Block Cipher Operation

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu

Audio/Video recordings of this lecture are available at:

http://www.cse.wustl.edu/~jain/cse571-14/

1. Double DES, Triple DES, DES-X
2. Encryption Modes for long messages:
   1. Electronic Code Book (ECB)
   2. Cipher Block Chaining (CBC)
   3. Cipher Feedback (CFB)
   4. Output Feedback (OFB)
   5. Counter (CTR) Mode
   6. XTS-AES Mode for Block-oriented Storage Devices

These slides are based partly on Lawrie Brown's slides supplied with William Stallings's book "Cryptography and Network Security: Principles and Practice," 6th Ed, 2013.

Double-DES

- $C = E_{K2}(E_{K1}(P))$
- **Meet-in-the-middle attack**
  - Developed by Diffie and Hellman in 1977
  - Can be used to attack any composition of 2 functions

$$X = E_{K1}(P) = D_{K2}(C)$$

  - Attack by encrypting P with all 2^{56} keys and storing
  - Then decrypt C with keys and match X value
  - Verify with one more pair
  - Takes max of $O(2^{56})$ steps ⇒ Total 2^{57} operations
- Only twice as secure as single DES

Triple-DES

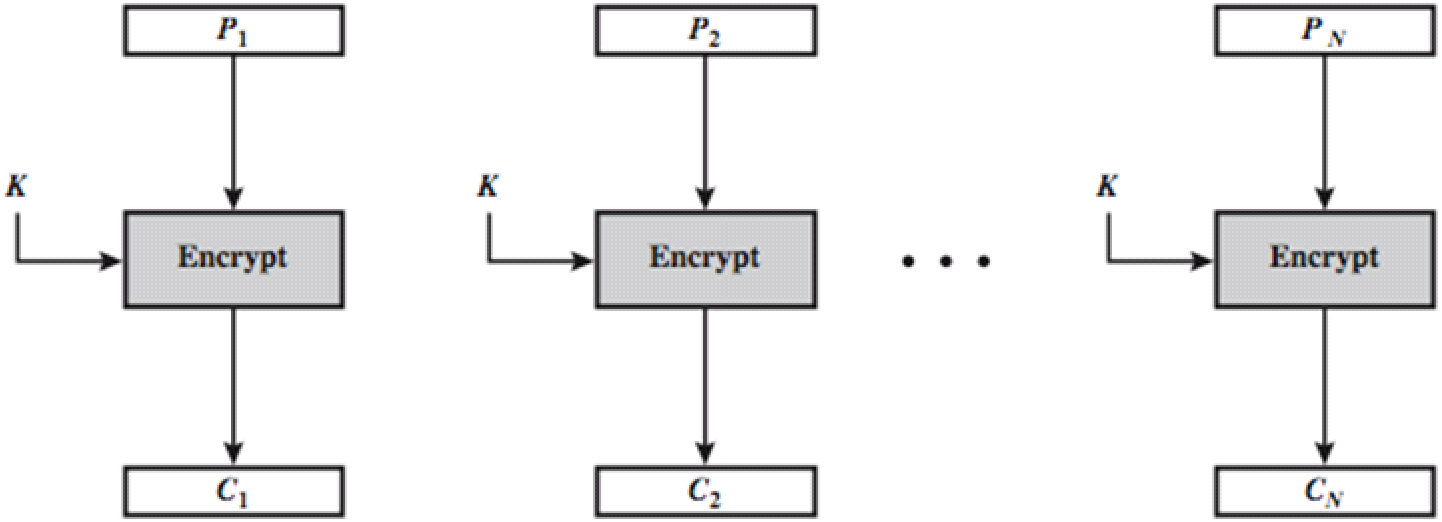
- Use DES 3 times: $C = E_{K3}(D_{K2}(E_{K1}(P)))$
- E-D-E provides the same level of security as E-E-E
- E-D-E sequence is used for compatibility with legacy
  - $K1=K2=K3 \Rightarrow$ DES
- PGP and S/MIME use this 3 key version
- Provides 112 bits of security
- Two keys with E-D-E sequence
  - $C = E_{K1}(D_{K2}(E_{K1}(P)))$
  - Standardized in ANSI X9.17 & ISO8732
  - No current known practical attacks
  - Several proposed impractical attacks might become basis of future attacks

Electronic Codebook (ECB) Mode

- How to encode multiple blocks of a long message?
- Each block is encoded independently of the others

$$C_i = E_K(P_i)$$

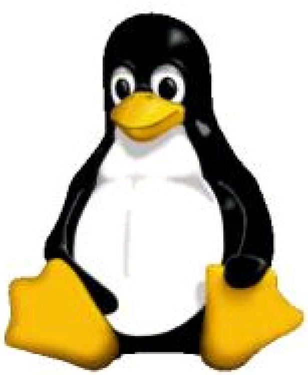
- Each block is substituted like a codebook, hence name.



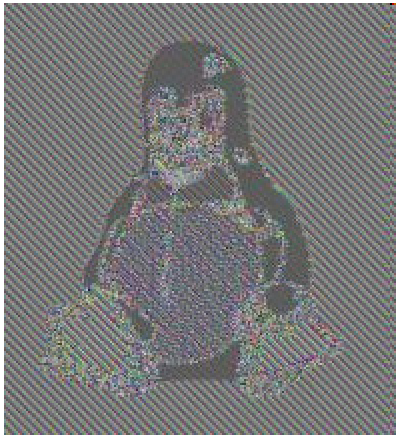
Ref: http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation

ECB Limitations

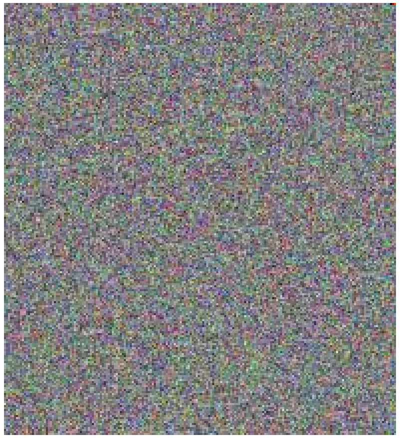
- Using the same key on multiple blocks makes it easier to break
- Identical Plaintext Identical Ciphertext
  Does not change pattern:



Original



ECB



Better

- NIST SP 800-38A defines 5 modes **that** can be used with any block cipher

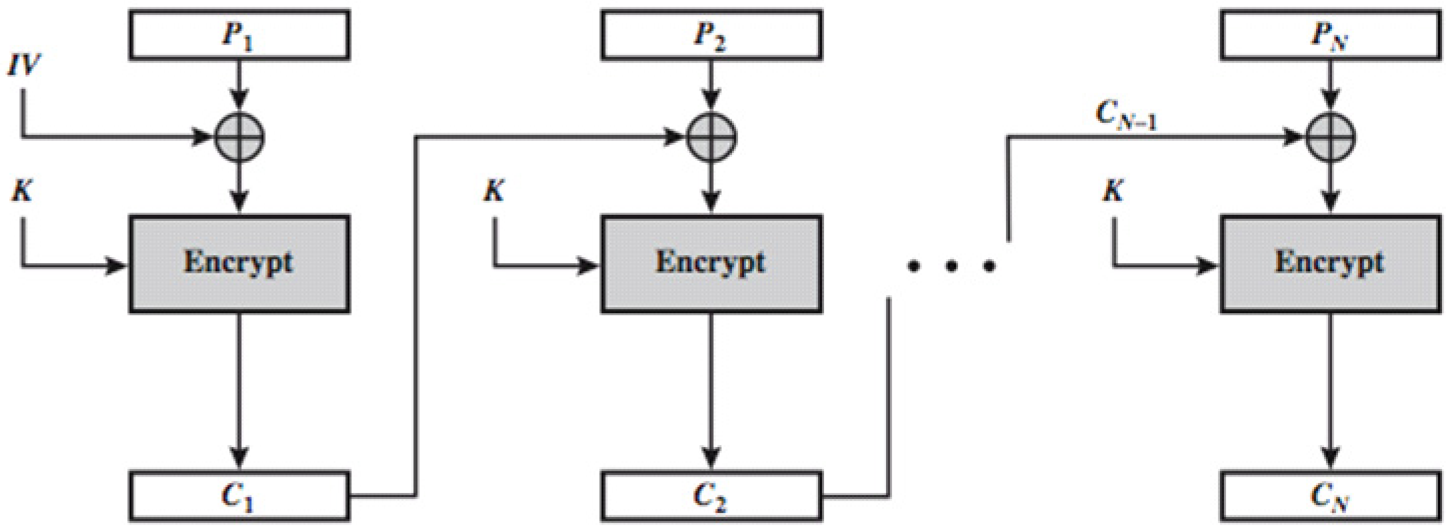
Ref: http://en.wikipedia.org/wiki/Modes_of_operation

Cipher Block Chaining (CBC)

- Add random numbers before encrypting
- Previous cipher blocks is chained with current plaintext block
- Use an Initial Vector (IV) to start process

$$C_i = E_K(P_i \text{ XOR } C_{i-1})$$

$$C_0 = IV$$



Advantages and Limitations of CBC

- Any change to a block affects all following ciphertext blocks
- Need **Initialization Vector** (IV)
  - Must be known to sender & receiver
  - If sent in clear, attacker can change bits of first block, and change IV to compensate
  - Hence IV must either be a fixed value, e.g., in Electronic Funds Transfers at Point of Sale (EFTPOS)
  - Or must be sent encrypted in ECB mode before rest of message
- Sequential implementation. Cannot be parallelized.

Message Padding

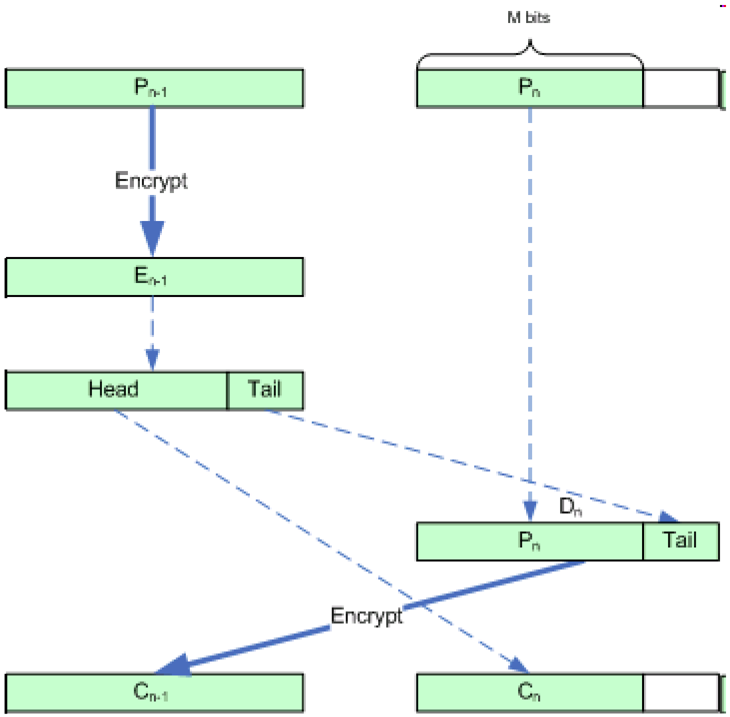
- Last block may be shorter than others ⇒ Pad
- Pad with count of pad size [ANSI X.923]
  1. E.g., [b1 b2 b3 0 0 0 0 5] = 3 data, 5 pad w 1 count byte

1. A 1 bit followed by 0 bits [ISO/IEC 9797-1]
2. Any known byte value followed by zeros, e.g., 80-00…
3. Random data followed by count [ISO 10126]
   1. E.g., [b1 b2 b3 84 67 87 56 05]
4. Each byte indicates the number of padded bytes [PKCS]
   1. E.g., [b1 b2 b3 05 05 05 05 05]
5. **Self-Describing Padding** [RFC1570]
   - Each pad octet contains its index starting with 1
   - E.g., [b1 b2 b3 1 2 3 4 5]

Ref: http://en.wikipedia.org/wiki/Padding_%28cryptography%29

Cipher Text Stealing (CTS)

- Alternative to padding
- Last 2 blocks are specially coded
- Tail bits of (n-1)st encoded block are added to nth block and order of transmission of the two blocks is interchanged.



Stream Modes of Operation

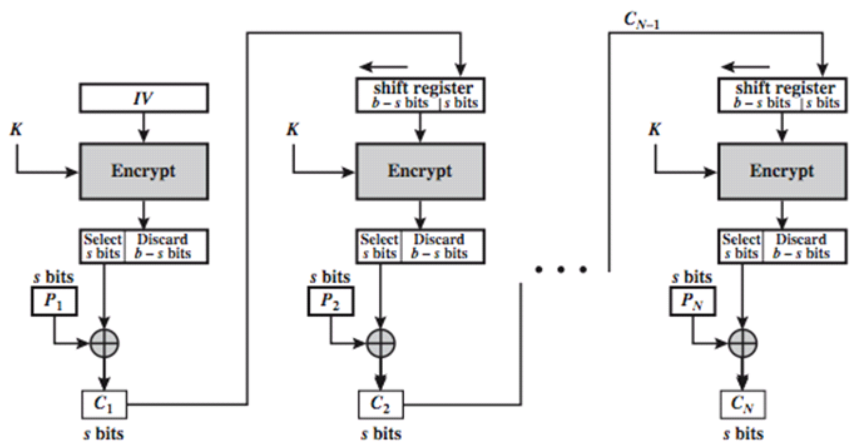
- Use block cipher as some form of **pseudo-random number** generator
- The random number bits are then XOR'ed with the message (as in stream cipher)
- Convert block cipher into stream cipher
  1. Cipher feedback (CFB) mode
  2. Output feedback (OFB) mode
  3. Counter (CTR) mode

Cipher Feedback (CFB)

- Message is added to the output of the block cipher
- Result is feed back for next stage (hence name)
- Standard allows any number of bit (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)

$$C_i = P_i \text{ XOR } E_K(C_{i-1})$$
$$C_{-1} = IV$$

- Errors propagate for several blocks after the error



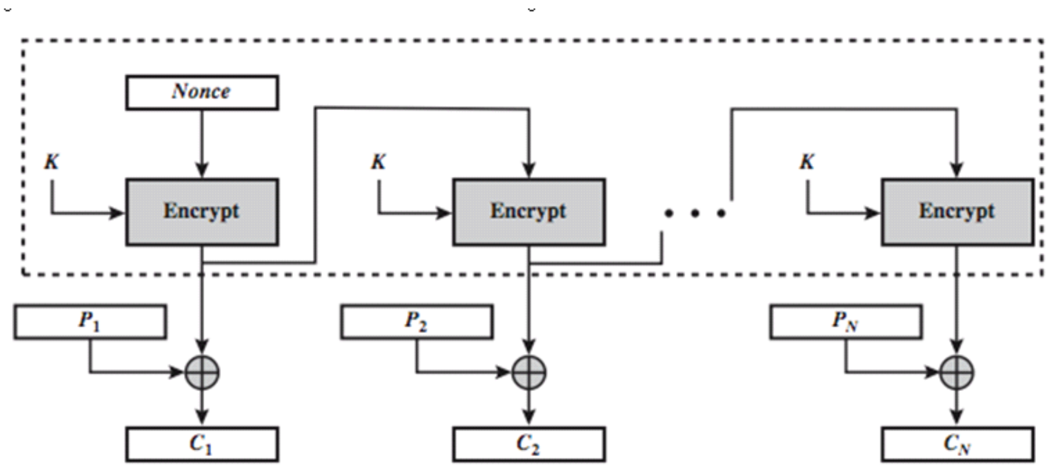
Output Feedback (OFB)

- Output of the cipher is feed back (hence name)
- Feedback is independent of message
- Can be computed in advance

$$O_i = E_K(O_{i-1})$$

$$C_i = P_i \text{ XOR } O_i$$

$$O_{-1} = IV$$



Advantages and Limitations of OFB

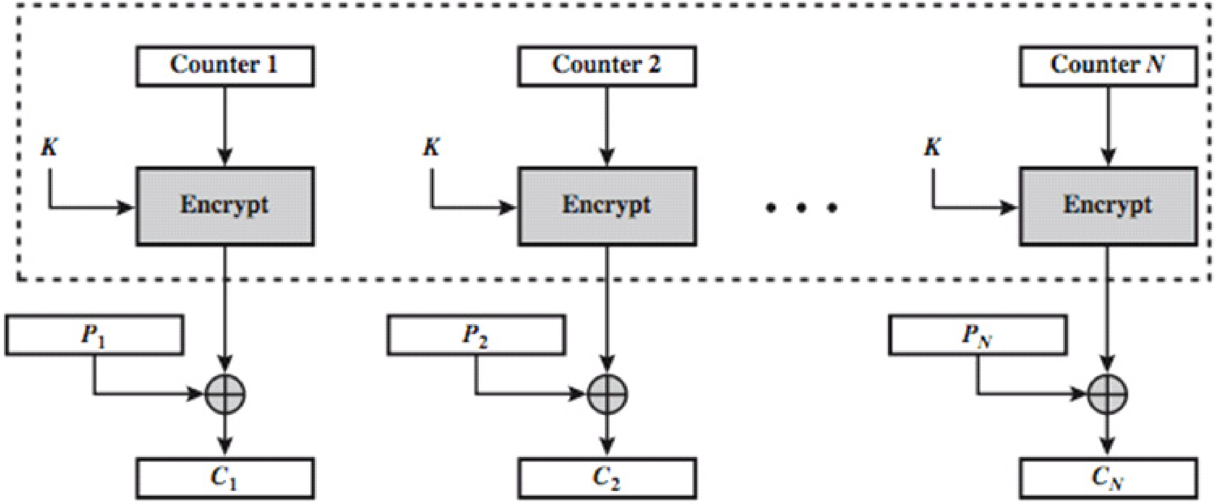
- Needs an IV which is unique for each use
  - if ever reuse attacker can recover outputs
- Bit errors do not propagate
- More vulnerable to message stream modification
- Sender & receiver must remain in sync
- Only use with full block feedback
  - Subsequent research has shown that only **full block feedback** (i.e., CFB-64 or CFB-128) should ever be used

Counter (CTR)

- Encrypt counter value rather than any feedback value
- Different key & counter value for every plaintext block (never reused)

$$O_i = E_K(i)$$

$$C_i = P_i \text{ XOR } O_i$$



Advantages and Limitations of CTR

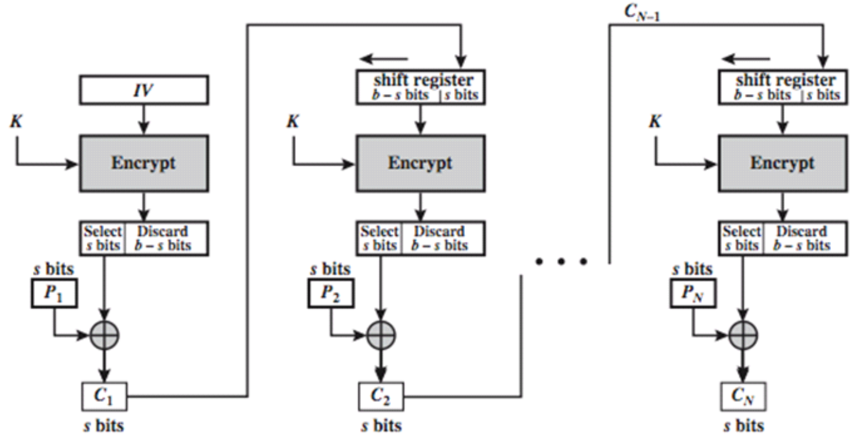
- Efficiency
  - Can do parallel encryptions in h/w or s/w
  - Can preprocess in advance of need
  - Good for bursty high speed links
- Random access to encrypted data blocks
- Provable security (good as other modes)
- But must never reuse key/counter values, otherwise could break

Storage Encryption

- File encryption:
  - Different keys for different files
  - May not protect metadata, e.g., filename, creation date,
  - Individual files can be backed up
  - Encrypting File System (EFS) in NTFS provides this svc
- Disk encryption:
  - Single key for whole disk or separate keys for each partition
  - Master boot record (MBR) may or may not be encrypted
  - Boot partition may or may not be encrypted.
  - Operating system stores the key in the memory
    Can be read by an attacker by cold boot
- Trusted Platform Module (TPM): A secure coprocessor chip on the motherboard that can authenticate a device
  $\Rightarrow$ Disk can be read only on that system.
  Recovery is possible with a decryption password or token

Storage Encryption (Cont)

- If IV is predictable, CBC is not usable in storage because the plain text is chosen by the writer
- Ciphertext is easily available to other users of the same disk
- Two messages with the first blocks=$b \oplus IV_1$ and $b \oplus IV_2$ will both encrypt to the same ciphertext
- Need to be able to read/write blocks without reading/writing other blocks



CBC

XTS-AES Mode

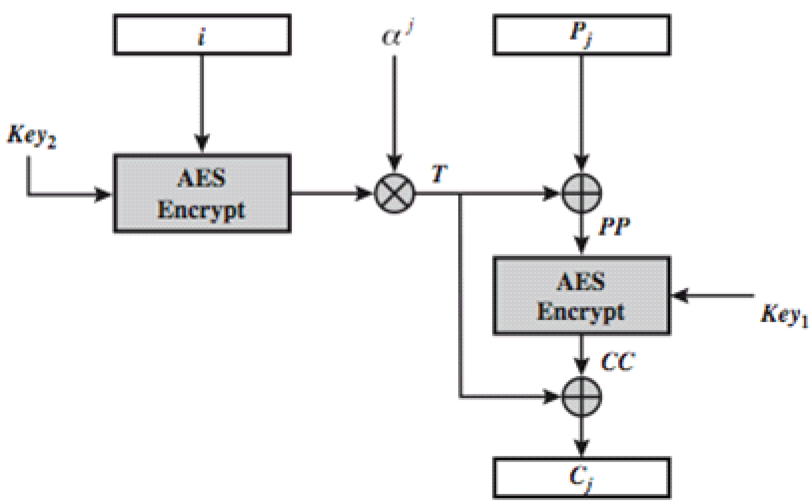
- XTS = **X**EX-based **T**weaked Codebook mode with Ciphertext **S**tealing (XEX = Xor-Encrypt-xor)
- Creates a unique IV for each block using AES and 2 keys

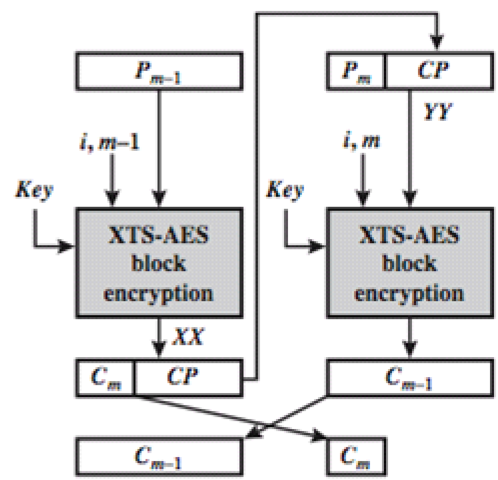
  $T_j = E_{K2}(i) \otimes \alpha^j$ Size of K2 = size of block

  $C_j = E_{K1}(P_j \oplus T_j) \oplus T_j$ *K1 256 bit for AES-256*

  where *i* is logical sector # & *j* is block # (sector = n blocks)

  α = primitive element in $GF(2^{128})$ defined by polynomial x





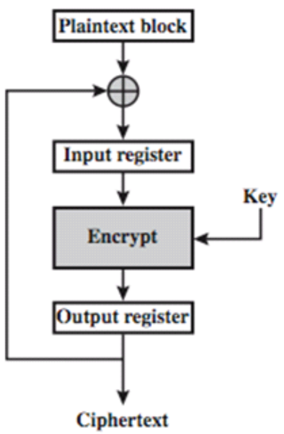
Advantages and Limitations of XTS-AES

- Multiplication is modulo $x^{128}+x^7+x^2+x+1$ in GF(2^{128})
- Efficiency
  - Can do parallel encryptions in h/w or s/w
  - Random access to encrypted data blocks
- Has both nonce & counter
- Defined in IEEE Std 1619-2007 for block oriented storage use
- Implemented in numerous packages and operating systems including TrueCrypt, FreeBSD, and OpenBSD softraid disk encryption software (also native in Mac OSX Lion's FileVault), in hardware-based media encryption devices by the SPYRUS Hydra PC Digital Attaché and the Kingston DataTraveler 5000.

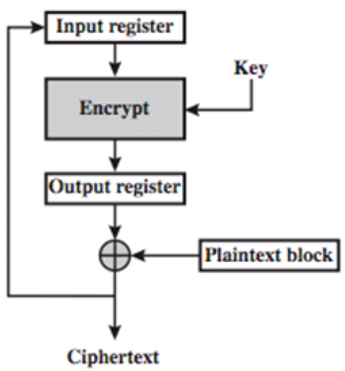
Ref: http://en.wikipedia.org/wiki/Disk_encryption_theory

Summary

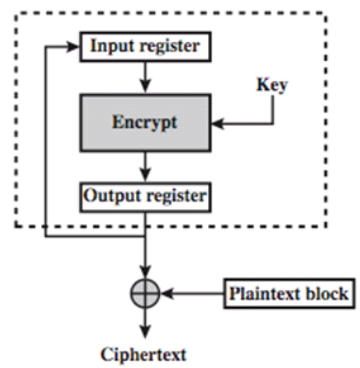
- 3DES generally uses E-D-E with 2 keys ⇒112b protection
- ECB: Same ciphertext for the same plaintext ⇒ Easier to break



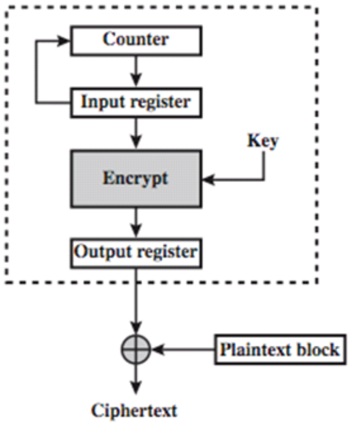
(a) Cipher block chaining (CBC) mode



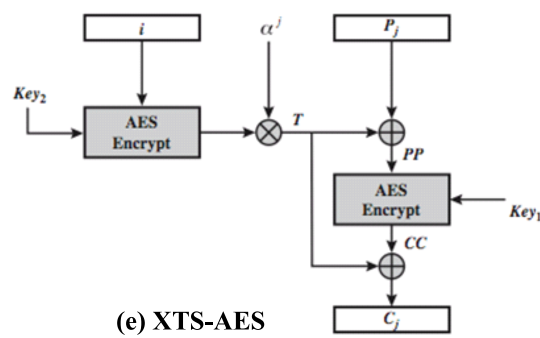
(b) Cipher feedback (CFB) mode



(c) Output feedback (OFB) mode



(d) Counter (CTR) mode



(e) XTS-AES

Homework 6

For each of the modes ECB, CBC and CTR:

a. Identify whether decrypted plaintext block P_3 will be corrupted if there is an error in block C_1 of the transmitted cipher text.

b. Assuming that the ciphertext contains N blocks, and that there was a bit error in the source version of P_1, identify through how many ciphertext blocks this error is propagated.

Lab Homework 6

- This homework requires two computers with SSH and telnet client and servers installed.
- You can download the following open source SSH and telnet clients:
  - http://www.freesshd.com/
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/
- These utilities are installed on CSE571XPC and CSE571XPS in our lab.
- Start wireshark on the client machine (CSE571XPS).
- telnet (Putty) to the server (CSE571XPC) and login with your username and password. Logout.
- Use "follow the TCP stream option" (right click on the packet) to see your username and password on the screen. Capture the screen and circle your password.
- ssh (Putty) to the server (CSE571XPC) and login with your username and password. Logout.
- Stop wireshark and read the trace. Capture the screen. Circle the password characters. Note the difference in the two logins?