CS 494/594
Computer and Network Security

Dr. Jinyuan (Stella) Sun
Dept. of Electrical Engineering and Computer Science
University of Tennessee
Fall 2010
Secret Key Cryptography

- Block cipher
- DES
- 3DES
- AES
Generic Block Encryption

• Block cipher: encryption/decryption in which a fixed-length block of plaintext is mapped to a ciphertext block of equal length
  • Random mapping: when any one bit of plaintext changes, every bit in ciphertext has 50% chance to change
  • Substitution: space complexity $O(k 2^k)$ for $k$-bit blocks
  • Permutation: space complexity $O(k \log k)$ for $k$-bit blocks
  • Fixed key length: can be the same length as the block or different
Example of Block Encryption

Figure 3-1:
Diffusion and Confusion

• Shannon’s proposal in 1949: develop a product cipher that alternates confusion and diffusion functions
• Diffusion: the statistical structure of the plaintext is dissipated into long-range statistics of the ciphertext by having each plaintext digit affect the value of many ciphertext digits
• Confusion: make the relationship between the statistics of the ciphertext and the value of the encryption key as complex as possible to thwart attempts to discover the key
• They capture the essence of the desired attributes of a block cipher
Data Encryption Standard (DES)

- Designed by IBM and published by NIST in 1977
- 64-bit input block → 64-bit output block with 56-bit key
- Not secure anymore: key size must be increased by 1 bit every 2 years
- 3DES: 112-bit key
DES Overview

Figure 3-2: Basic Structure of DES
Permutations of The Data

- Do not enhance security

<table>
<thead>
<tr>
<th>Initial Permutation (IP)</th>
<th>Final Permutation (IP')</th>
</tr>
</thead>
<tbody>
<tr>
<td>58 50 42 34 26 18 10 2</td>
<td>40 8 48 16 56 24 64 32</td>
</tr>
<tr>
<td>60 52 44 36 28 20 12 4</td>
<td>39 7 47 15 55 23 63 31</td>
</tr>
<tr>
<td>62 54 46 38 30 22 14 6</td>
<td>38 6 46 14 54 22 62 30</td>
</tr>
<tr>
<td>64 56 48 40 32 24 16 8</td>
<td>37 5 45 13 53 21 61 29</td>
</tr>
<tr>
<td>57 49 41 33 25 17 9 1</td>
<td>36 4 44 12 52 20 60 28</td>
</tr>
<tr>
<td>59 51 43 35 27 19 11 3</td>
<td>35 3 43 11 51 19 59 27</td>
</tr>
<tr>
<td>61 53 45 37 29 21 13 5</td>
<td>34 2 42 10 50 18 58 26</td>
</tr>
<tr>
<td>63 55 47 39 31 23 15 7</td>
<td>33 1 41 9 49 17 57 25</td>
</tr>
</tbody>
</table>
Generating Per-Round Keys

- 16 48-bit keys generated
- A subset of 48-bit from the 56 bits

Figure 3-5: Round $i$ for generating $K_i$
A DES Round

Figure 3-6: DES round
Mangler Function

Figure 3-8: Chunk transformation

- R is expanded from 32-bit to 48-bit
- Each S-box is a 6-bit to 4-bit decoder, or 4 4-bit to 4-bit
S-Box

• A substitution which produces a 4-bit output for each possible 6-bit input
• The 4-bit output of each of the 8 S-boxes is combined into a 32-bit quantity whose bits are then permuted
• The permutation ensures: bits of the output of an S-box on one round of DES affects the input of multiple S-boxes on the next round
• Output bits of S-box should not be close to a linear function of input bits
Design Parameters

- Block size: larger block sizes mean greater security but reduced encryption/decryption speed for a given algorithm
- Key size: larger key size means greater security but may decrease encryption/decryption speed
- Number of rounds: multiple rounds offer increasing security, more is not better, sufficient is good enough
- Key generation algorithm: greater complexity in this algorithm should lead to greater difficulty of cryptanalysis
- Round function: greater complexity generally means greater resistance to cryptanalysis
The Avalanche Effect

- Desired property of encryption: a change in one bit of the plaintext or one bit of the key should produce a change in many bits of the ciphertext.

- Table (a): two plaintext with 1-bit difference and a single key are selected.

- Table (b): two keys with 1-bit difference and a single plaintext are selected.

<table>
<thead>
<tr>
<th>Round</th>
<th>Number of bits that differ</th>
<th>Round</th>
<th>Number of bits that differ</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>21</td>
<td>2</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>4</td>
<td>32</td>
<td>4</td>
<td>32</td>
</tr>
<tr>
<td>5</td>
<td>34</td>
<td>5</td>
<td>30</td>
</tr>
<tr>
<td>6</td>
<td>32</td>
<td>6</td>
<td>32</td>
</tr>
<tr>
<td>7</td>
<td>31</td>
<td>7</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>29</td>
<td>8</td>
<td>34</td>
</tr>
<tr>
<td>9</td>
<td>42</td>
<td>9</td>
<td>40</td>
</tr>
<tr>
<td>10</td>
<td>44</td>
<td>10</td>
<td>38</td>
</tr>
<tr>
<td>11</td>
<td>32</td>
<td>11</td>
<td>31</td>
</tr>
<tr>
<td>12</td>
<td>30</td>
<td>12</td>
<td>33</td>
</tr>
<tr>
<td>13</td>
<td>30</td>
<td>13</td>
<td>28</td>
</tr>
<tr>
<td>14</td>
<td>26</td>
<td>14</td>
<td>26</td>
</tr>
<tr>
<td>15</td>
<td>29</td>
<td>15</td>
<td>34</td>
</tr>
<tr>
<td>16</td>
<td>34</td>
<td>16</td>
<td>35</td>
</tr>
</tbody>
</table>
Attacks on DES

- **Brute-force attack**: 56-bit key size not long enough
- 4 weak and 12 semi-weak keys: when $C_0$ and $D_0$ are one of 4 values, 1111..., 0000..., 1010..., 0101...
- Cryptanalysis by exploiting weakness in S-box design
- Differential cryptanalysis: observe the behavior of pairs of text blocks evolving along each round of the cipher, can find a DES key given $2^{47}$ chosen plaintexts
- Linear cryptanalysis: finding linear approximations to describe the transformations performed in DES, can find a DES key given $2^{43}$ known plaintexts
- Timing attacks: information about the key or the plaintext is obtained by observing how long to decrypt various ciphertexts
Multiple Encryption DES

• Encrypting twice with the same key: Problem?

\[
\text{plaintext} \xrightarrow{K} \xrightarrow{K} \text{ciphertext}
\]

• Encrypting twice with two keys: Problem?
(Read [Kaufman] 4.4.1.2 on page 111, the problem is called “meet-in-the-middle attack”)

\[
\text{plaintext} \xrightarrow{K_1} \xrightarrow{K_2} \text{ciphertext}
\]
Triple DES (3DES)

- 3 DES encryptions with 2 keys: 64-bit block, 112-bit key

Why three encryptions, not less or more?
Why two keys, not three?
Why EDE, not EEE or EDD?
Other Block Ciphers

- IDEA: International Data Encryption Algorithm, 64-bit block, 128-bit key

- AES: Advanced Encryption Standard, 128-bit block, 128/192/256-bit key
AES

- Rijndael: invented by Dutch cryptographers
- AES parameters:

<table>
<thead>
<tr>
<th></th>
<th>4/16/128</th>
<th>6/24/192</th>
<th>8/32/256</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key size (words/bytes/bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plaintext block size (words/bytes/bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of rounds</td>
<td>10</td>
<td>12</td>
<td>14</td>
</tr>
<tr>
<td>Round key size (words/bytes/bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expanded key size (words/bytes)</td>
<td>44/176</td>
<td>52/208</td>
<td>60/240</td>
</tr>
</tbody>
</table>
AES Overview
An Encryption Round
Four Stages

One permutation and three substitutions

- Substitute bytes: uses an S-box to perform a byte-by-byte substitution of the block
- ShiftRows: a simple permutation
- MixColumns: a substitution that makes use of arithmetic over GF($2^8$)
- AddRoundKey: a simple bitwise XOR of the current block with a portion of the expanded key
- Each stage is easily reversible
Substitute Bytes

- SubBytes: table lookup with a 16x16 S-box of bytes
- Substitute byte transformation:
S-Box

- **Hex**: $95 \rightarrow 2A$
- **InvSubBytes**: inverse substitute byte transformation, uses an inverse S-box
Example of SubBytes

State Matrices
ShiftRows

- Shift row transformation:

![ShiftRows Diagram]

- Example:

<table>
<thead>
<tr>
<th></th>
<th>87</th>
<th>F2</th>
<th>4D</th>
<th>97</th>
</tr>
</thead>
<tbody>
<tr>
<td>EC</td>
<td>6E</td>
<td>4C</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>4A</td>
<td>C3</td>
<td>46</td>
<td>E7</td>
<td></td>
</tr>
<tr>
<td>8C</td>
<td>D8</td>
<td>95</td>
<td>A6</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>87</th>
<th>F2</th>
<th>4D</th>
<th>97</th>
</tr>
</thead>
<tbody>
<tr>
<td>6E</td>
<td>4C</td>
<td>90</td>
<td>EC</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>E7</td>
<td>4A</td>
<td>C3</td>
<td></td>
</tr>
<tr>
<td>A6</td>
<td>8C</td>
<td>D8</td>
<td>95</td>
<td></td>
</tr>
</tbody>
</table>

- InvShiftRows: inverse shift row transformation, circular shifts in the opposite direction except 1st row
Mixcolumns

- Each byte of a column is mapped into a new value that is a function of all four bytes in that column:

\[
\begin{bmatrix}
  02 & 03 & 01 & 01 \\
  01 & 02 & 03 & 01 \\
  01 & 01 & 02 & 03 \\
  03 & 01 & 01 & 02 \\
\end{bmatrix}
\begin{bmatrix}
  s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\
  s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\
  s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\
  s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \\
\end{bmatrix}
= \begin{bmatrix}
  s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\
  s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\
  s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\
  s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \\
\end{bmatrix}
\]

- Example:

\[
\begin{array}{cccc}
  87 & F2 & 4D & 97 \\
  6E & 4C & 90 & EC \\
  46 & E7 & 4A & C3 \\
  A6 & 8C & D8 & 95 \\
\end{array}
\rightarrow
\begin{array}{cccc}
  47 & 40 & A3 & 4C \\
  37 & D4 & 70 & 9F \\
  94 & E4 & 3A & 42 \\
  ED & A5 & A6 & BC \\
\end{array}
\]

- InvMixColumns: inverse mix column transformation
AddRoundKey

- Columnwise operation: the 128-bit state is bitwise XORed with the 128-bit round key

\[
\begin{array}{cccc}
47 & 40 & A3 & 4C \\
37 & D4 & 70 & 9F \\
94 & E4 & 3A & 42 \\
ED & A5 & A6 & BC \\
\end{array}
\quad \oplus \quad
\begin{array}{ccccc}
AC & 19 & 28 & 57 \\
77 & FA & D1 & 5C \\
66 & DC & 29 & 00 \\
F3 & 21 & 41 & 6A \\
\end{array}
\quad =
\begin{array}{cccc}
EB & 59 & 8B & 1B \\
40 & 2E & A1 & C3 \\
F2 & 38 & 13 & 42 \\
1E & 84 & E7 & D2 \\
\end{array}
\]

- Inverse add round key transformation: identical to AddRoundKey because XOR is its own inverse
Key Expansion

- 128-bit or 4-word key is expanded to 44 words
Strength of Rijndael

- Resistant to brute-force attack
- Resistant to differential and linear cryptanalysis
Reading Assignments

- [Kaufman] Chapter 3, 4.4, 8.5