Block Ciphers
Lucifer, DES, RC5, AES

CS 470
Introduction to Applied Cryptography
Ali Aydın Selçuk
Block Ciphers & S-P Networks

- Block Ciphers: Substitution ciphers with large block size (≥ 64 bits)
- How to define a good substitution for such large blocks?
- “SP Networks” (Shannon, 1949)
  - small, carefully designed substitution boxes (“confusion”)
  - their output mixed by a permutation box (“diffusion”)
  - iterated a certain number of times
Lucifer

- Early 1970s: First serious needs for civilian encryption (in electronic banking)
- IBM’s response: Lucifer, an iterated SP cipher
- Lucifer (v0):
  - Two fixed, 4x4 s-boxes, $S_0$ & $S_1$
  - A fixed permutation $P$
  - Key bits determine which s-box is to be used at each position
  - $8 \times 64/4 = 128$ key bits (for 64-bit block, 8 rounds)
Feistel Ciphers

- A straightforward SP cipher needs twice the hardware: one for encryption (S, P), one for decryption (S⁻¹, P⁻¹).
- Feistel’s solution:

\[ E_K(x) = (L, R) \]

where the \( f \) function is SP:

\[ f(x, k_i) \]

- Lucifer v1: Feistel SP cipher; 64-bit block, 128-bit key, 16 rounds.
Data Encryption Standard (DES)

• Need for a standardized cipher to protect computer and communications data

• NBS’ request for proposals (1973)

• IBM’s submission Lucifer is adopted after a revision by NSA, reducing the key size to 56 bits.
The DES Contraversy

• Design process was not made public. Any hidden trapdoors in the s-boxes?
  (Now, with the design criteria better understood, this speculation is mostly over.)

• 56-bit key length is too short. So that NSA can break it?
Strengthening DES

• Multiple DES encryption: 3DES
  \[ E_{K_3}(D_{K_2}(E_{K_1}(x))) \]
  – Why not 2DES?
  – Why “D”?

• DES-X (Rivest, 1995)
  \[ E_K(x \oplus K_1) \oplus K_2 \]
  – overhead cost minimal
  – construction is provably secure (Rogaway & Killian)
After DES

• DES was designed mainly for h/w; it was slow in s/w. It was also suspect, due to the secret design process.

• By the late ’80s, need for an independently developed, fast-in-s/w cipher was clear.

• Several prominent examples emerged in this era: IDEA, Blowfish, RC5…
**RC5**
(Rivest, 1994)

- Extremely simple & flexible
- Variable block size \(w\), key size \(b\), no. of rounds \(r\); specified as RC5-\(w/r/b\).
- Encryption algorithm:
  \[
  \begin{align*}
  L_1 &= L_0 + K_0 \\
  R_1 &= R_0 + K_1 \\
  \text{for } i &= 2 \text{ to } 2r+1 \text{ do} \\
  L_i &= R_{i-1} \\
  R_i &= ((L_{i-1} \oplus R_{i-1}) \ll R_{i-1}) + K_i
  \end{align*}
  \]
- For 64-bit block size \((w=32)\), 24 rounds \((r=12)\) is secure
Advanced Encryption Standard (AES)

Successful public design process:

• NIST’s request for proposals for a new enc. standard to replace DES (1997)
• 15 submissions (1998)
• 5 finalists (1999)
  Mars (IBM)
  RC6 (RSA)
  Twofish (Schneier et al.)
  Serpent (Anderson et al.)
  Rijndael (Daemen & Rijmen)
• Winner: Rijndael (2000)
AES (Rijndael)

- An SP cipher with one algebraically designed s-box (optimal against linear & diff. cryptanalysis)
- 128-bit block size
  128, 192, or 256-bit key.
- 10-14 rounds of:
  SubBytes, ShiftRows, MixColumns, AddRoundKey
- Decryption is similar to encryption (by design)
- Very good security; also very high performance in s/w, h/w, and restricted devices (smart cards)
AES: SubBytes

- There is a fixed (8x8)-bit s-box, with optimal algebraic properties.
- Each byte in the state (i.e., block) is replaced by its entry in the S-box; $b_{ij} = S(a_{ij})$. 
AES: ShiftRows

- Bytes in each row of the state are shifted cyclically to the left by 0, 1, 2, 3 positions, respectively.
AES: MixColumns

- Each column of the state is multiplied by a fixed matrix $M$: $b_i = M a_i$
AES: AddRoundKey

- Each byte of the state is combined with a byte of the round subkey by XOR.
AES: Encryption Algorithm

1. Key expansion
2. Initial round
   - AddRoundKey
3. Rounds \((N_r - 1)\) times, where \(N_r\) is 10, 12, or 14 for 128, 192, 256 bit keys, respectively.
   - SubBytes
   - ShiftRows
   - MixColumns
   - AddRoundKey
4. Final round
   - SubBytes
   - ShiftRows
   - AddRoundKey