CIS 6930/4930 Computer and Network Security

Topic 3.1 Secret Key Cryptography (Cont'd)

Principles for S-Box Design

- S-box is the only non-linear part of DES
- Each row in the S-Box table should be a permutation of the possible output values
- Output of one S-box should affect other S-boxes in the following round

Desirable Property: Avalanche Effect

- Roughly: a small change in either the plaintext or the key should produce a big change in the ciphertext
- Better: any output bit should be inverted (flipped) with probability 0.5 if any input bit is changed
  - What is the expected number of different bits between two length-*n* random bit streams?
    - 0.5n (Bernoulli distribution)
- *f* function
  - must be difficult to un-scramble
  - should achieve avalanche effect
  - output bits should be uncorrelated

DES Avalanche Effect: Example

- 2 plaintexts with 1 bit difference:

  0x0000000000000000 and

  0x8000000000000000

  encrypted using the same key:

  0x016B24621C181C32

- Resulting ciphertexts differ in 34 bits (out of 64)
- Similar results when keys differ by 1 bit

Example (cont'd)

- An experiment: number of rounds vs. number of bits difference

Round #	0	1	2	3	4	5	6	7	8
Bits changed	1	6	21	35	39	34	32	31	29

9	10	11	12	13	14	15	16
42	44	32	30	30	26	29	34

DES: Keys to Avoid Using

- "Weak keys": These are keys which, after the first key permutation, are:
  - 28 0's followed by 28 0's
  - 28 1's followed by 28 1's
  - 28 0's followed by 28 1's
  - 28 1's followed by 28 0's
- Property of weak keys
  - Easy clue for brute force attacks.
  - Sixteen identical subkeys.
  - Encrypting twice produces the original plaintext.

Weak keys

- Alternating ones + zeros (0x0101010101010101)
- Alternating 'F' + 'E' (0xFEFEFEFEFEFEFEFE)
- 0xE0E0E0E0F1F1F1F1
- 0x1F1F1F1F0E0E0E0E
- DES *weak keys* produce sixteen identical subkeys

More Keys to Avoid!

- "Semi-weak keys": These are keys which, after the first key permutation, are:
  1. 28 0's followed by alternating 0's and 1's
  2. 28 0's followed by alternating 1's and 0's

  …

  12. Alternating 1's and 0's followed by alternating 1's and 0's
- Property of semi-weak keys
  - For a semi-weak key pair (K_1, K_2), $K_1\{K_2\{m\}\} = m$

Semi-weak key pairs

- 0x011F011F010E010E and 0x1F011F010E010E01
- 0x01E001E001F101F1 and 0xE001E001F101F101
- 0x01FE01FE01FE01FE and 0xFE01FE01FE01FE01
- 0x1FE01FE00EF10EF1 and 0xE01FE01FF10EF10E
- 0x1FFE1FFE0EFE0EFE and 0xFE1FFE1FFE0EFE0E
- 0xE0FEE0FEF1FEF1FE and 0xFEE0FEE0FEF1FEF1
- $K_1\{K_2\{m\}\} = m$

DES Key Size

- 56 bits is currently too small to resist brute force attacks using readily-available hardware
- Ten years ago it took $250,000 to build a machine that could crack DES in a few hours
- Now?

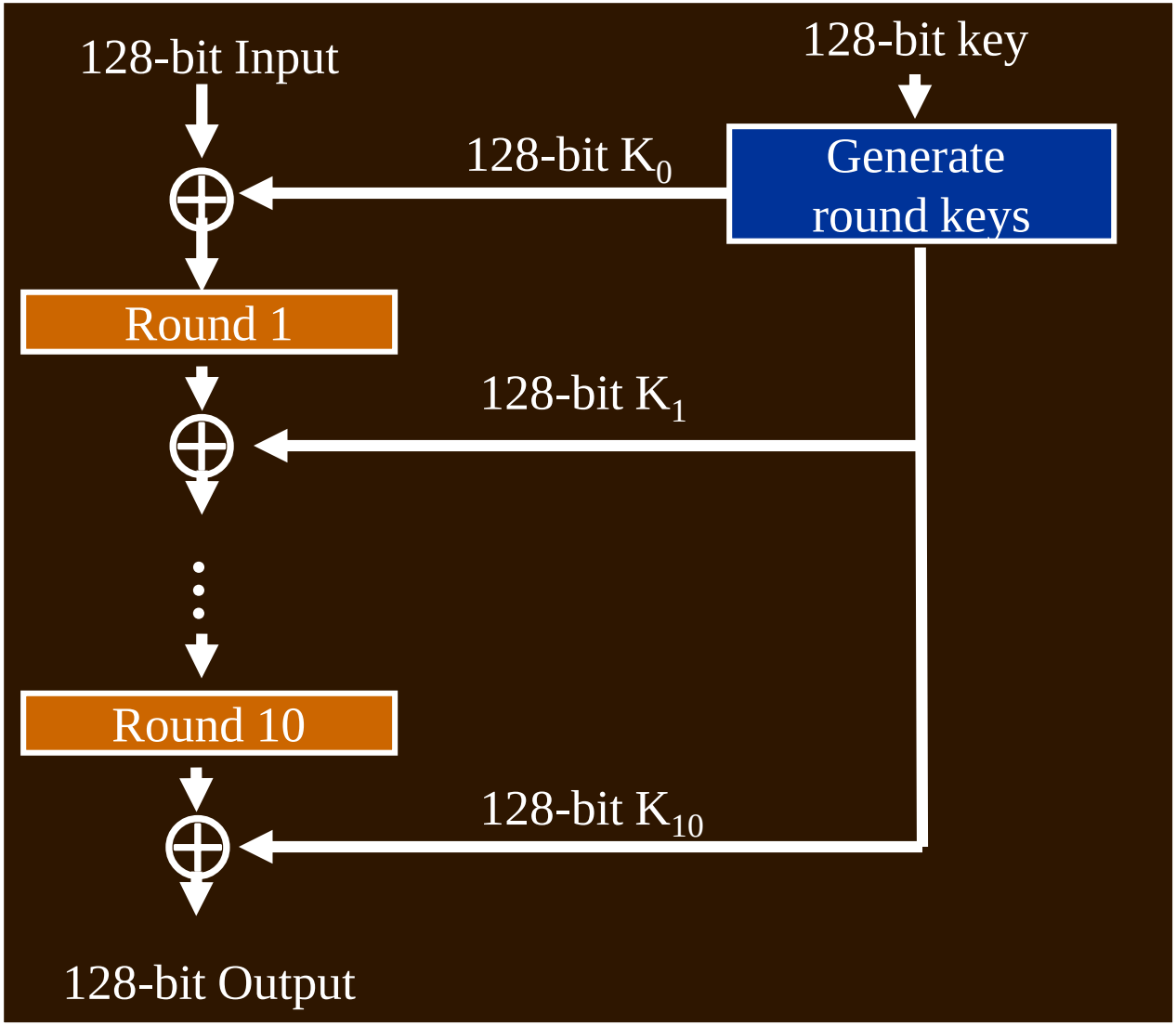
Cryptanalysis of DES

- Differential cryptanalysis exploits differences between encryptions of two different plaintext blocks
  - provides insight into possible key values
- Linear cryptanalysis requires known plaintext / ciphertext pairs, analyzes relationships to discover key value
- No attacks on DES so far are significantly better than brute force attacks, for comparable cost

Advanced Encryption Standard (AES)

- Selected from an open competition, organized by NSA
  - winner: Rijndael algorithm, standardized as AES
- Some similarities to DES (rounds, round keys, alternate permutation+substitution)
  - but not a Feistel cipher
- Block size = 128 bits
- Key sizes = 128, 192, or 256

AES-128 Overview



AES Assessment

- No known successful attacks on full AES
  - Best attacks work on 7–9 rounds
- For brute force attacks, AES-128 needs much more effort than DES

Attacks on AES

- Differential Cryptanalysis: based on how differences in inputs correlate with differences in outputs
- Linear Cryptanalysis: based on correlations between input and output
- Side Channel Attacks: Implementations of the cipher on systems inadvertently leak data
  - Timing Attacks: measure the time it takes to do operations

CIS 6930/4930 Computer and Network Security

Topic 3.2 Modes of Operation

Processing with Block Ciphers

- Most ciphers work on blocks of fixed (small) size
- How to chain cipher text together?
- Modes of operation
  - ECB (Electronic Code Book)
  - CBC (Cipher Block Chaining)
  - OFB (Output Feedback)
  - CFB (Cipher Feedback)
  - CTR (Counter)

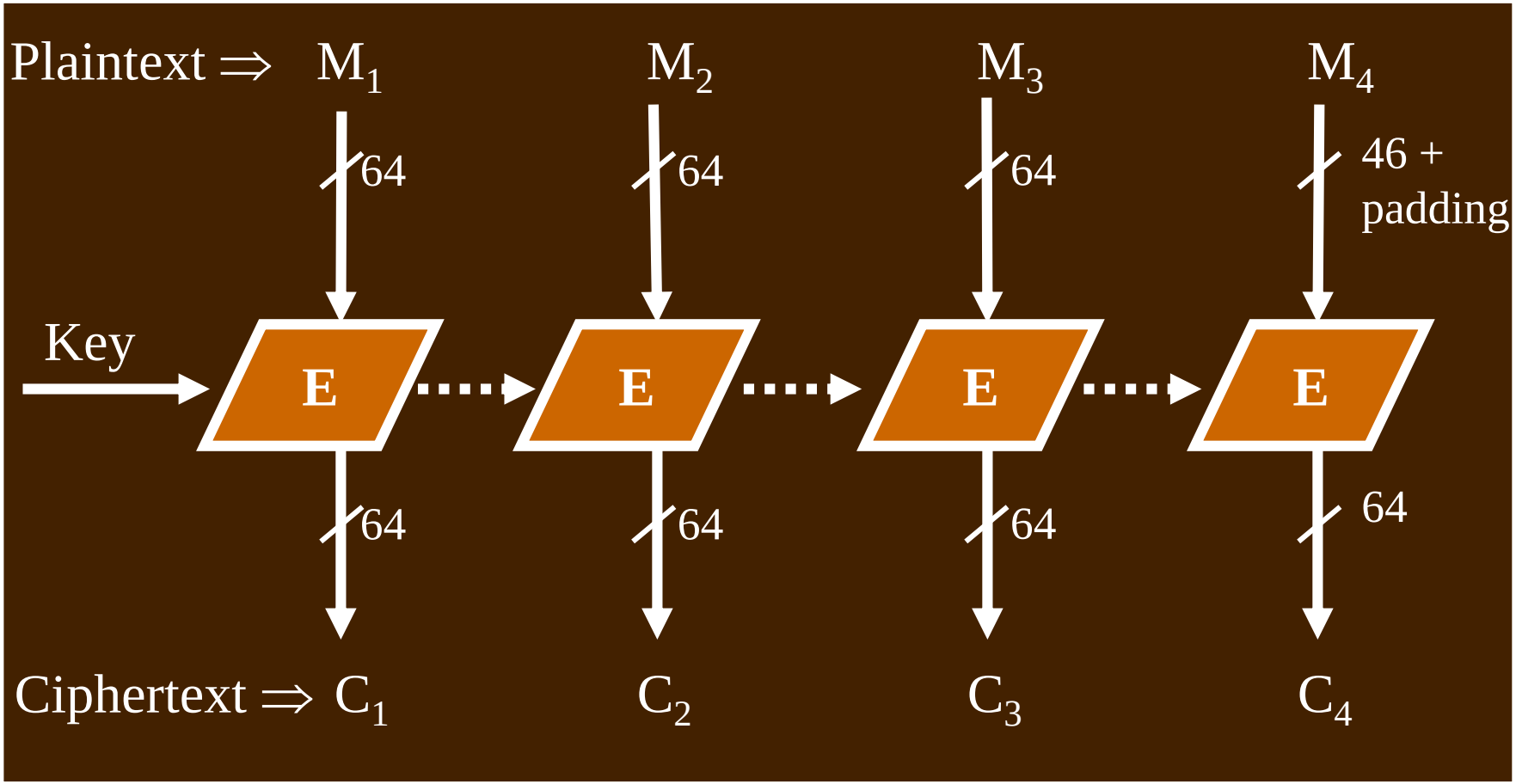
Issues for Block Chaining Mode

- Ciphertext manipulation
  - Can an attacker modify ciphertext block(s) in a way that will produce a predictable/desired change in the decrypted plaintext block(s)?
  - Note: assume the structure of the plaintext is known, e.g., first block is employee #1 salary, second block is employee #2 salary, etc.
- Information leakage
  - Does it reveal info about the plaintext blocks?

Issues… (Cont'd)

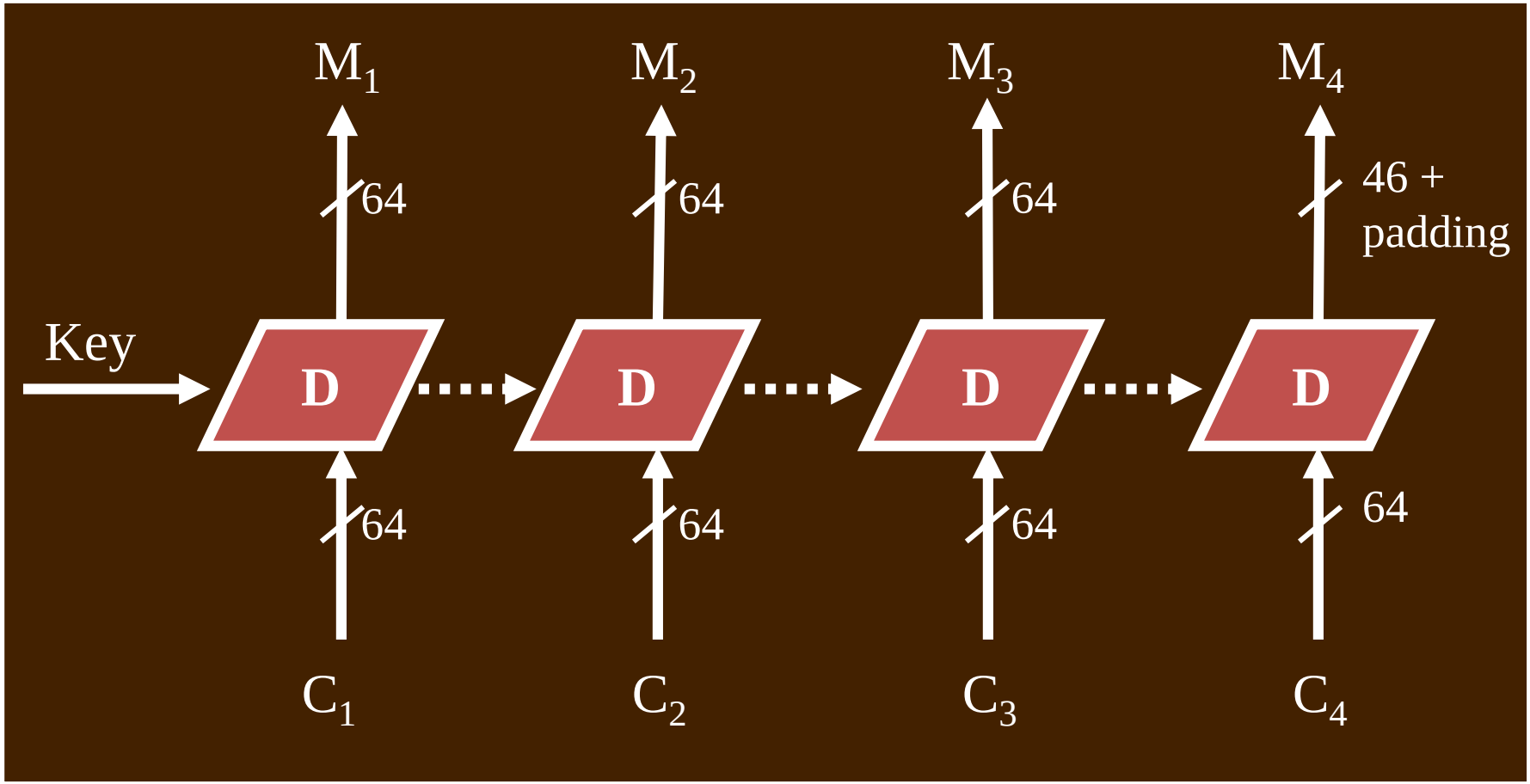
- Parallel/Sequential
  - Can blocks of plaintext (ciphertext) be encrypted (decrypted) in parallel?
- Error propagation
  - If there is an error in a plaintext (ciphertext) block, will there be an encryption (decryption) error in more than one ciphertext (plaintext) block?

Electronic Code Book (ECB)



- The easiest mode of operation; each block is independently encrypted

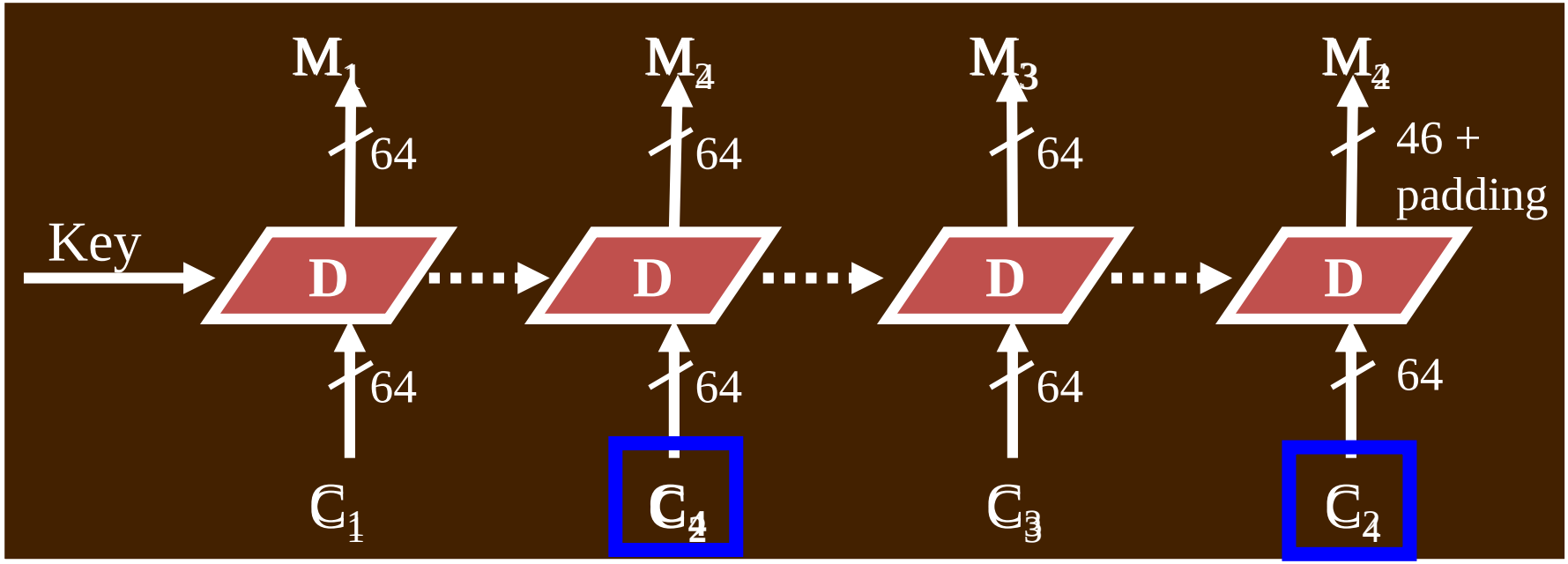
ECB Decryption



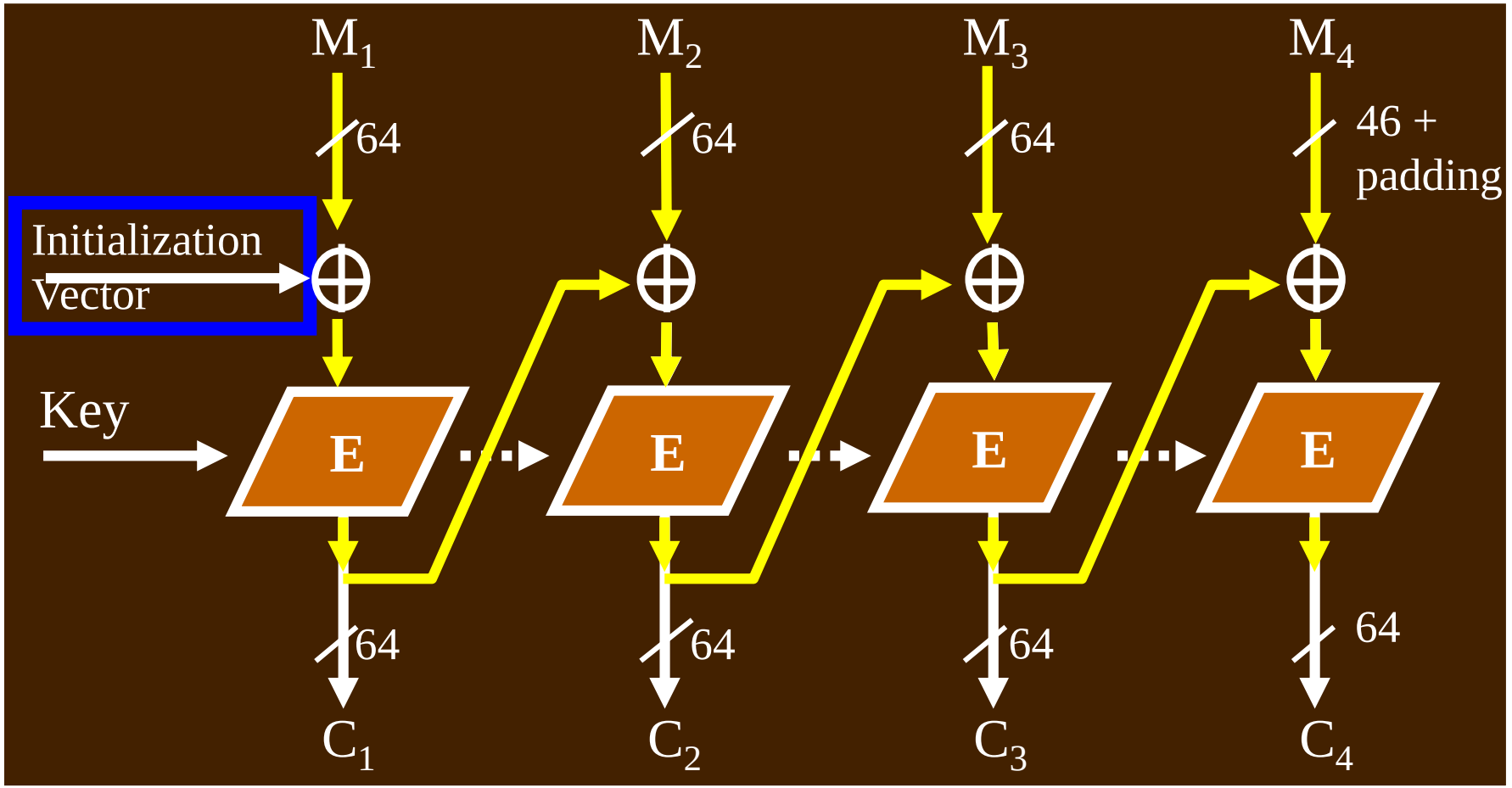
- Each block is independently decrypted

ECB Properties

- Does information leak?
- Can ciphertext be manipulated?
- Parallel processing possible?
- Do ciphertext errors propagate?



Cipher Block Chaining (CBC)

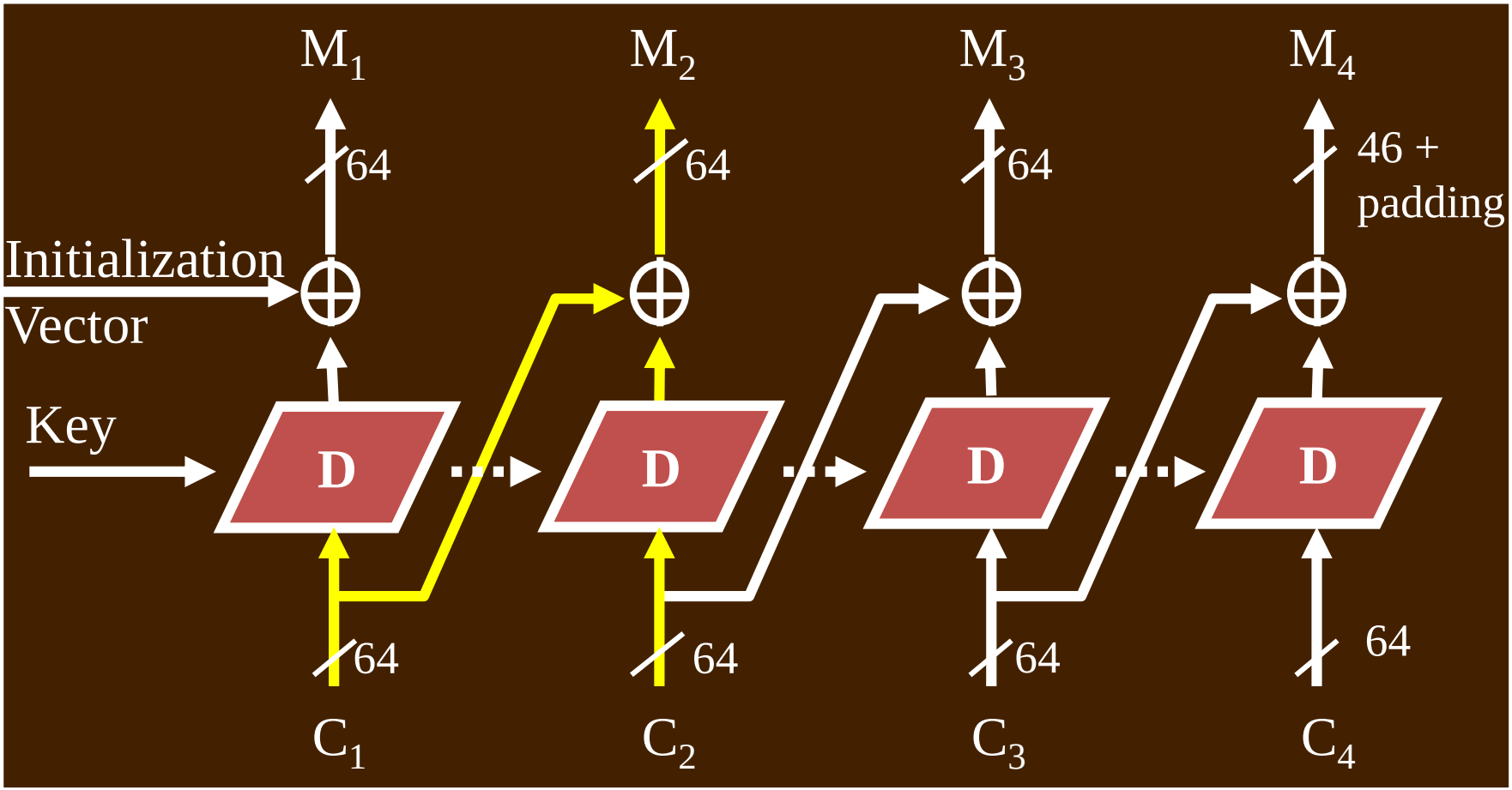


- Chaining dependency: each ciphertext block depends on all preceding plaintext blocks

Initialization Vectors

- Initialization Vector (IV)
  - Used along with the key; not secret
  - For a given plaintext, changing either the key, or the IV, will produce a different ciphertext
  - Why is that useful?
- IV generation and sharing
  - Random; may transmit with the ciphertext
  - Incremental; predictable by receivers

CBC Decryption

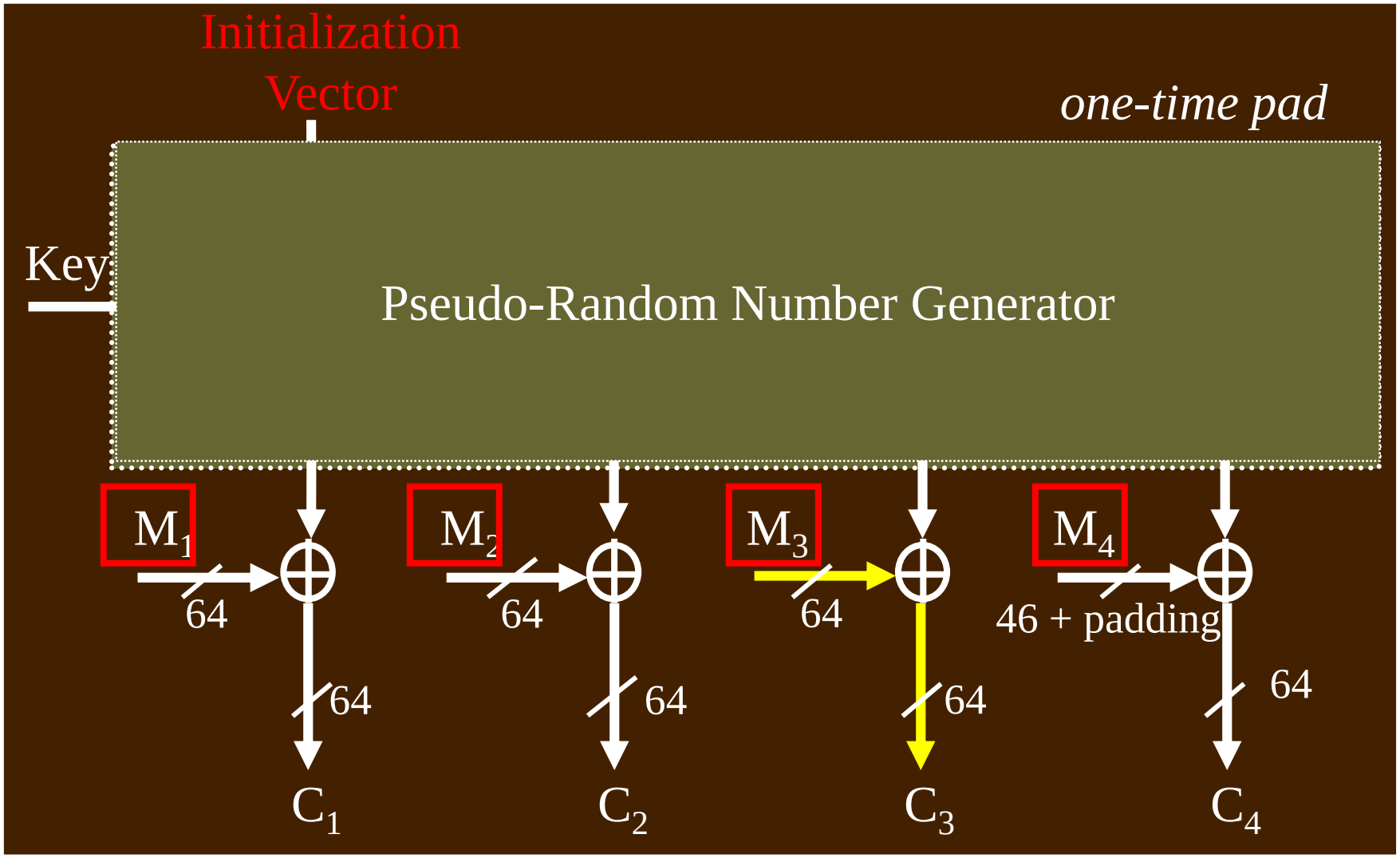


- How many ciphertext blocks does each plaintext block depend on?

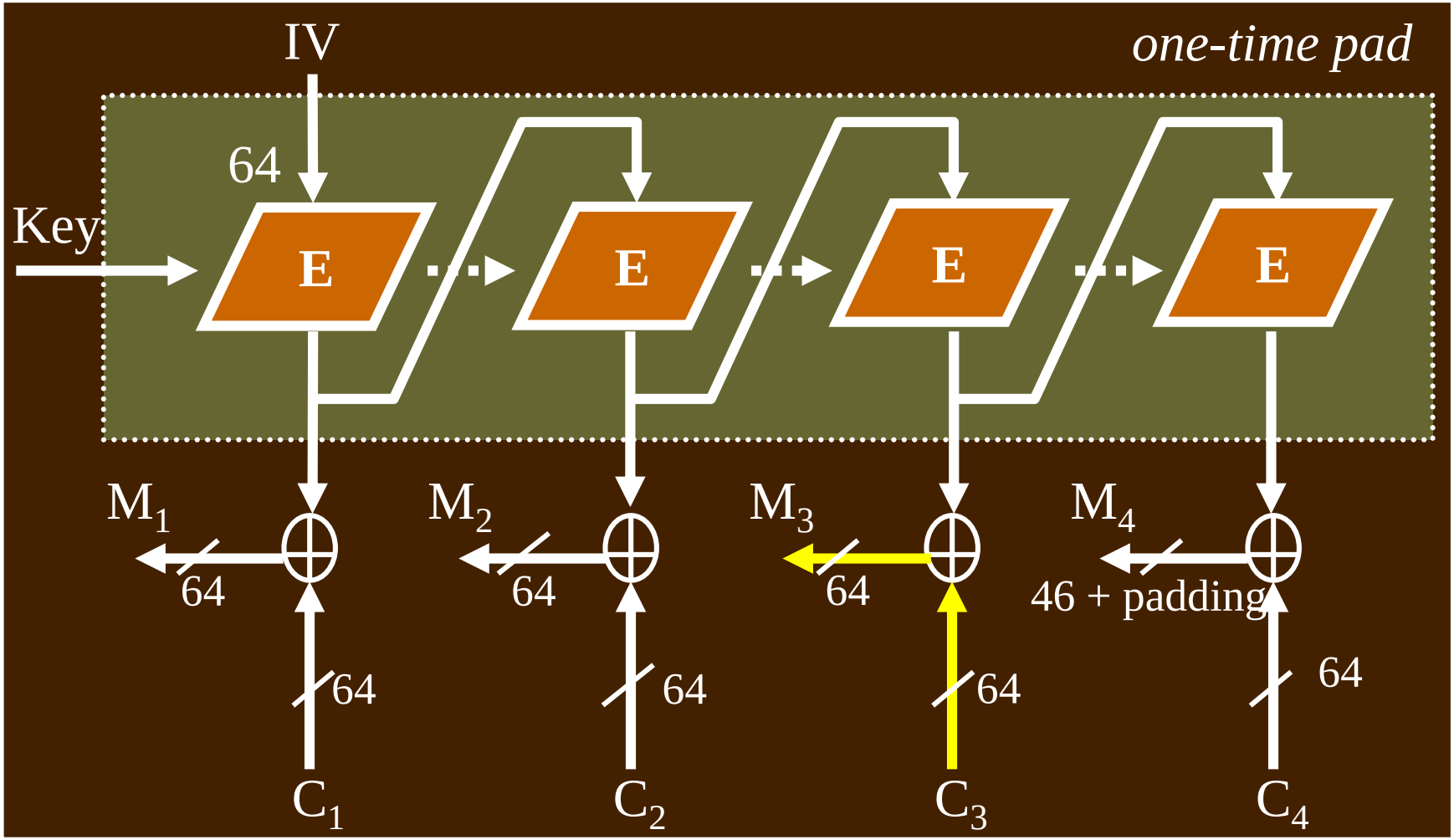
CBC Properties

- Does information leak?
  - Identical plaintext blocks will produce different ciphertext blocks
- Can ciphertext be manipulated predictably?
  - ???
- Parallel processing possible?
  - no (encryption), yes (decryption)
- Do ciphertext errors propagate?
  - yes (encryption), a little (decryption)

Output Feedback Mode (OFB)



OFB Decryption



No block decryption required!

OFB Properties

- Does information leak?
  - identical plaintext blocks produce different ciphertext blocks
- Can ciphertext be manipulated predictably?
  - ???
- Parallel processing possible?
  - yes (generating pad), yes (XORing with blocks)
- Do ciphertext errors propagate?
  - ???

OFB … (Cont'd)

- If you know one plaintext/ciphertext pair, can easily derive the one-time pad that was used
  - i.e., should not reuse a one-time pad!
- Conclusion: IV must be different every time