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

Three basic operations:

\[
\begin{align*}
X \oplus K &= Y \\
X \cdot K &\mod (2^{16} + 1) &= Y
\end{align*}
\]

Corresponding inverse operations:

\[
\begin{align*}
Y \oplus K^{-1} &= X \\
Y \cdot K^{-1} &\mod (2^{16} + 1) &= X
\end{align*}
\]
Half-round of IDEA: Transformation

Forward transformation:

\[
\begin{align*}
X_a &\quad K_a & X_b &\quad K_b & X_c &\quad K_c & X_d &\quad K_d \\
Y_a &\quad K_a^{-1} & Y_b &\quad K_c^{-1} & Y_c &\quad K_b^{-1} & Y_d &\quad K_d^{-1}
\end{align*}
\]

Inverse transformation:

\[
\begin{align*}
X_a &\quad Y_a = X_a \oplus W_{out} & X_b &\quad Y_b = X_b \oplus W_{out} \\
W_{in} &\quad X_c \oplus X_d & V_{in} &\quad X_c \oplus X_d \\
W_{out} &\quad Y_d = X_d \oplus V_{out} & Y_c &\quad Y_c = X_c \oplus V_{out}
\end{align*}
\]

Half-round of IDEA: Sub-encryption

Forward transformation

\[
\begin{align*}
W_a &= X_a \oplus X_b & V_a &= X_c \oplus X_d \\
V_{in} &= X_c \oplus X_d & W_{out} &= X_a \oplus X_b \\
&= W_{out} & V_{out} &= V_{in} \\
Y_a &= X_a \oplus W_{out} & Y_b &= X_b \oplus W_{out} & Y_c &= X_c \oplus V_{out} & Y_d &= X_d \oplus V_{out}
\end{align*}
\]

Inverse transformation

\[
\begin{align*}
W_a &= X_a \oplus X_b & V_a &= X_c \oplus X_d \\
V_{in} &= X_c \oplus X_d & W_{out} &= X_a \oplus X_b \\
&= W_{out} & V_{out} &= V_{in} \\
X_a &= Y_a \oplus W_{out} & X_b &= Y_b \oplus W_{out} & X_c &= Y_c \oplus V_{out} & X_d &= Y_d \oplus V_{out}
\end{align*}
\]
IDEA - Key Scheduling

128 bit

K

Z_1 Z_2 Z_3 Z_4 Z_5 Z_6 Z_7 Z_8

K

Z_9 Z_{10} Z_{11} Z_{12} Z_{13} Z_{14} Z_{15} Z_{16}

Rotate 25 positions left

Implementing IDEA in Hardware

Modular Multiplication

Special Cases

k bits

\[ a \times x \mod 2^k = p_L \]

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

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

Special Case (1)

\[ a \times \text{mod} \ 2^{k+1} = (p_H \ 2^k + p_L) \text{mod} \ (2^k+1) = \]
\[ = (p_H \ (2^k+1-1) + p_L) \text{mod} \ (2^k+1) = \]
\[ = p_L \cdot p_H \text{mod} \ (2^k+1) = \]
\[ = p_L \ - p_H \quad \text{if } p_L - p_H \geq 0 \]
\[ \left\{ \begin{array}{ll}
  p_L - p_H + (2^k+1) & \text{if } p_L - p_H < 0 \\
  p_L - p_H + \text{borrow} & \\
\end{array} \right. \]

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

Modular Multiplication

Special Case (2)

\[ a \times \text{mod} \ 2^{k-1} = (p_H \ 2^k + p_L) \text{mod} \ (2^k-1) = \]
\[ = (p_H \ (2^k \text{mod} \ 2^k-1) + p_L) \text{mod} \ (2^k-1) = \]
\[ = p_H + p_L \text{mod} \ (2^k-1) = \]
\[ = p_H + p_L \quad \text{if } p_H + p_L < 2^k - 1 \]
\[ \left\{ \begin{array}{ll}
  p_H + p_L - (2^k - 1) & \text{if } p_H + p_L \geq 2^k - 1 \\
\end{array} \right. \]
\[ = p_L + p_H + \text{carry} \]

carry = carry from addition \( p_L + p_H \)
**RC5**  
(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\( w \) bits
  - Typical value:  
    \[ w=32 \Rightarrow 64\text{-bit input/output block} \]
- **r** - number of rounds
- **b** - key size in bytes  \( 0 \leq b \leq 255 \)
  - key size in bits = 8\( b \) bits

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

### Encryption

<table>
<thead>
<tr>
<th>Operation</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \parallel B = M$</td>
<td>$A \parallel B = C$</td>
</tr>
<tr>
<td>$A = A + S[0]$</td>
<td>for $i = r$ downto 1 do</td>
</tr>
</tbody>
</table>
| $B = B + S[1]$ | $\{ \begin{align*}
A &= ((B - S[2i+1]) >>> A) \oplus A \\
B &= B - S[1] \\
A &= A - S[0]
\end{align*} \}$ |
| for $i = 1$ to $r$ do | $M = A \parallel B$ |
| $A = (A \oplus B) <<< B + S[2i]$ | |
| $B = (B \oplus A) <<< A + S[2i+1]$ | |
| $C = A \parallel B$ | |

### Decryption

<table>
<thead>
<tr>
<th>Operation</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \parallel B = C$</td>
<td>$A \parallel B = M$</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 $i = r$ downto 1 do</td>
<td></td>
</tr>
<tr>
<td>$A = ((A - S[2i]) &gt;&gt;&gt; B) \oplus B$</td>
<td></td>
</tr>
<tr>
<td>$B = ((B - S[2i+1]) &gt;&gt;&gt; A) \oplus A$</td>
<td></td>
</tr>
<tr>
<td>$A = A + S[0]$</td>
<td></td>
</tr>
<tr>
<td>$B = B + S[1]$</td>
<td></td>
</tr>
<tr>
<td>$C = A \parallel B$</td>
<td></td>
</tr>
</tbody>
</table>

---

## 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
- $A = S[i] = (S[i] + A + B) <<< 3$
- $B = L[i] = (L[i] + A + B) <<< (A+B)$
- $i = (i+1) \mod t$
- $j = (j+1) \mod c$

### Key Scheduling

- $k$ bits of the main key

- $2 \cdot r + 2$ round keys = $(2 \cdot r + 2) \cdot w$ bits

### Magic Constants

- $P_w = \text{Odd}\left((e-2) \cdot 2^w\right)$
- $Q_w = \text{Odd}\left((\phi^{-1}) \cdot 2^w\right)$

### Golden Ratio

- $\phi = \frac{x}{y} = \frac{y}{x-y} = 1.6180...$

### Base of Natural Logarithms

- $e = 2.7182...$
RC5 - Resistance to differential and linear cryptanalysis

<table>
<thead>
<tr>
<th>Plaintext requirement</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><strong>Differential Cryptanalysis</strong></td>
<td>$2^{22}$</td>
<td>$2^{36}$</td>
<td>$2^{22}$</td>
<td>$2^{37}$</td>
<td>$2^{26}$</td>
<td>$2^{63}$</td>
<td>$&gt;2^{64}$</td>
</tr>
<tr>
<td><strong>Linear Cryptanalysis</strong></td>
<td>$2^{37}$</td>
<td>$2^{47}$</td>
<td>$&gt;2^{57}$</td>
<td>$&gt;2^{64}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Differential cryptanalysis cannot be applied to RC5 with #rounds ≥ 13
Linear cryptanalysis cannot be applied to RC5 with #rounds ≥ 7

Security of Modern Ciphers

Resistance of modern ciphers against known attacks

<table>
<thead>
<tr>
<th>Ciphers</th>
<th>Mostly insecure, seconds on PC</th>
<th>Mostly insecure, seconds on PC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proprietary ciphers built in application software</td>
<td>mostly insecure, seconds on PC</td>
<td>mostly insecure, seconds on PC</td>
</tr>
<tr>
<td>Proprietary ciphers with unknown specification</td>
<td>uncertain, impossible to verify</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>
<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>
<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>
<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