MATH 314 Lecture Notes

4/4/2019

Scribe: Caroline Eilman

Summary: This class AES (Advanced Encryption System) was discussed, and the mechanics of SAES (Simplified AES) were outlined and practiced.

Notes:

Advanced Encryption Standard (AES)

- Submitted in late 90s and adapted into AES
- Not a Feistel Cipher
- One S-box which was openly explained algebraically upon its release
- Faster, more random, more secure than DES
- In SAES, uses F_{16} field mod the irreducable polynomial $x^4 + x + 1$

Calculating Round Keys in SAES:
The master key is 16 bits.
Keys, in a sense, are made of two "words" in SAES or 8 bit chunks.
W_0 and W_1 are each made of half the master key, 0 the left and 1 the right.
After that:

- Even $W_{2i} = g(W_{2i-1}) XOR W_0$
- Odd $W_{2i+1} = W_{2i} XOR W_0$

G(word)=

1. W_{2i-1} split into two 4 bit halves.
2. Take both halves and find their new s-box values.
3. Use the new first half as the second half of the final round key.
4. Use the new second half and XOR it with $x^{i+2} mod(x^4 + x + 1)$
5. Use that new value as the first half of the final round key.
6. Repeat these steps until you have the fifth "word".

Each round key is made up of two words. Simply add the second word to the end of the first.
Rk0=w0+w1
Rk1=w2+w3
Rk2=w4+w5

SAES Steps (same as AES, just simpler, and only 2 rounds)

1. XOR round key with text
2. Shift Rows
3. Mix Columns
4. Substitute (break into 4 bit 'nibbles' run through s-box)

*For Round 0, only perform step 4.
*For last round, skip step 3.

Bit Representation of $\mathbf{F_{16}}$ field

Represent each individual "number" as a four bit binary number.
The first bit is whether x^3 exists.
The second bit is whether x^2 exists.
The third bit is whether x exists.
The fourth bit is whether 1 exists.
(If it 'exists', use a 1, if not, then 0.)

S-box

To use, take a 4 bit chunk. First two numbers are the row and the second two determine the column.
The new 4 bit chunk is the transformed 4 bit chunk.

	00	01	10	11
00	1001	0100	1010	1011
01	1101	0001	1000	0101
10	0110	0010	0000	0011
11	1100	1110	1111	0111

Example:

Master key: `0100 1010 1111 0101`
W0=`0100 1010`
W1=`1111 0101`
W2=$g(W_1)$ XOR W_0
`1111 0101 > 0101 1111`

S box
`0001 0111`
XOR 0001 with $x^{i+2} = x^{1+2} = x^3$
0001
+1000=
1001
$G(W_1) = 10010111$
XORed with W_0
$W_2 = 11011101$

$W_3 = W_2 XOR W_1 = 11011101 XOR 11110101 = 00101000$
so on until have all 5 words.
W_4=1000 0111
W_5=1010 1111

$R_{k0} = W_0 + W_1$=0100 1010 1111 0101
$R_{k1} = W_2 + W_3$=1101 1101 0010 1000
$R_{k2} = W_4 + W_5$=1000 0111 1010 1111

Now to take these round keys and use them:
Plaintext: `1000 0111 0011 1011`
Round 0:
Only do step 4, xor with round key.
R_{k0} XOR plaintext=1100 1101 1100 1110

Round 1:
Do all steps.
This is about where the class stopped due to time constraints.