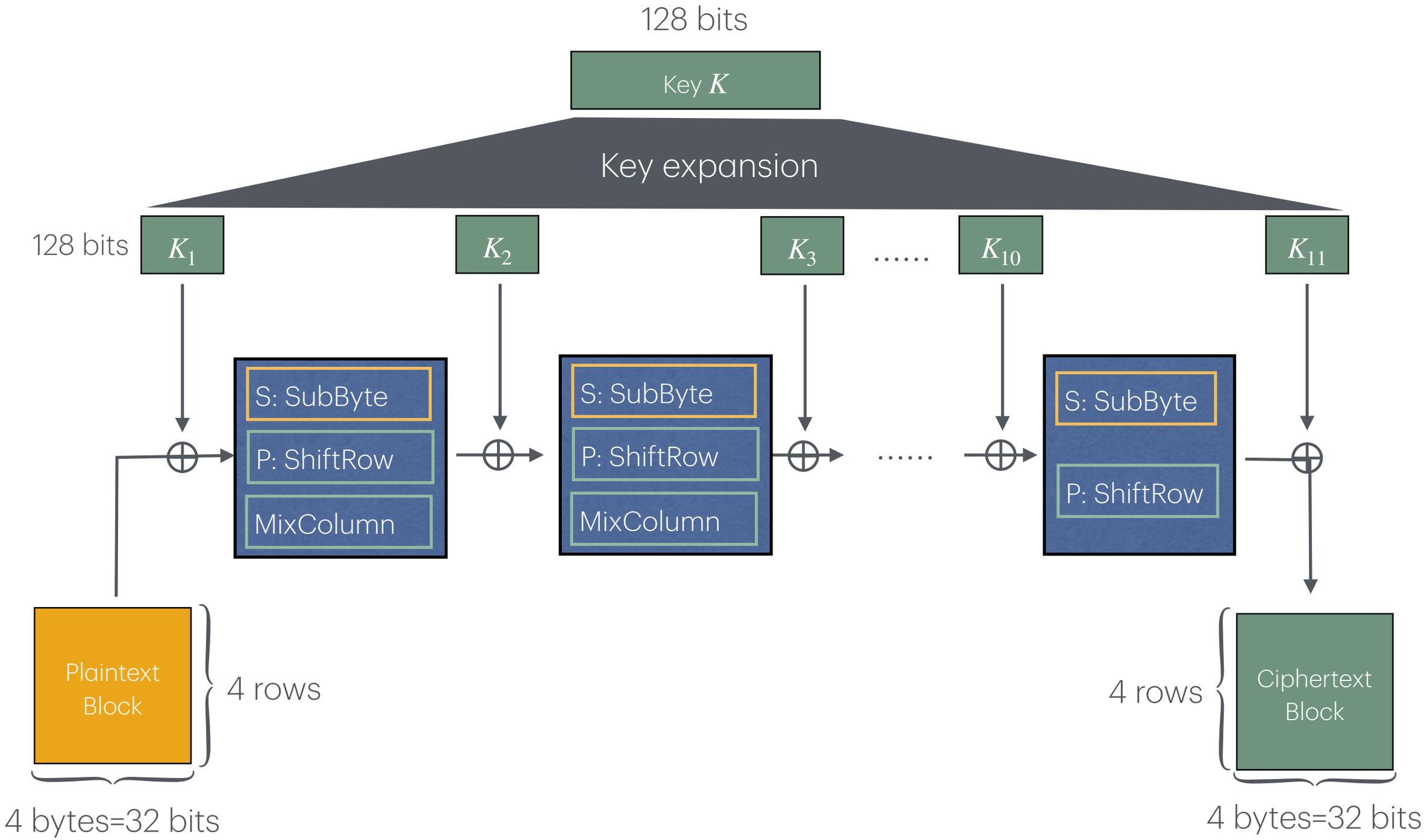
Example: AES-128

- Essentially a 10-round SPN
- The last round will use two round keys.



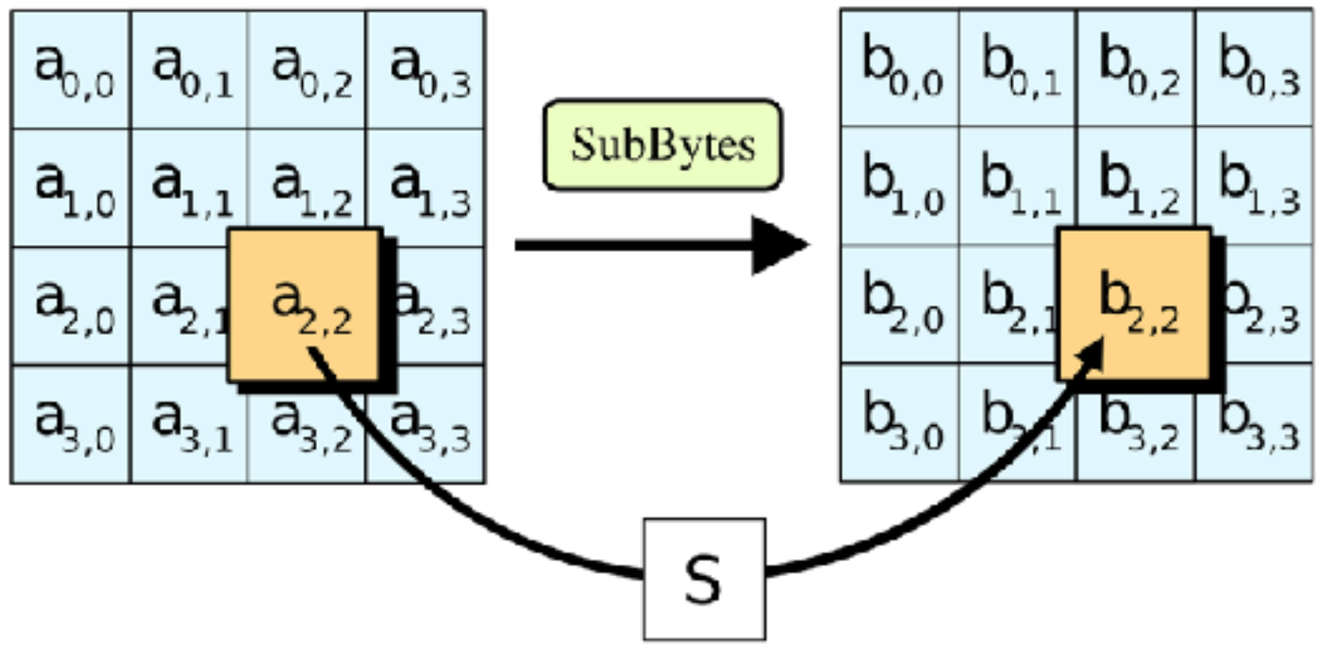
Example: AES-128

Zoom-in



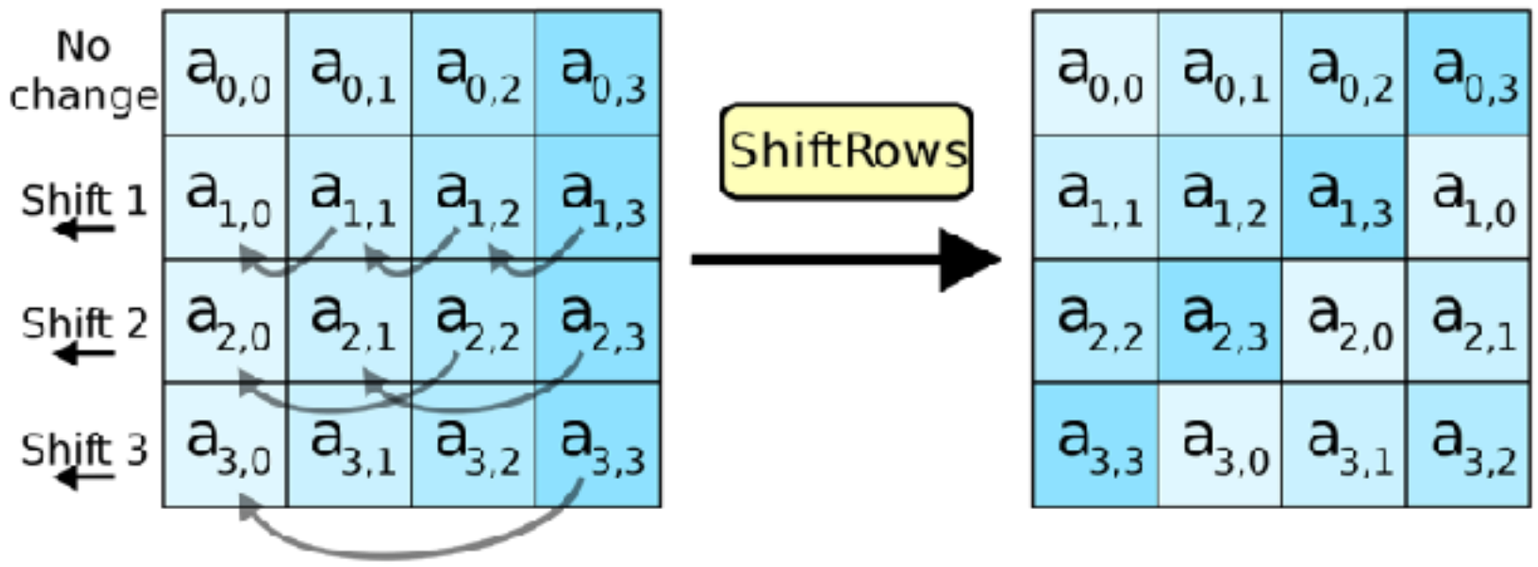
The round function for AES-128

- **(S-Box) ByteSub**: a (fixed) 1-byte lookup table.
  - Each byte in the input block is replaced with another byte.
- Only source of non-linearity.



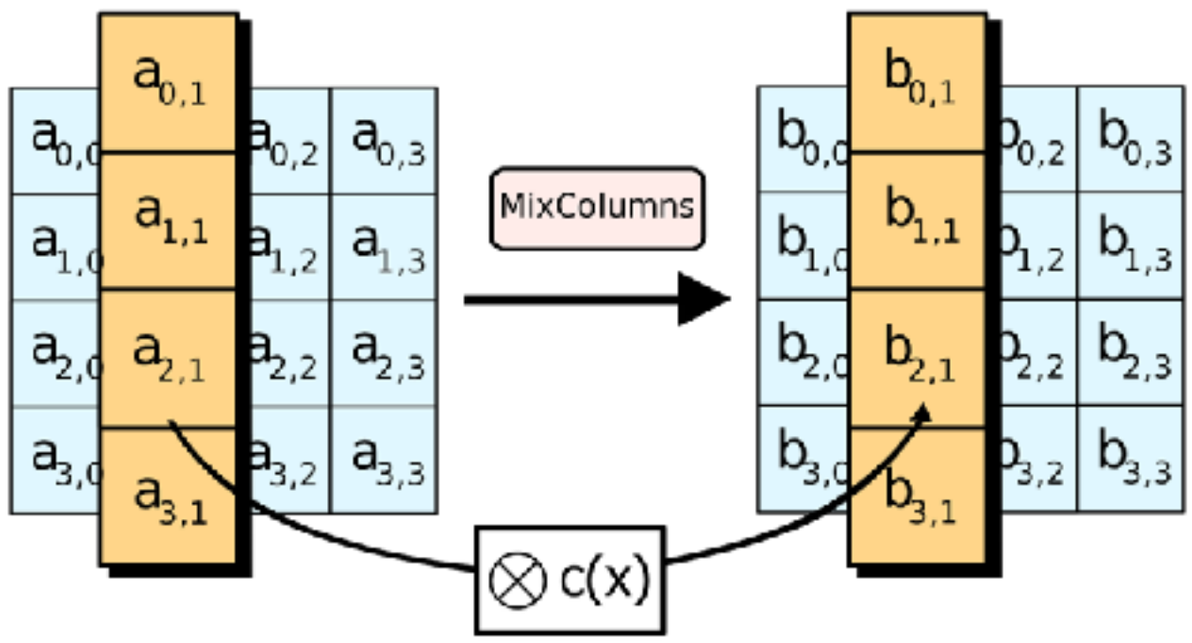
The round function for AES-128

- **(P-Box) ShiftRow**: Simple permutation
  - Each row of the block is shifted cyclically by a different distance
    - Specifically for AES: 0/1/2/3 for the 1st/2nd/3rd/4th row
- Prevent each column from being encrypted independently.



The round function for AES-128

- **MixColumn**: Apply a linear transform on each column
  - Kind-of a substitution, but purely linear
  - Provides further diffusion



Properties of AES

- **Highly secure**
  - Simple design but resistant to known attacks
  - Best *key recovery* attack: four times better than exhaustive search [Bogdanov-Khovratovich-Rechberger'11]
  - *Related key* attack on AES-256: [Biryukov-Khovratovich'09]
    - Given 2^{99} input/output pairs from four related keys in AES-256 can recover keys in time $\approx 2^{99}$
  - Side-channel attack on specific (inproper) implementations.
  - Quantum attack (?): Grover's Algorithm in $2^{n/2}$ time. No quantum computer, though.

Properties of AES

- **Fast**
  - Highly parallelizable and very efficient on a variety of platforms including 8-bit and 64-bit platforms
  - Efficient hardware implementations: E.g. on Intel's AES instruction set
    - **aesenc**, **aesenclast**: do one round of AES
      - 128-bit registers: xmm1=state, xmm2=round key
      - **aesenc xmm1, xmm2** ; puts result in xmm1
    - **aeskeygenassist**: performs AES key expansion
    - Claim 14x speed-up over OpenSSL on same hardware

Block Cipher Modes

Chosen-Plaintext Attack (CPA) Security

- Previously:
  - Information-theoretic Security (perfect secrecy): the ciphertext leaks zero info of the plaintext (except for the length).
    - Secure even if the attacker has unlimited computing power
  - Computation Security: the ciphertext leaks info that can *not* be exploited by a *computationally-bounded* attacker in any meaningful time with non-negligible probability.
- Ciphertext only & Passive Attacker (Eavesdropper) & Only see *one* ciphertext

Chosen-Plaintext Attack (CPA) Security

- This lecture: a stronger notion
  - **Chosen-Plaintext Attack (CPA) Security**: The attacker is allowed to submit arbitrary plaintext(s) to the cipher and receive the corresponding ciphertext(s).
    - The attacker can be *adaptive*.
- The weakest requirement for a realworld cipher.

Chosen-Plaintext Attack (CPA) Security

- Illustration on the formal definition:

Server has a secret bit $b \in \{0,1\}$

Round 0

$(m_0^0, m_1^0), |m_0^0| = |m_1^0|$

$c^0 = E(m_b^0)$

Round 1

$(m_0^1, m_1^1), |m_0^1| = |m_1^1|$

$c^1 = E(m_b^1)$

$\vdots$

$(m_0^q, m_1^q), |m_0^q| = |m_1^q|$

$c^q = E(m_b^q)$

CPA-secure: Attacker cannot infer b significantly better than random guess after multiple rounds of interaction.

Chosen-Plaintext Attack (CPA) Security

- One direct result: a *same* plaintext msg, if encrypted twice, must result in *different* ciphertexts.
  - I.e., a CPA-secure enc. alg. *must be randomized*.

$b = 0$ **if** $c^0 = c^1$

else $b = 1$ **!**

Server has a secret bit $b \in \{0,1\}$

Round 0

$m = m_0^0 = m_1^0$

$c^0 = E(m)$

Round 1

$m = m_0^1 \neq m_1^1$

$c^1 = E(m_b^1)$

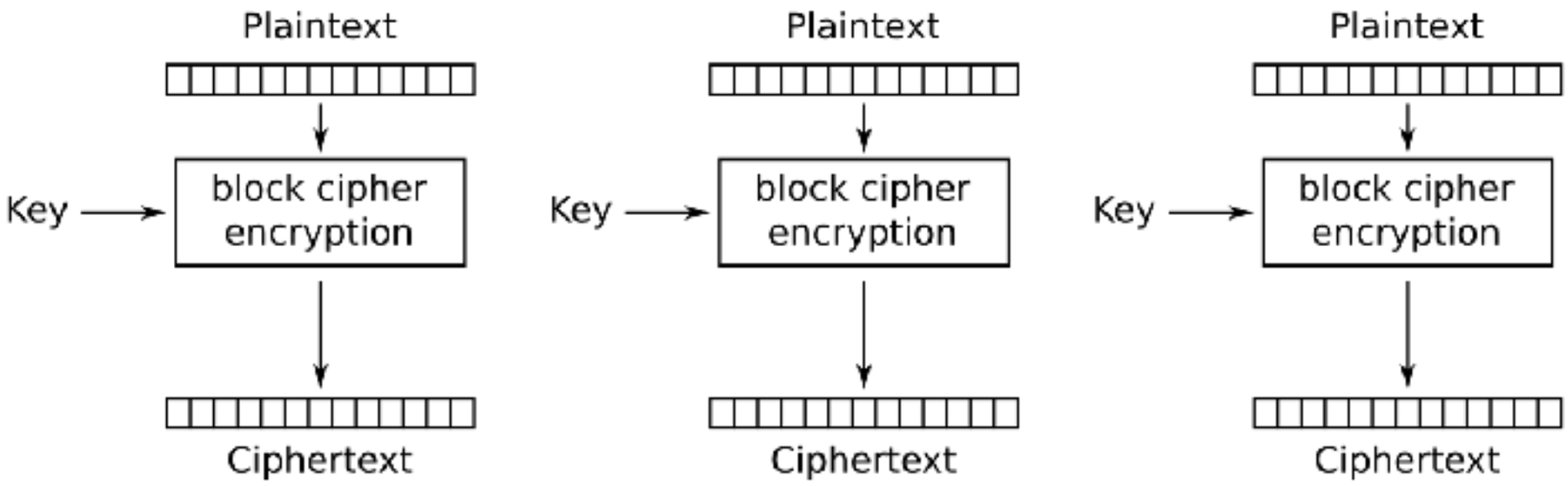
Block Cipher Modes

- **Encryption modes** indicate how messages longer than one block are encrypted and decrypted
- 4 modes of operation were standardized in 1980 for Digital Encryption Standard (DES)
  - Can be used with any block cipher
  - Electronic Codebook Mode (ECB), Cipher Feedback Mode (CFB), Cipher Block Chaining mode (CBC), and Output Feedback mode (OFB)
- 5 modes were specified with the current standard Advanced Encryption Standard (AES) in 2001
  - The 4 above and the Counter mode (CTR)

Electronic Codebook (ECB) mode

The Mode You Shouldn't Use

- Split plaintext to equal-size blocks; Padding the last block.
- **Same** key is used to encrypt every block **independently**.
  - *Same* plaintext block results in the *same* ciphertext block: NOT CPA secure.
  - Reveal statistical info.



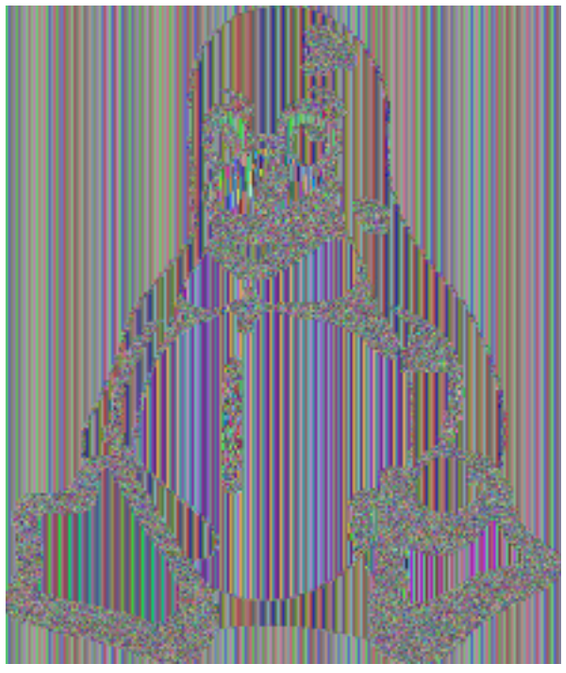
Electronic Codebook (ECB) mode encryption

Electronic Codebook (ECB) mode

The Mode You Shouldn't Use

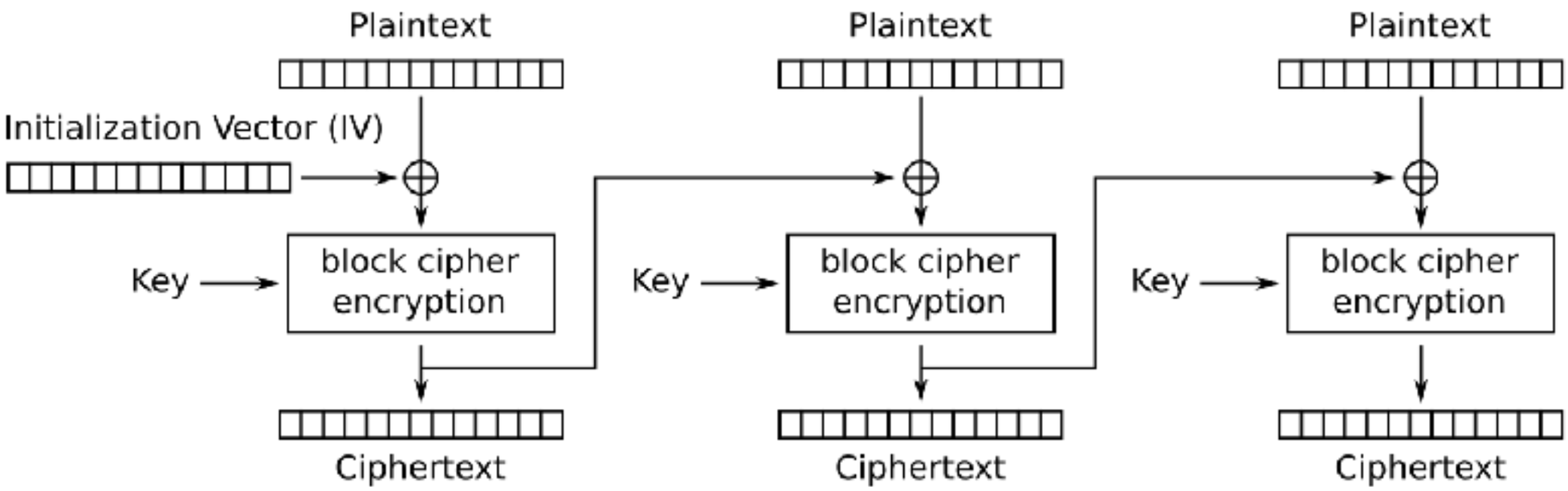
- Split plaintext to equal-size blocks; Padding the last block.
- **Same** key is used to encrypt every block **independently**.
  - *Same* plaintext block results in the *same* ciphertext block: NOT CPA secure.
  - Reveal statistical info.

ECB



Cipher Block Chaining (CBC) mode

- Randomization: use a *random* **Initialization Vector** (IV)
  - For decryption, IV is sent along with the ciphertext
- Dependence between blocks: each block's ciphertext is fed to the next block as IV.



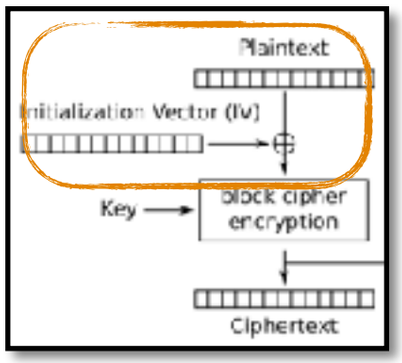
Cipher Block Chaining (CBC) mode encryption

Cipher Block Chaining (CBC) mode

- Properties of CBC mode
  - **Provably CPA-secure** if
    1. the underlying block cipher is secure, and
    2. $q^2L^2 \ll |X|$ (q: #msgs; L: max msg len; $X = \{0,1\}^{128}$)
       - So should change key after $\sim 2^{48}$ AES blocks (for maintaining a $< 1/2^{32}$ attack prob.)
- Sequential encryption, un-parallelizable
- Needs to pad the last block.

Cipher Block Chaining (CBC) mode

- **Warning**: IV must be <u>random</u>, otherwise not CPA-secure.
  - Suppose given ciphertext, attacker can predict the IV for next msg.



Server has a secret bit $b \in \{0,1\}$

$0^n = m_0^0 = m_1^0$

m_0^0: 00000.....0000

m_1^0: 00000.....0000

Round 0

IV_0 $\quad c^0 = E(k, 0 \oplus \text{IV}_0)$

Attacker predict IV_1

$m_0^1 = \text{IV}_1 \oplus \text{IV}_0$

$m_1^1 \neq m_0^1$

Round 1

IV_1 $\quad c^1 = E(k, m_b^1 \oplus \text{IV}_1)$

$b = 0$ **if** $c^0 = c^1$

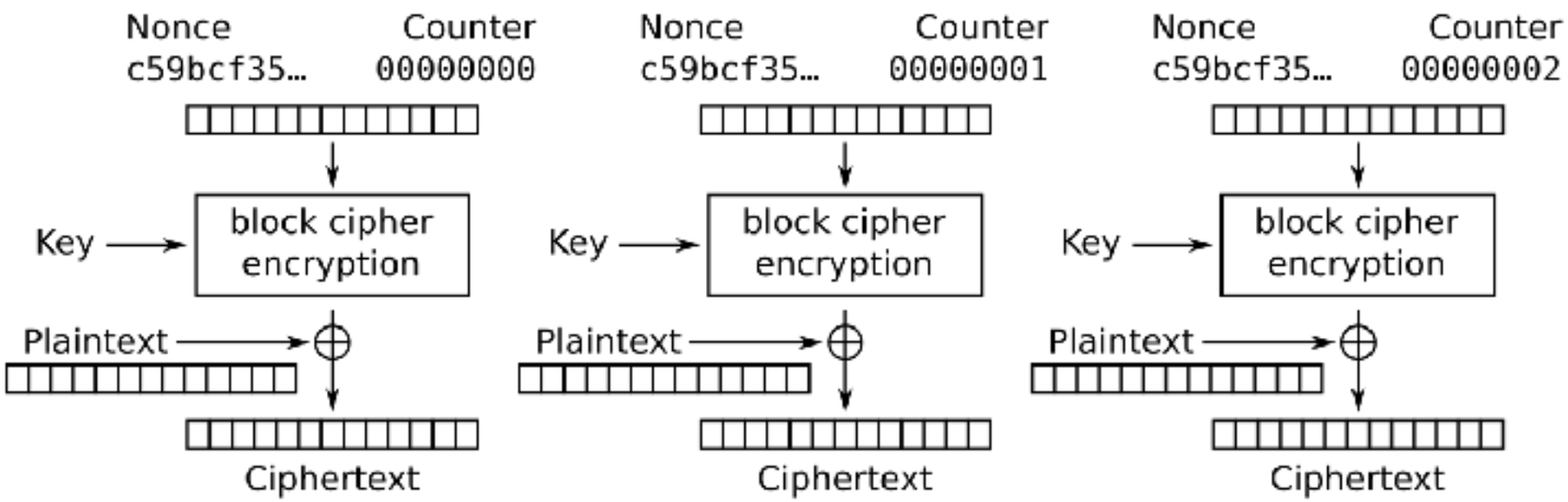
else $b = 1$ **!**

Cipher Block Chaining (CBC) mode

- **Warning**: IV must be random, otherwise not CPA-secure.
  - Suppose given ciphertext, attacker can predict the IV for next msg.
- This predictable-IV bug exists in the implementation of SSL/TLS 1.0
  - IV for record i is last Ciphertext block of record $i-1$
  - Long-thought to be only theoretical and cannot be exploited in practice
  - However, a successful man-in-the-middle attack was found based on this vulnerability [Duong-Rizzo, 2011]

Counter (CTR) mode

- Use block cipher as PRG to create one-time pad. Like stream cipher.
- [The pad for block i] $= E(k, \text{nounce} \,||\, \text{counter}_i)$
  - nounce: **must** be *unique* for each *message;* may *not* be random.
  - counter: unique for each *block;* usually just a numbering, i.e., not random.



Counter (CTR) mode encryption

Counter (CTR) mode

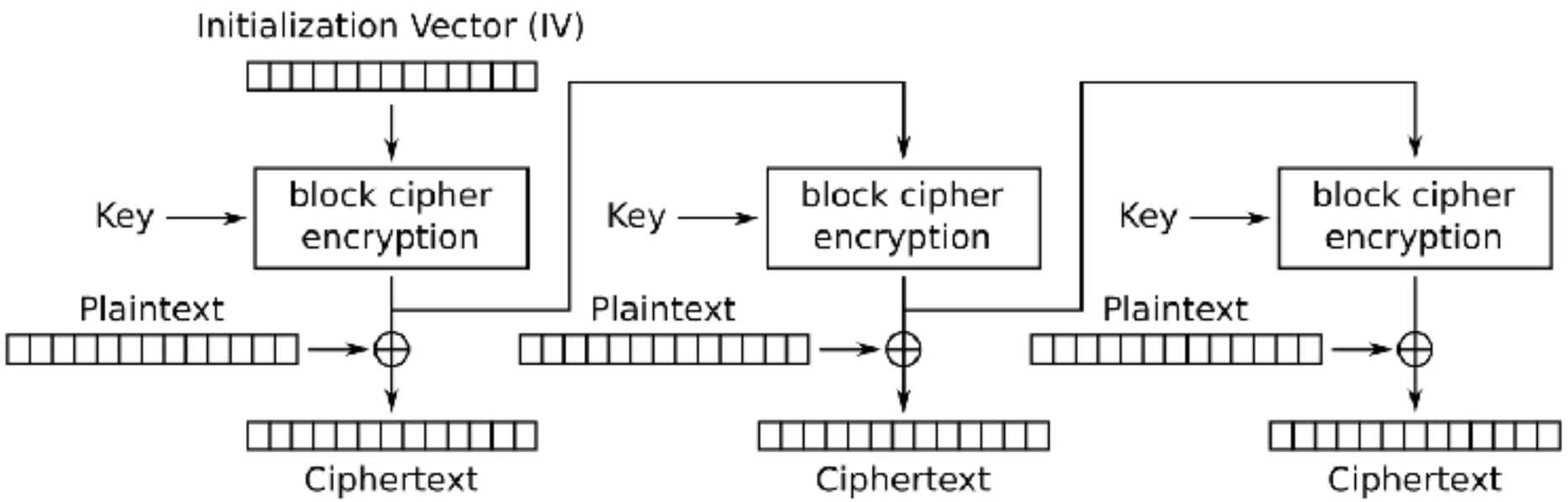
- Properties of CTR mode
  - **Provably CPA-secure** if
    1. the underlying block cipher is secure, and
    2. $q^2 L \ll |X|$ (q: #msgs; L: max msg len; $X = \{0,1\}^{128}$)
       - So should change key after $\sim 2^{64}$ AES blocks (for maintaining a $< 1/2^{32}$ attack prob.)
  - Highly-parallelizable.
  - Asynchronously: the pad can be generated in advance, independent of the plaintext; the rest is just XOR.
  - No padding needed.

CBC vs CTR

	CBC	CTR
Parallel?	No	Yes
Security	$q^2L^2 \ll \|X\|$	$q^2L \ll \|X\|$
Padding?	Yes	No
1 byte msg expansion	16x	no expansion

Other Block Cipher Modes

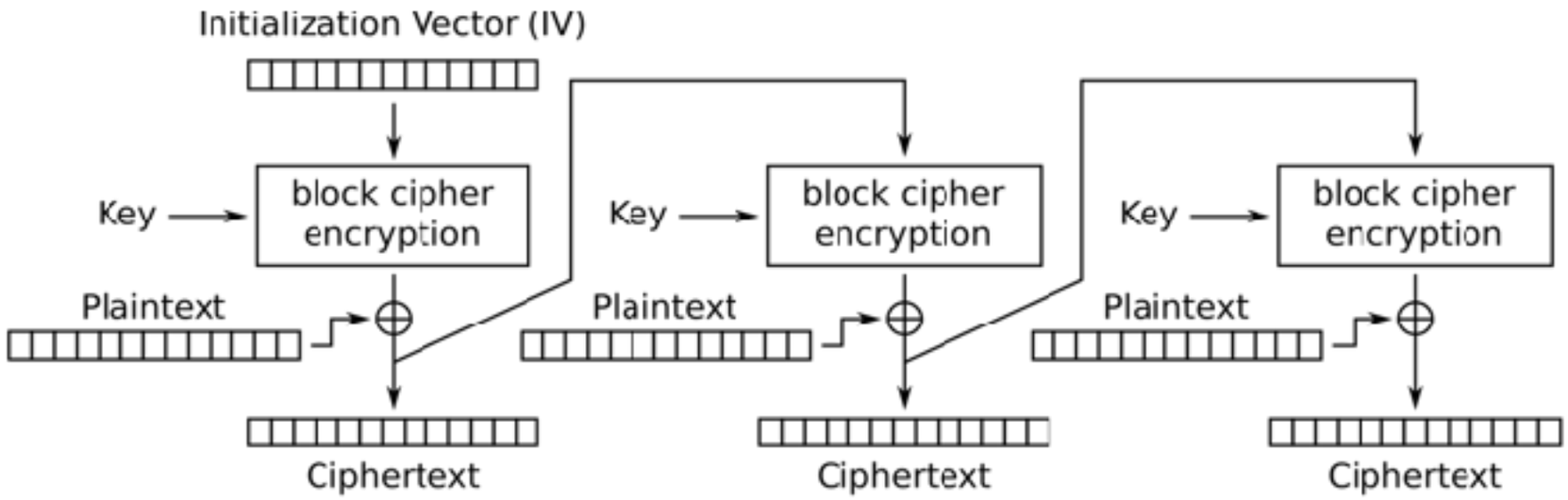
- **Output Feedback (OFB) mode**:
  - Act as PRG to generate one-time pad.
  - Not parallelizable: the pad from previous block act as IV for the next block.
  - Asynchronous: the pad can be generated in advance, independent of plaintext.
  - CPA-secure



Output Feedback (OFB) mode encryption

Other Block Cipher Modes

- **Cipher Feedback (CFB) mode**:
  - Mix of OFB and CBC
  - Not parallelizable: the ciphertext from previous block act as IV for the next block.
  - CPA-secure



Cipher Feedback (CFB) mode encryption

Summary

- DES
  - Construction
  - Attack and Strengthening
- AES
  - Construction
- Block Cipher Modes
  - CPA-security
  - ECB mode: do not use
  - CBC: IV + Chaining blocks
  - CTR: with nounce+counter, act as a PRG.
  - OFB & CFB

Acknowledgement

- The slides of this lecture is developed heavily based on
  - Slides from Prof Dan Boneh's lecture on Cryptography
  - Slides from Prof Ziming Zhao's lecture on Symm. Encryption

Questions ?