CSEN1001

Computer and Network Security

Mervat AbuElkheir
Mohamed Abdelrazik
Ahmad Helmy

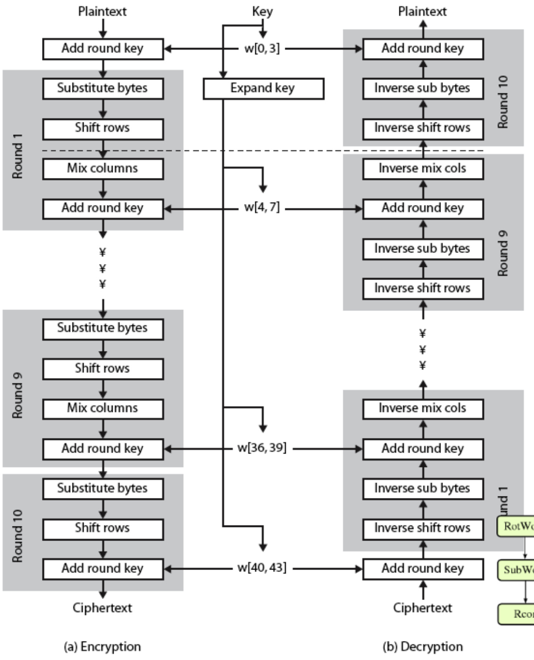
Lecture (5)

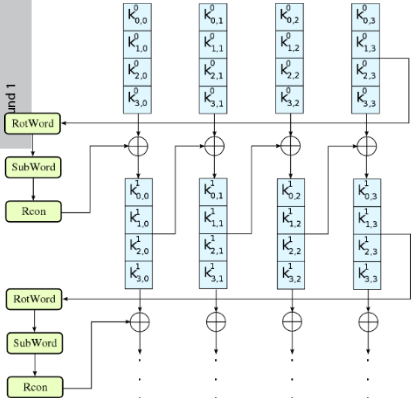
Modes of Operation

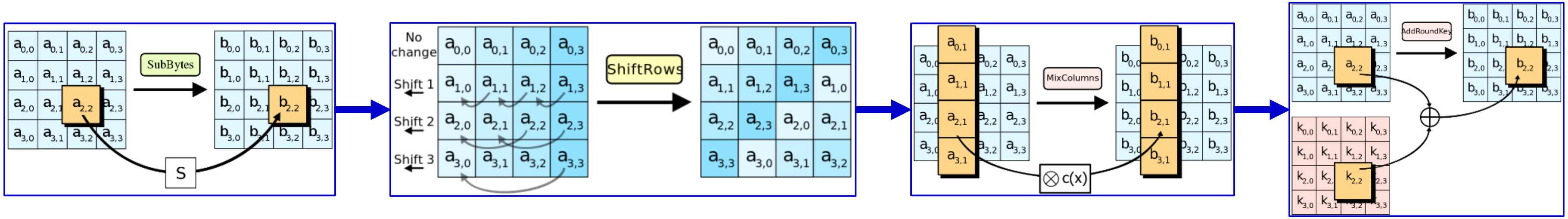
Recall: One AES Round

Remember

- Each round consists of 4 processes
- All n rounds are applied to one block of plaintext!
- For a nice detailed explanation of AES, refer to: https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture8.pdf





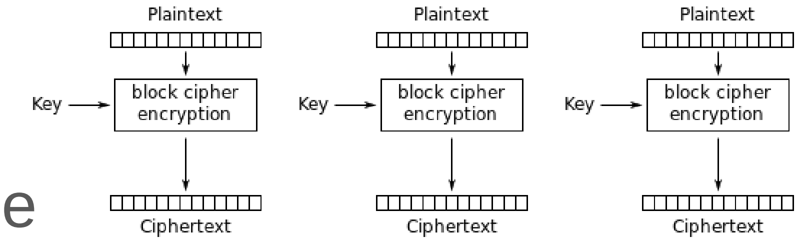


Modes of Operation

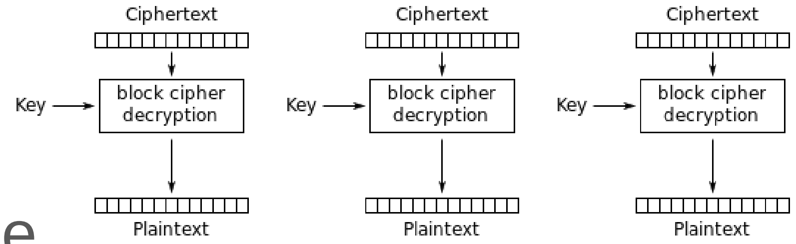
- Block ciphers encrypt fixed size blocks
  - eg. DES encrypts 64-bit blocks with 56-bit key
- Need some way to en/decrypt arbitrary amounts of data in practise
- **ANSI X3.106-1983 Modes of Use** (now FIPS 81) defines 4 possible modes
- Subsequently 5 defined for AES & DES
- Have **block** and **stream** modes

Electronic Codebook Mode (ECB)

- Message is broken into independent blocks which are encrypted
- Each block is a value which is substituted, like a codebook, hence name
- Each block is encoded independently of the other blocks
  - $C_i = DES_{K1}(P_i)$
- Uses: secure transmission of single values



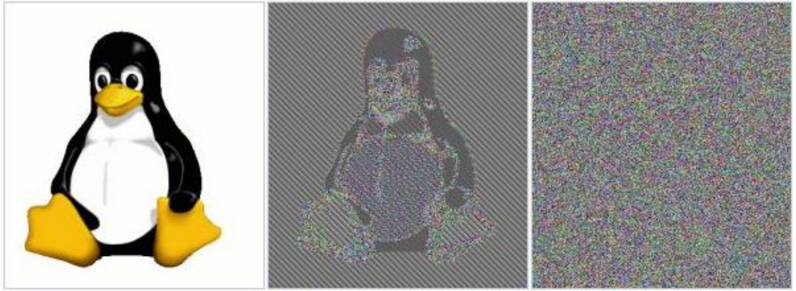
Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

Limitations of ECB

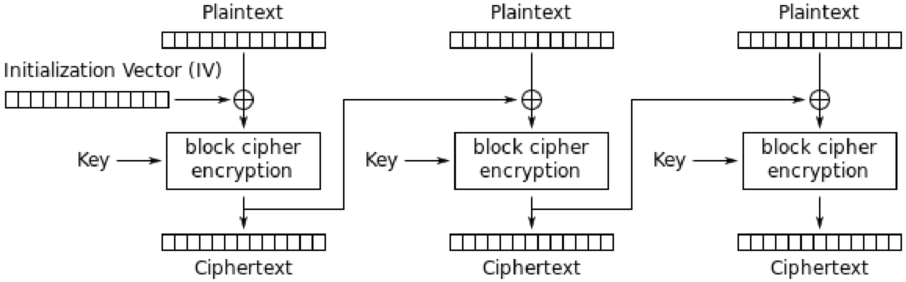
- Message repetitions may show in ciphertext
  - if aligned with message block
  - particularly with data such as graphics
  - or with messages that change very little, which become a code-book analysis problem
- Weakness is due to the encrypted message blocks being independent
- Main use is sending a few blocks of data



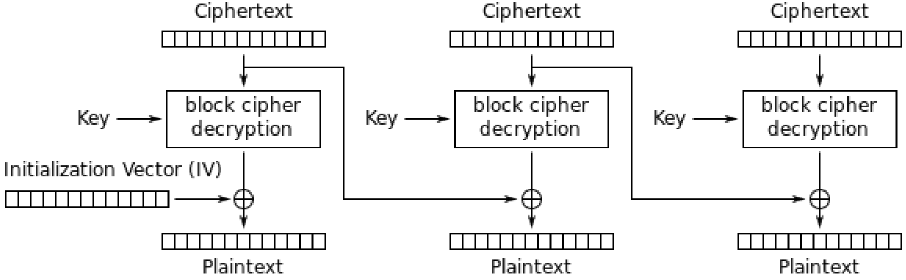
Original image | Encrypted using ECB mode | Modes other than ECB result in pseudo-randomness

Cipher Block Chaining Mode (CBC)

- Message is broken into blocks
- Linked together in encryption operation
- Each previous cipher block is chained with current plaintext block, hence name
- Use **Initial Vector (IV)** to start process
  - $C_i = DES_{K1}(P_i \text{ XOR } C_{i-1})$
  - $C_0 = IV$
- Uses: bulk data encryption, authentication



Cipher Block Chaining (CBC) mode encryption



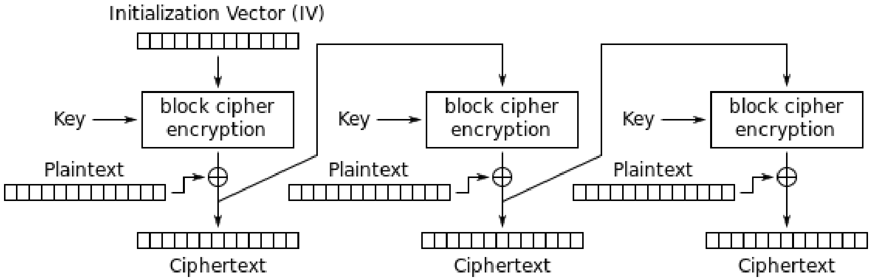
Cipher Block Chaining (CBC) mode decryption

Limitations of CBC

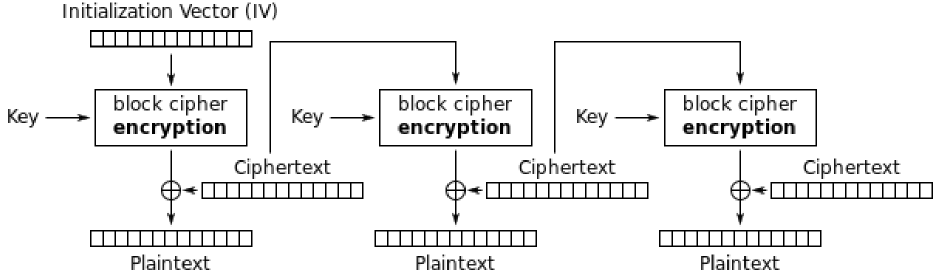
- A ciphertext block depends on **all** blocks before it
- Any change to a block affects all following ciphertext blocks
- Need **Initialization Vector** (IV)
  - which must be known to sender & receiver
  - if predictable, attacker can change bits of first block, and change IV to compensate
    - $C_1 = E(K, [IV \oplus P_1])$
    - $P_1 = IV \oplus D(K, C_1)$
    - $P_1[i] = IV[i] \oplus D(K, C_1)[i]$
    - $P_1[i]' = IV[i]' \oplus D(K, C_1)[i]$
  - hence IV must be an unpredictable value
  - can be sent encrypted in ECB mode before rest of message

Cipher Feedback Mode (CFB)

- Message is treated as a stream of bits
- Added to the output of the block cipher
- Result is feed back for next stage (hence name)
- Standard allows any number of bits (1,8, 64 or 128 etc.) to be feed back
  - denoted CFB-1, CFB-8, CFB-64, CFB-128 etc.
- Most efficient to use all bits in block (64 or 128)
  - $C_i = P_i \text{ XOR } DES_{K1}(C_{i-1})$
  - $C_0 = IV$
- Uses: stream data encryption, authentication



Cipher Feedback (CFB) mode encryption

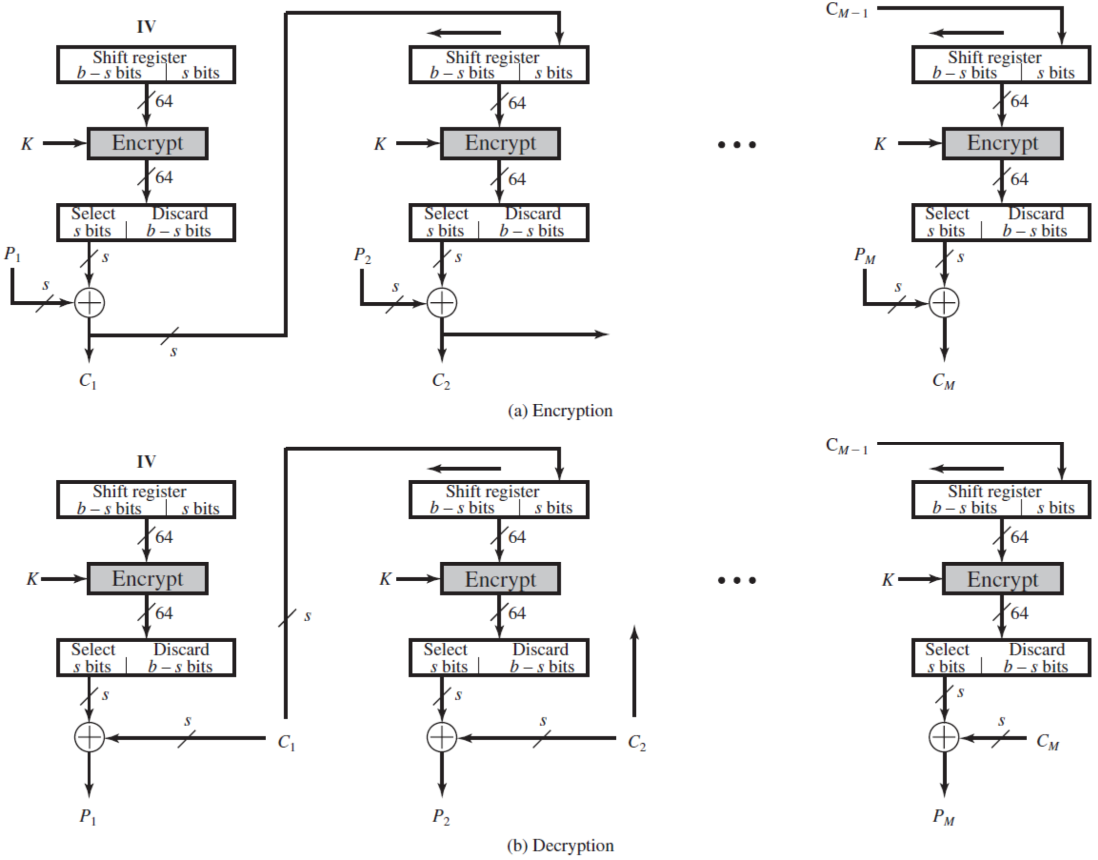


Cipher Feedback (CFB) mode decryption

Cipher Feedback Mode (CFB)

The use of shift registers to enable self-synchronization

- If x bits are lost from the ciphertext, the cipher will output incorrect plaintext until the shift register once again equals a state it held while encrypting, at which point the cipher has resynchronized

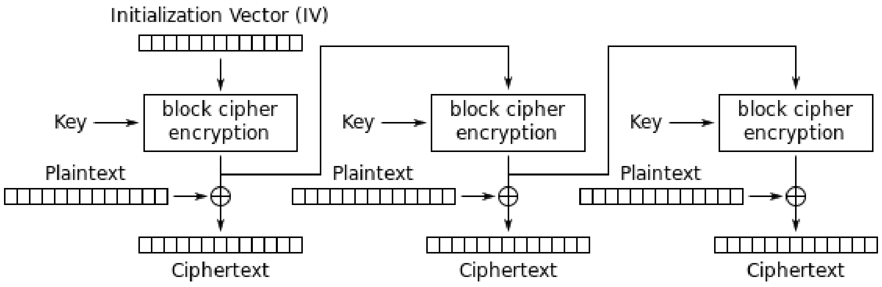


Limitations of CFB

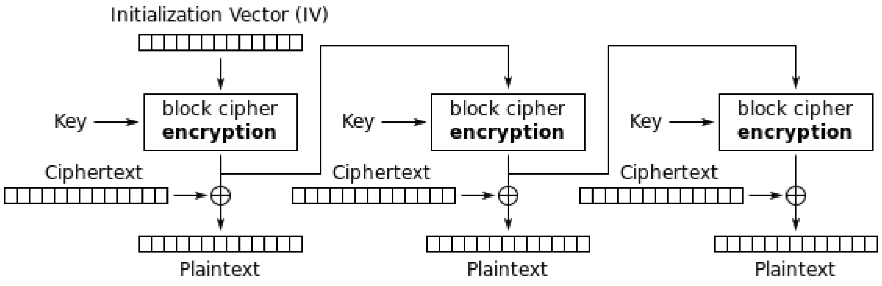
- Appropriate when data arrives in bits/bytes
- Most common stream mode
- Limitation is need to stall while doing block encryption after every n-bits
- Note that the block cipher is used in **encryption** mode at **both** ends
- Errors propagate for several blocks after the error

Output Feedback Mode (OFB)

- Message is treated as a stream of bits
- Output of cipher is added to message
- Output is then fed back (hence name)
- Feedback is independent of message
- Can be computed in advance
  - $C_i = P_i \text{ XOR } O_i$
  - $O_i = DES_{K1}(O_{i-1}) \quad i>1$
  - $O_1 = DES(Nonce)$
- Uses: stream encryption on noisy channels



Output Feedback (OFB) mode encryption



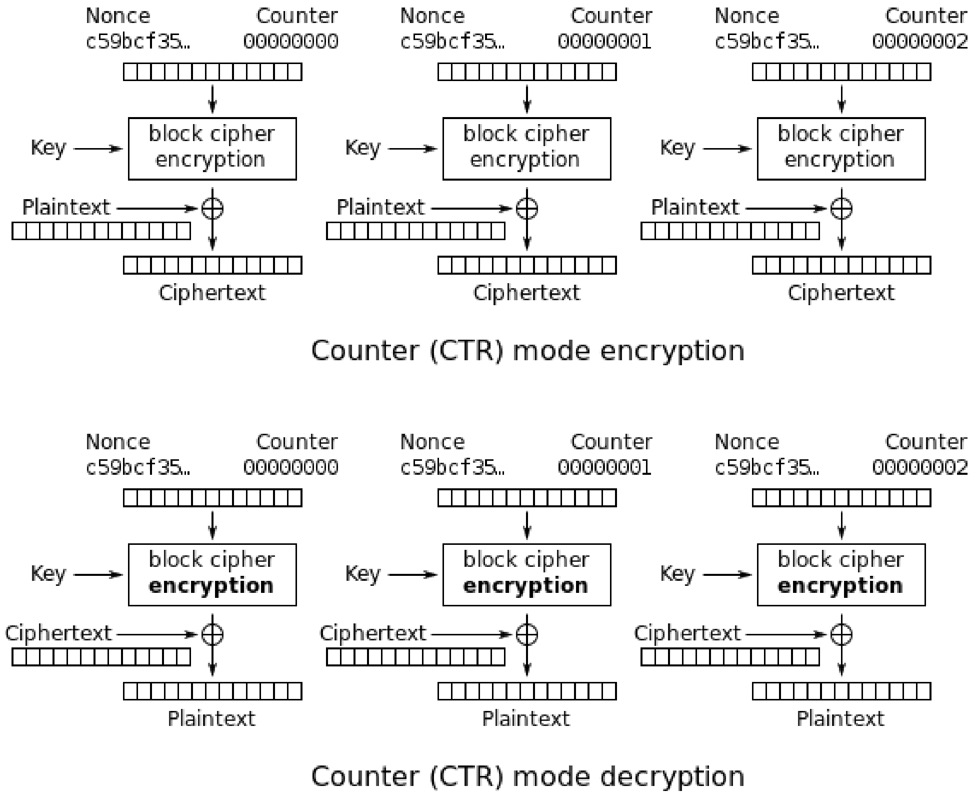
Output Feedback (OFB) mode decryption

Limitations of OFB

- Bit errors do not propagate
- More vulnerable to message stream modification
- Must **never** reuse the same sequence (key+IV)
- Sender & receiver must remain in sync
- Originally specified with m-bit feedback
- Subsequent research has shown that only **full block feedback** (i.e. CFB-64 or CFB-128) should ever be used

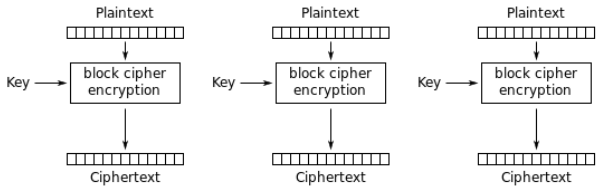
Counter Mode (CTR)

- Relatively "new" mode, though proposed early on
- Similar to OFB but encrypts counter value rather than any feedback value
- Must have a different key & counter value for every plaintext block (never reused)
  - $C_i = P_i \text{ XOR } O_i$
  - $O_i = DES_{K1}(i)$
- Uses: high-speed network encryptions

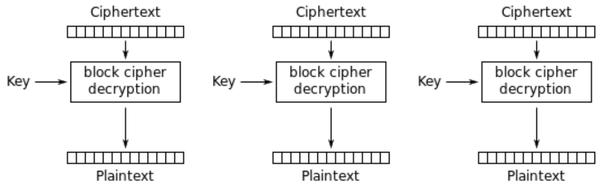


Limitations of CTR

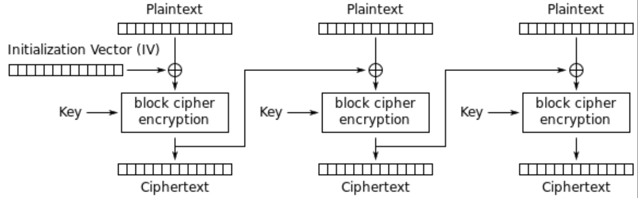
- Efficiency
  - can do parallel encryptions in h/w or s/w
  - can preprocess in advance of need
  - good for bursty high speed links
- Random access to encrypted data blocks
- Provable security (good as other modes)
- But must ensure never reuse key/counter values, otherwise could break (cf. OFB)



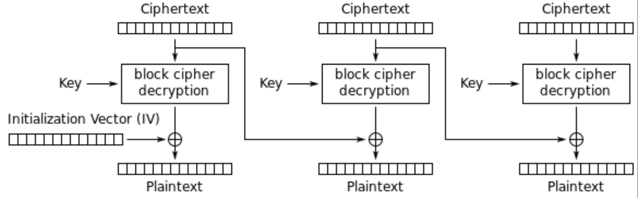
Electronic Codebook (ECB) mode encryption



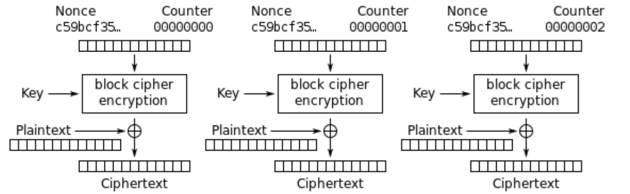
Electronic Codebook (ECB) mode decryption



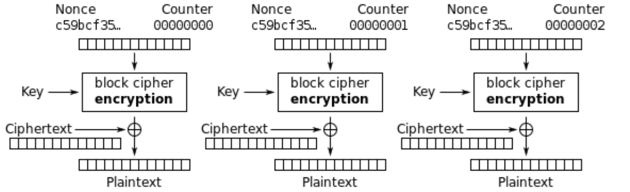
Cipher Block Chaining (CBC) mode encryption



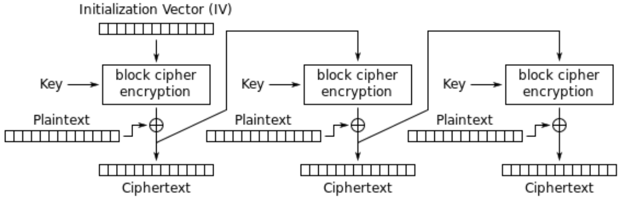
Cipher Block Chaining (CBC) mode decryption



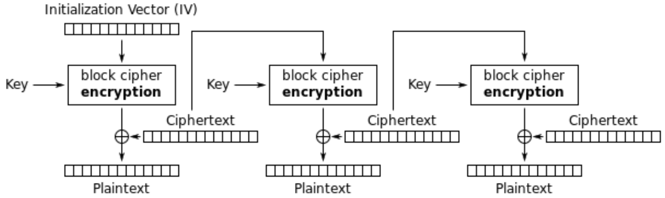
Counter (CTR) mode encryption



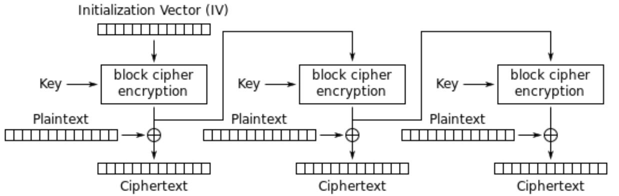
Counter (CTR) mode decryption



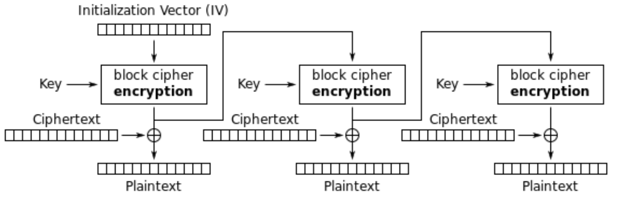
Cipher Feedback (CFB) mode encryption



Cipher Feedback (CFB) mode decryption



Output Feedback (OFB) mode encryption



Output Feedback (OFB) mode decryption

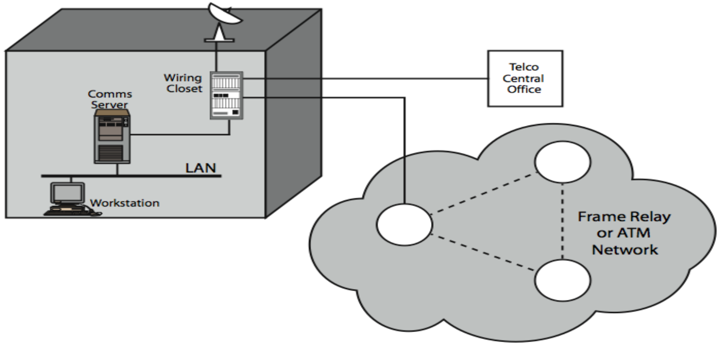
Take note of differences in:

- Propagation (and scope) of error in plaintext
- Propagation (and scope) of error in ciphertext
- Parallel execution
- Cryptanalysis
- Stream encryption
- Random Access

Mode	Description	Typical Application
Electronic Codebook (ECB)	Each block of 64 plaintext bits is encoded independently using the same key.	• Secure transmission of single values (e.g., an encryption key)
Cipher Block Chaining (CBC)	The input to the encryption algorithm is the XOR of the next 64 bits of plaintext and the preceding 64 bits of ciphertext.	• General-purpose block-oriented transmission • Authentication
Cipher Feedback (CFB)	Input is processed *s* bits at a time. Preceding ciphertext is used as input to the encryption algorithm to produce pseudorandom output, which is XORed with plaintext to produce next unit of ciphertext.	• General-purpose stream-oriented transmission • Authentication
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption algorithm is the preceding encryption output, and full blocks are used.	• Stream-oriented transmission over noisy channel (e.g., satellite communication)
Counter (CTR)	Each block of plaintext is XORed with an encrypted counter. The counter is incremented for each subsequent block.	• General-purpose block-oriented transmission • Useful for high-speed requirements

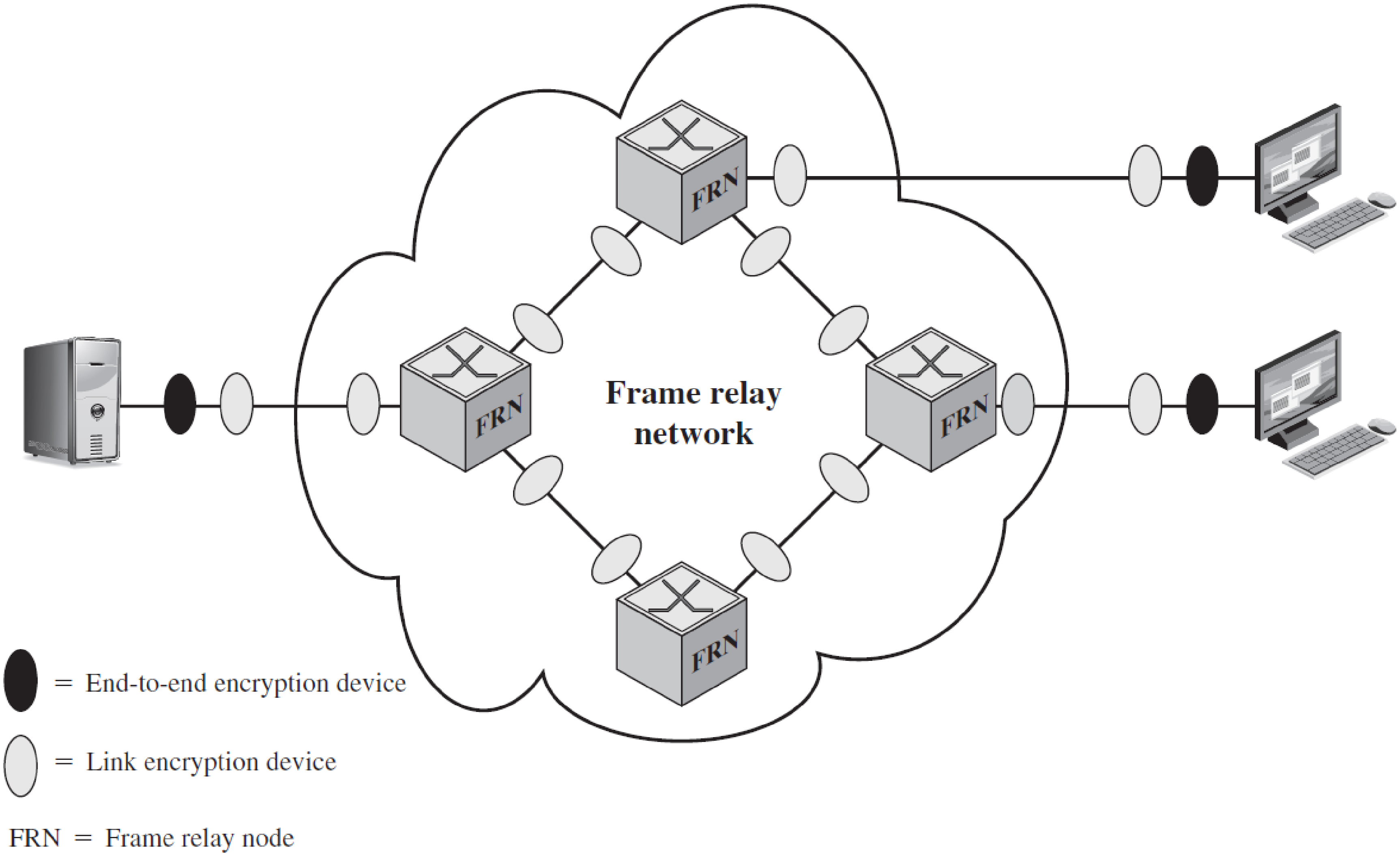
Confidentiality using Symmetric Encryption

- Have two major placement alternatives
- **Link encryption**
  - encryption occurs independently on every link
  - implies must decrypt traffic between links
  - requires many devices, but paired keys
- **End-to-end encryption**
  - encryption occurs between original source and final destination
  - need devices at each end with shared keys



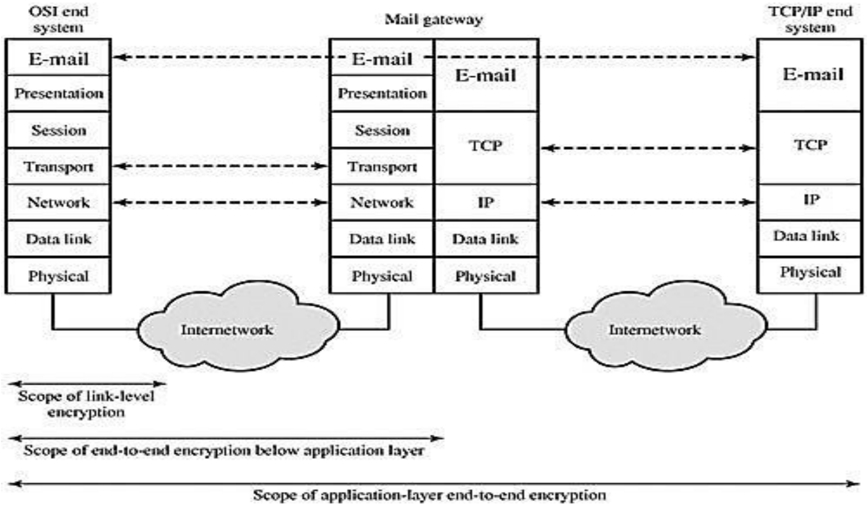
Placement of Encryption

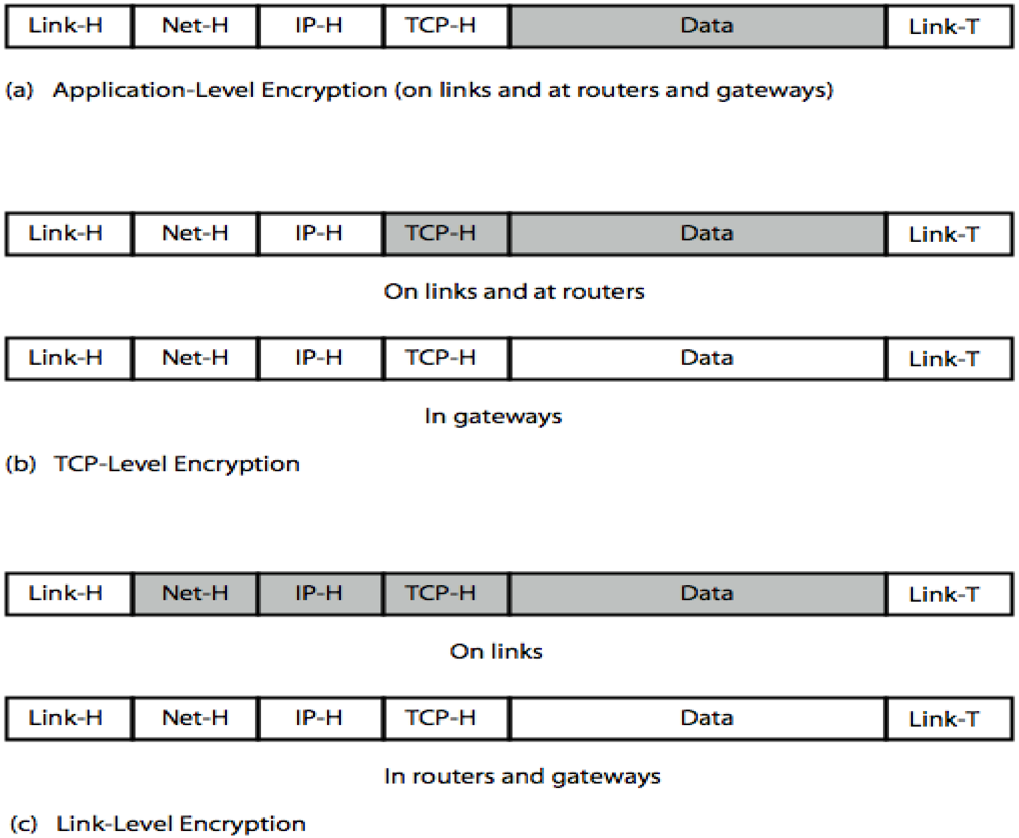
- When using end-to-end encryption we must leave the headers in the clear
  - so network can correctly route information
- Hence although contents are protected, traffic pattern flows are not
- Ideally want both at once
  - end-to-end protects data contents over entire path and provides authentication
  - link protects traffic flows from monitoring


Frame relay
network
FRN
FRN
FRN
FRN
= End-to-end encryption device
= Link encryption device
FRN = Frame relay node

Placement of Encryption

- Can place encryption function at various layers in OSI Reference Model
  - link encryption occurs at layers 1 or 2
  - end-to-end can occur at layers 3, 4, 6, 7
  - as we move higher, less information is encrypted but it is more secure though more complex with more entities and keys





Shading indicates encryption.

TCP-H	=	TCP header
IP-H	=	IP header
Net-H	=	Network-level header (e.g., X.25 packet header, LLC header)
Link-H	=	Data link control protocol header
Link-T	=	Data link control protocol trailer