MATH 314 - Class Notes

10/13/2015

Scribe: Kirill Vorobyev

Summary: Covered mini-mission 6, involving simple DES, with classmates comparing answers/strategies. Went over SageMath SDES code demo and discussed, in depth, how the functions operate. Started discussion on differential cryptanalysis.

Notes:
SageMath Code Demo Explanation:

- Start with list of lists containing S_1 and S_2
- `Bin(S)` converts numbers written with 1's and 0's into equivalent binary representation
- `Expander(L)` expands a string from 6 to 8 bits for SDES
- `XOR(L,M)` performs binary xor addition on two string **L,M** provided they are the same length
- `roundkey(K,i)` returns 8 bit key for round i from master 9 bit key
- `bind2int(L)` converts binary list to integer
- `split(M)` splits list into two equal halves
- `SDES(R,`K_i`)` performs function f
- `SDESround(M,`K_r`)` performs 1 round of SDES using roundkey K_r
- `SDES(M,K,r)` performs SDES on **M** with the key **K** using **r** rounds

Differential Cryptanalysis

- One method to attack Feistel System Ciphers
- SDES is a Feistel System Cipher
- Recall what a Feistel System is

How differential cryptanalysis works on 3 rounds of SDES:

- Choose Plaintext
- Split into L_0, R_0
- Encrypt 3 times/rounds using SDES
- Start with known inputs L_0 and R_0
- L_0 $\qquad\qquad$ R_0

1. $L_1 = R_0$ $\qquad\qquad$ $R_1 = L_0 \oplus f(R_0, K_1)$

2. $L_2 = L_0 \oplus f(R_0, K_1)$ $\quad R_2 = ...$
3. $L_3 = R_2$ $\quad R_3 = L_2 \oplus f(R_2, K_3)$

- End with known outputs L_3 and R_3
- $R_3 = L_0 \oplus f(R_0, K_1) \oplus f(R_2, K_3)$

<u>We want to find **K**</u>
Now choose a new plaintext L_0^* , R_0^* where $R_0 = R_0^*$ but $L_0 \neq L_0^*$
Feed through SDES as well and get out:

- $R_3^* = L_0^* \oplus f(R_0^*, K_1) \oplus f(R_2^*, K_3)$

  $R_3 = L_0 \oplus f(R_0, K_1) \oplus f(R_2, K_3) \leftarrow$ move down for reference

$R_3 \oplus R_3^* = L_0^* \oplus L_0 \oplus f(R_2^*, K_3) \oplus f(R_2, K_3)$

$(R_3 \oplus R_3^*) \oplus (L_0^* \oplus L_0) = f(L_3^*, K_3) \oplus f(L_3, K_3)$
Note: $R_2^* = L_3^* and R_2 = L_3$
At this point we know L_0,L_0^*,R_3,R_3^*,L_3,L_3^* and f. The only thing we do not know is K_3

$f(L_3, K_3) : E(L_3) \oplus K_3$
This is input 1.
We must take the first 4 bits of the above and run them through S-box 1 to get our **output**
$f(L_3^*, K_3) : E(L_3^*) \oplus K_3$
This is input 1*.
We must take the first 4 bits of the above and run them through S-box 1 to get our **output***

Note: Solving for input 1 is the same as solving for the first 4 bits of K_3
We don't know <u>Input1 or Input1*</u>
but if we figure out what they are we can get the first 4 bits of K_3
Note that Input1 $\oplus$ Input1* is the first 4 bits of
$(E(L_3) \oplus K_3) \oplus (E(L_3^*) \oplus (K_3)) = E(L_3) + E(L_3^*)$
We know <u>Input1 $\oplus$ Input1*</u>

Ex.
We know $L_3 = 101110$
We know $L_3^* = 000010$
$E(L_3) = 10111110$
$E(L_3^*) = 00000010$
First 4 bits of $E(L_3) \oplus E(L_3^*)$ are 1011
This is Input1 $\oplus$ Input1*
We can use the fact that Input1 $\oplus$ Input1* = 1101 to reduce the possible inputs into S-boxes to 16 possibilities