IDEA  

IDEA X. Lai, J. Massey
ETH, 1990-91

• 128-bit key (billion machines each checking billion keys per second) still would require 10 trillion years, to check all keys

• used in PGP (Pretty Good Privacy) - the most popular public domain program for secure e-mail

• constructed to provide an absolute resistance against differential cryptanalysis
IDEA

Three basic operations:

\[ X \oplus K \quad Y = X + K \mod 2^{16} \quad Y = X \cdot K \mod (2^{16} + 1) \]

where 0 represents \( 2^{16} \)

Corresponding inverse operations:

\[ Y \oplus K^{-1} \quad X = Y - K \mod 2^{16} \quad X = Y \cdot K^{-1} \mod (2^{16} + 1) \]

Half-round of IDEA: Transformation

Forward transformation:

\[ X_a \quad X_b \quad X_c \quad X_d \]
\[ \circlearrowleft K_a \quad \square K_b \quad \square K_c \quad \circlearrowright K_d \]
\[ Y_a \quad Y_b \quad Y_c \quad Y_d \]

Inverse transformation:

\[ Y_a \quad Y_b \quad Y_c \quad Y_d \]
\[ \circlearrowleft K_a^{-1} \quad \square -K_c \quad \square -K_b \quad \circlearrowright K_d^{-1} \]
\[ X_a \quad X_b \quad X_c \quad X_d \]
Half-round of IDEA: Sub-encryption

Forward transformation

\[ W_{in} = X_a \oplus X_b \]
\[ V_{in} = X_c \oplus X_d \]

\[ Y_a = X_a \oplus W_{out} \]
\[ Y_b = X_b \oplus W_{out} \]
\[ Y_c = X_c \oplus V_{out} \]
\[ Y_d = X_d \oplus V_{out} \]

Inverse transformation

\[ W_{in} = X_a \oplus X_b \]
\[ V_{in} = X_c \oplus X_d \]

\[ X_a = Y_a \oplus W_{out} \]
\[ Y_b = Y_b \oplus W_{out} \]
\[ X_c = Y_c \oplus V_{out} \]
\[ X_d = Y_d \oplus V_{out} \]
IDEA - Key Scheduling

128 bit

\[ \begin{array}{cccccccc}
Z_1 & Z_2 & Z_3 & Z_4 & Z_5 & Z_6 & Z_7 & Z_8 \\
\end{array} \]

\[ \begin{array}{cccccccc}
& & & & & & & \\
\end{array} \]

\( \rightarrow \) Rotate 25 positions left

\[ \begin{array}{cccccccc}
Z_9 & Z_{10} & Z_{11} & Z_{12} & Z_{13} & Z_{14} & Z_{15} & Z_{16} \\
\end{array} \]

\[ \begin{array}{cccccccc}
& & & & & & & \\
\end{array} \]

\( \rightarrow \) Rotate 25 positions left

\[ \begin{array}{cccccccc}
& & & & & & & \\
\end{array} \]

Implementing IDEA in Hardware
Modular Multiplication

Special Cases

\[ a \times x \mod 2^k = p_L \]
\[ a \times x \mod 2^{k+1} = p_L - p_H - \text{borrow} \]
\[ a \times x \mod 2^{k-1} = p_L + p_H + \text{carry} \]

Modular Multiplication

Special Case (1)

\[ a \times x \mod 2^{k+1} = (p_H 2^k + p_L) \mod (2^{k+1}) = \]
\[ = (p_H (2^{k+1}-1) + p_L) \mod (2^{k+1}) = \]
\[ = p_L - p_H \mod (2^{k+1}) = \]
\[ = \begin{cases} 
  p_L - p_H & \text{if } p_L - p_H \geq 0 \\
  p_L - p_H + (2^{k+1}) & \text{if } p_L - p_H < 0
\end{cases} \]
\[ = p_L - p_H + \text{borrow} \]

borrow = borrow from subtraction \( p_L - p_H \)
Modular Multiplication

Special Case (2)

\[ a \times \mod 2^{k-1} = (p_H 2^k + p_L) \mod (2^k-1) = \]
\[ = (p_H (2^k \mod 2^{k-1}) + p_L) \mod (2^k-1) = \]
\[ = p_H + p_L \mod (2^k-1) = \]
\[ = \begin{cases} 
    p_H + p_L & \text{if } p_H + p_L < 2^k - 1 \\
    p_H + p_L - (2^k-1) & \text{if } p_H + p_L \geq 2^k - 1 
\end{cases} \]
\[ = p_L + p_H + \text{carry} \]

carry = carry from addition \( p_L + p_H \)

RC5
RC5  Ron Rivest, MIT, 1994

(Ron’s Code 5, Rivest’s Cipher 5)

- **variable key length** (40 bits in the former export version, 128 bits to achieve the same strength as IDEA)

- **variable block size** (depends on the processor word length)

- **variable number of rounds** (determines resistance to linear and differential cryptanalysis; for 9 rounds this resistance is greater than for DES)

- **simplicity of description**

---

RC5

One of the fastest ciphers

Basic operation

Rotation by a variable number of bits

\[ Y = Y \ll X \]
**RC5** \(w/r/b\)

- \(w\) - word size in bits \(w = 16, 32, 64\)
  - input/output block = 2 words = \(2 \cdot w\) bits
  - Typical value: \(w=32 \Rightarrow 64\)-bit input/output block
- \(r\) - number of rounds
- \(b\) - key size in bytes \(0 \leq b \leq 255\)
  - key size in bits = \(8 \cdot b\) bits

**Recommended version:** RC5 32/12/16
- 64 bit block
- 12 rounds
- 128 bit key

<table>
<thead>
<tr>
<th>Encryption</th>
<th>Decryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (|) B = M</td>
<td></td>
</tr>
<tr>
<td>A = A + S[0]</td>
<td></td>
</tr>
<tr>
<td>B = B + S[1]</td>
<td></td>
</tr>
<tr>
<td>for (i= 1) to (r) do</td>
<td></td>
</tr>
<tr>
<td>{</td>
<td></td>
</tr>
<tr>
<td>A = (A(\oplus)B) &lt;&lt;&lt; B + S[2i]</td>
<td></td>
</tr>
<tr>
<td>B = (B(\oplus)A) &lt;&lt;&lt; A + S[2i+1]</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>C = A (|) B</td>
<td></td>
</tr>
<tr>
<td>A (|) B = C</td>
<td></td>
</tr>
<tr>
<td>for (i= r) downto 1 do</td>
<td></td>
</tr>
<tr>
<td>{</td>
<td></td>
</tr>
<tr>
<td>B = ((B - S[2i+1]) &gt;&gt;&gt; A) (\oplus) A</td>
<td></td>
</tr>
<tr>
<td>A = ((A - S[2i])&gt;&gt;&gt;B) (\oplus) B</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
<tr>
<td>B = B - S[1]</td>
<td></td>
</tr>
<tr>
<td>A = A - S[0]</td>
<td></td>
</tr>
<tr>
<td>M = A (|) B</td>
<td></td>
</tr>
</tbody>
</table>
RC5 - Key Scheduling

\[ k \text{ bits of the main key} \]
\[ 2 \cdot r + 2 \text{ round keys} = (2 \cdot r + 2) \cdot w \text{ bits} \]

Two magic constants:
\[ P_w = \text{Odd } ((e-2) \cdot 2^w) \]
\[ Q_w = \text{Odd } ((\varphi-1) \cdot 2^w) \]
\[ e \text{ - base of natural logarithms } \]
\[ e = 2.7182... \]
\[ \varphi \text{ - golden ratio } \]
\[ \frac{x}{y} = \frac{y}{x-y} = 1.6180... \]

RC5 - Key Scheduling

Initialize
\[ S[0] = P_w \]
for i=0 to t-1 do
\[ S[i] = S[i] + Q_w \]
Mix
\[ i = j = 0 \]
\[ A = B = 0 \]
do 3 \cdot \max\{t, c\} times
\[
\begin{align*}
A &= S[i] = (S[i] + A + B) \ll 3 \\
B &= L[j] = (L[j] + A + B) \ll (A+B) \\
i &= (i+1) \mod t \\
j &= (j+1) \mod c
\end{align*}
\]
## RC5 - Resistance to differential and linear cryptanalysis

<table>
<thead>
<tr>
<th># rounds</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>9</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Differential Cryptanalysis</td>
<td>$2^{22}$</td>
<td>$2^{26}$</td>
<td>$2^{32}$</td>
<td>$2^{37}$</td>
<td>$2^{46}$</td>
<td>$2^{63}$</td>
<td>$&gt;2^{64}$</td>
</tr>
<tr>
<td>Linear Cryptanalysis</td>
<td>$2^{37}$</td>
<td>$2^{47}$</td>
<td>$2^{57}$</td>
<td>$&gt;2^{64}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Differential cryptanalysis cannot be applied to RC5 with $#\text{rounds} \geq 13$

Linear cryptanalysis cannot be applied to RC5 with $#\text{rounds} \geq 7$

## Security of Modern Ciphers
**Resistance of modern ciphers against known attacks**

<table>
<thead>
<tr>
<th>Proprietary ciphers built in application software</th>
<th>mostly insecure, seconds on PC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proprietary ciphers with unknown specification</td>
<td>uncertain, impossible to verify</td>
</tr>
<tr>
<td>40-bit “international” version of ciphers</td>
<td>Keys recoverable using several hours with a small network of computers</td>
</tr>
<tr>
<td>DES</td>
<td>Keys can be recovered within 24 hours using a specialized machine worth about $300,000</td>
</tr>
<tr>
<td>Triple DES, DESX, RC5, IDEA</td>
<td>All known attacks impractical</td>
</tr>
</tbody>
</table>

**State of research regarding the security of secret-key ciphers**

- limited number (20-50) of researchers actively involved in cryptanalysis and design of new ciphers
- number of published ciphers > 50
- evaluations of the cipher strength given by designers typically unreliable

**“Honest” cipher** = the best known attack is an exhaustive key search attack

**One can rely only on ciphers analyzed by a large group of qualified researchers**