Modes of Operation
Encrypting a Large Message

• How do you encrypt a message larger than 64 bits?

• Modes of Block Cipher Operations:
  – Electronic Code Book (ECB)
  – Cipher Block Chaining (CBC)
  – K-Bit Cipher Feedback Mode (CFB)
  – K-Bit Output Feedback Mode (OFB)
  – Counter Mode (CTR)
Electronic Code Book (ECB)

Also see Fig. 4-1
ECB Problem #1

\[(M_1 == M_3) \Rightarrow (C_1 == C_3)\]
ECB Problem #2

• Without additional integrity protection
  – cipher block substitution and rearrangement attacks
  – fabrication of specific information

• Lack the basic protection against integrity attacks on the ciphertext at message level
Cipher Block Chaining (CBC)

(M₁ == M₃) very unlikely leads to (C₁ == C₃)

Also see Fig. 4-5
CBC Decryption

Also see Fig. 4-6
CBC Properties

• Chaining dependency
  – Each ciphertext block depends on all preceding plaintext blocks

• Error propagation
  – Each error $c_j$ affects the decipherment of $c_j$ and $c_{j+1}$

• Error recovery
  – An error in $c_j$ does not propagate beyond $c_{j+1}$

• Threat

• Modifying Ciphertext Blocks
  – Changing $c_n$ has a predictable effect on $m_{n+1}$
  – Most likely, changing $c_n$ will garble $m_n$ to some random 64-bit value
**k bit Output Feedback Mode (k-bit OFB)**

- It is a **stream cipher**: generate a **one-time pad** and apply it to a stream of plaintext with $\oplus$
OFB Properties

• Advantage
  – One-time pad can be generated in advance: extremely fast encryption when the message comes
  – If some bits of the ciphertext are garbled, only those bits of plaintext get garbled
  – Arbitrary sized chunk

• Disadvantage
  – If the plaintext and ciphertext are known, the adversary can modify the plaintext into anything. Generate whatever message he wants to transmit
$k$ bit Cipher Feedback Mode ($k$-bit CFB)

- Also see Fig 4-9 of the textbook
CFB Properties

• Advantage compared with CBC/OFB
  – With 8-bit CFB, if a byte is lost, one byte of plaintext will be lost and the next 8 bytes will be garbled. After that, the plaintext will decrypt properly.
  – If a byte is added to the ciphertext, a byte of garbage will be added, and the following 8 bytes will be garbled, the rest will be ok.

• Disadvantage
  – Random stream can no longer be computed in advance
Counter Mode (CTR)

\[ K \rightarrow E \rightarrow \oplus \rightarrow c_1 \]
\[ m_1 \rightarrow \oplus \rightarrow c_1 \]

\[ K \rightarrow E \rightarrow \oplus \rightarrow c_2 \]
\[ m_2 \rightarrow \oplus \rightarrow c_2 \]

\[ K \rightarrow E \rightarrow \oplus \rightarrow c_3 \]
\[ m_3 \rightarrow \oplus \rightarrow c_3 \]
CTR Properties

• Advantages
  – Like OFB, cryptography can be pre-computed, encryption is simple
  – Like CBC, decrypt the message starting at any point

• Disadvantages
  – Attacker could get the $\oplus$ of two plaintext blocks if different data is encrypted with the same key and IV
Generating MACs

- MAC – Message Authentication Code
  - Also known as cryptographic checksum, Message Integrity Code (MIC)
  - Assumption: sender and receiver share a common secret key
  - Only send last block of CBC (CBC residue) along with the plaintext message
  - The modification of the message will make the CBC residue incorrect – insures integrity
Ensuring Privacy and Integrity Together

Scheme 1??

Scheme 2??
Ensuring Privacy and Integrity Together

Scheme 3
Ensuring Privacy and Integrity Together

- CBC with a Weak Cryptographic Checksum
- Use separate (but related) secret keys for encryption and MAC (two encryption passes)
- CBC(message | hash)
- ...
3DES

• Major limitations of DES
  – Key length is too short (56 bits)
• Multiple encryption to compensate for the short basic DES key
  – Make it more secure
• Standard practice: $E(K_3, D(K_2, E(K_1, P)))$
• $K_1 = K_3$ results in an equivalent 112-bit DES that provides an sufficient key space
• Distinct $K_1, K_2, K_3$ results even stronger 168-bit DES
• EDE: can run as a single DES with $K_1 = K_2$
Standard Method for EDE

- Two keys: K1 and K2

Many Variations!!
3DES

• How many encryptions?
  – We do not want to do any more encryptions than are necessary

• Encrypting Twice with the Same Key
  – Not to be much more secure than single encryption with K

• Encrypting Twice with Two Keys
  – Meet-in-the-middle-attack

• Triple Encryption with only Two Keys
  – Using $K_1$ twice is sufficiently secure
  – $K_2$ used in decrypt mode
Meet-in-the-middle Attack

Encryption

\[ P \xrightarrow{E} X \xrightarrow{E} C \]

Decryption

\[ P \xleftarrow{D} X \xleftarrow{D} C \]

Observation:

\[ X = E_{K_1}(P) = D_{K_2}(C) \]

- For a few \(<plaintext, ciphertext> <p, c>\)
  - Encrypt \(m\) for all \(2^{56}\) values of \(K_1\)
  - Store the results in a table sorted by the value of \(X\)
  - Decrypt \(C\) for all \(2^{56}\) values for \(K_2\)
  - Store the results in a table sorted by the value of \(X\)
  - Search through the sorted lists to find matching entries.
Meet-in-the-middle Attack

• Analysis
  – With one pair \(<p_1, c_1>\), the number of keys that can survive the test is \(2^{48}\)
  – For each pair of keys \((K_1, K_2)\), the probability that it can find a non-empty entry in the table is \(2^{-8}\)
  – With another pair \(<p_2, c_2>\), the number of keys that can survive both test is \(2^{-16}\)
  – The probability that the correct keys is determined is \(2^{-16}\)
CBC Outside vs. Inside

• CBC on the outside
  – No change in the property - It is possible to make a predictable change to plaintext block n

• CBC on the inside
  – Any change to ciphertext block n garbles all plaintext blocks from n to the end of the message – more secure

• Self-synchronization
3DES Is Not Ideal...

- Efficiency demands schemes with longer keys to begin with!
- 3DES runs one third as fast as DES on the same platform
- New candidates are numerous - RC5, IDEA, two-fish, CAST, etc