FEAL Cipher

Team RZC:
Carlos Leonardo
Zachary Miller
Outline

• FEAL block cipher.
• Our Ad-Hoc Attack.
• Results and analysis
• Other Attacks on FEAL
• What we learned
• Future work
• Conclusion
• Questions
• References
FEAL block cipher.

- It uses 64 bits long Key, plaintext and ciphertext \[1\].
- It was supposed to be as secure as DES without using special hardware \[2\].
- Various Attacks have proven FEAL insecure\[2,3\].
- The 1st version used 4 Feistel rounds and later it was extended to 8 rounds \[1\].
S-Function

• Implements the S-Boxes and accepts three byte parameters and return a single byte:
  o byte A
  o byte B
  o byte \( \delta = 0 \) or 1

• \( S(A,B,\delta) = \text{RotateL2}(T) \)

• \( T = A + B + \delta \mod 256 \)
Function $F_k$

- Used during Sub-Key Generation.
- Parameters: $\alpha$ and $\beta$: 32 bit halves of initial key
- Returns two 16-bit Sub-Keys
Function F

• Used for Encryption
• Parameters:
  o $\alpha$: 32 bit from Feistel Round
  o $\beta$: 16 bit sub-key
Key Generation
Encryption

Decryption
Ad-Hoc Attack on FEAL

- To attach the FEAL Cipher, we first reduced the number of rounds to 1.
- We then modify the F-Function as described in [2].
Modified FEAL Algorithm

Modified F-Function

Plaintext block (64 bits)

Ciphertext block (64 bits)

One Round FEAL
Ad-Hoc Attack

• We can directly derive $K_2$ for any plaintext-ciphertext pair
• To find $K_1$, we set up equations and guess a value for a given byte in the key
• Each guess is independent so we are doing 256 operations 4 times.
• Each guess for a given byte of $k_1$ is called a candidate key
Ad-Hoc Attack Cont.

- Calculate candidate keys for each plaintext-ciphertext pair
- Unfortunately each byte for K1 is not the same
- However, candidate keys occur frequently
- In our observations
  - 4 candidate bytes was the smallest
  - 84 was the largest
- Based on this, we figure:
  - To derive a workable key, find a permutation of the common occurring key bytes (84^4 ~ 49 million) in worse observed case.
  - Fortunately this is the worse case:
    - Observed much better
Our Ad-Hoc Program

- Takes in 2 parameters
  - n = number of plaintext-ciphertext pairs
  - k = original 64-bit key used to find ciphertexts
- Runs the FEAL Encipherment algorithm on each randomized plaintext
- Derives a value for K2
- Finds candidate keys for each byte of K1
- Repeats on all plaintext-ciphertext pairs
- Outputs the list of all candidate keys for each byte of K1 and the value for K2.
Results

• This actually performed well.
  o The orders of the number of candidate keys for a given byte stayed fairly small (computable).

• Needed between 1,000 and 10,000 message pairs to compute all possible candidate keys
  o We even tried up to 10 million and we were not adding any new keys

• This means that the guessing of the keys takes $10,000(2^8 \times 4)$ or $10,000(2^{10})$ or $10,240,000$ operations.
  o Could be less depending on the original 64-bit key
## Sample of Results

<table>
<thead>
<tr>
<th>Original Key</th>
<th>#K1_1</th>
<th>#K1_2</th>
<th>#k1_3</th>
<th>#k1_4</th>
<th>Total Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0123456789ABCDEF</td>
<td>54</td>
<td>45</td>
<td>27</td>
<td>60</td>
<td>3936600</td>
</tr>
<tr>
<td>AAAAAAAAAAAAAAAAAAAAAAAAA</td>
<td>80</td>
<td>79</td>
<td>24</td>
<td>72</td>
<td>10920960</td>
</tr>
<tr>
<td>FEDCBA9876543210</td>
<td>14</td>
<td>46</td>
<td>40</td>
<td>4</td>
<td>103040</td>
</tr>
<tr>
<td>5738290BC37D3FEA</td>
<td>80</td>
<td>84</td>
<td>48</td>
<td>56</td>
<td>18063360</td>
</tr>
</tbody>
</table>
Comparison

• [2] Presented a Linear Attack on FEAL-4, FEAL-6 and FEAL-8
  o Needed 5 plaintexts to break FEAL-4 in 6 minutes.
  o Unsure of how many operations.
  o FEAL-6 needed 100 texts and FEAL-8 needed $2^{15}$.
  o Much better than our attack

• [3] Differential Attack which broke FEAL-8 with 1,000 pairs with a 95% accuracy
What We learned

• In general, Cryptography is a difficult subject. Algorithms can be relatively easy to implement, but can be fairly difficult to break given certain circumstances.

• One approach will not solve all of the problems, cryptanalysis often requests to adapt to new situations.

• Split big tasks and keep going.
Future Work

- Running some statistical measurements to gauge how effective this method is.
- Find another place to add a 3rd key for the attack which could increase the likelihood that a given key would work for more plaintext ciphertext pairs.
References


Questions?