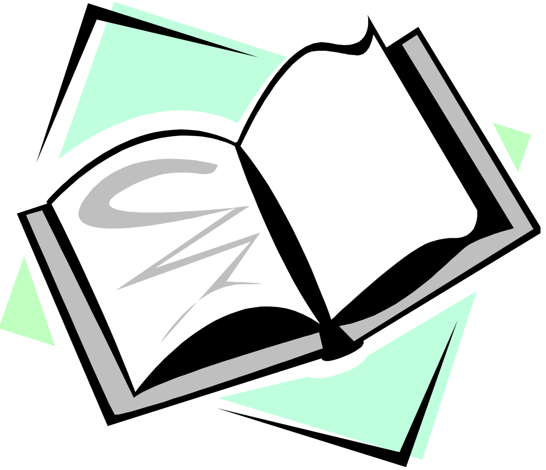
Cryptography
CS 555

Topic 10: Block Cipher Security & AES

Outline and Readings

- Outline
  - Attacks against block ciphers
  - Differential cryptanalysis
  - Linear cryptanalysis
  - Double & triple encryption
  - AES



- Readings:
  - Katz and Lindell: 5.4, 5.5, 5.6

Block Cipher Security

- Two attack objectives
  – Key Recovery
  – Distinguish from a random permutation

- Four attack modes
  – Ciphertext only
  – Known plaintext
  – Chosen plaintext
  – Chosen ciphertext

Attacking Block Ciphers

- Standard attacks
  - exhaustive key search
  - dictionary attack
  - differential cryptanalysis
  - linear cryptanalysis

- Side channel attacks against implementations.
  - Timing attacks
  - Power consumption attacks
  - Fault injection attacks

Chosen-Plaintext Dictionary Attacks Against Block Ciphers

- Construct a table with the following entries
  - (K, $E_K[0]$) for all possible key K
  - Sort based on the second field (ciphertext)
  - How much time does this take?
- To attack a new key K (under chosen message attacks)
  - Choose 0, obtain the ciphertext C, looks up in the table, and finds the corresponding key
  - How much time does this step take?
- Trade off space for time

Differential Cryptanalysis

- Main idea:
  - This is a chosen plaintext attack,
  - The attacker knows many (plaintext, ciphertext) pairs
  - Difference $\Delta_P = P_1 \oplus P_2$, $\Delta_C = C_1 \oplus C_2$
  - Distribution of Δ_C's given Δ_P may reveal information about the key (certain key bits)
  - After finding several bits, use brute-force for the rest of the bits to find the key.

Differential Cryptanalysis of DES

- Surprisingly … DES was resistant to differential cryptanalysis.
- At the time DES was designed, the authors knew about differential cryptanalysis. S-boxes were designed to resist differential cryptanalysis.
- Against 8-round DES, attack requires 2^{38} known plaintext-ciphertext pairs.
- Against 16-round DES, attack requires 2^{47} chosen plaintexts.
- Differential cryptanalysis not effective against DES in practice.

Linear Cryptanalysis of DES

- Introduced in 1993 by M. Matsui

- Instead of looking for isolated points at which a block cipher behaves like something simpler, it involves trying to create a simpler approximation to the block cipher as a whole.

Basic idea of linear cryptanalysis

- Suppose that
- **(*) Pr** [$M_{i1} \oplus M_{i2} \oplus \ldots \oplus M_{iu}$
  $\oplus\ C_{j1} \oplus C_{j2} \oplus \ldots \oplus C_{jv}$
  $\oplus\ K_{p1} \oplus k_{p2} \oplus \ldots \oplus k_{pw} = 1$] $= 0.5 + \varepsilon$
- Then one can recover some key bits given large number of PT/CT pairs
- For DES, exists (*) with $\varepsilon = 2^{-21}$
- Using this method, one can find 14 key bits using $(2^{21})^2$ PT/CT pairs

Linear Cryptanalysis of DES

- M. Matsui showed (1993/1994) that DES can be broke:
  - 8 rounds: 2^{21} known plaintext
  - 16 rounds: 2^{43} known plaintext, 40 days to generate the pairs (plaintext, ciphertext) and 10 days to find the key
- The attack has no practical implication, requires too many pairs.
- Exhaustive search remains the most effective attack.

DES Strength Against Various Attacks

Attack Method	Known	Chosen	Storage complexity	Processing complexity
Exhaustive precomputation	-	1	2^{56}	1
Exhaustive search	1	-	negligible	2^{55}
Linear cryptanalysis	2^{43} 2^{38}	- -	For texts	2^{43} 2^{50}
Differential cryptanalysis	- 2^{55}	2^{47} -	For texts	2^{47} 2^{55}

The weakest point of DES remains the size of the key (56 bits)!

Double Encryption:

- Given a block cipher $\mathbf{E}_k[m]$,
- Define $\mathbf{Enc}_{k1,k2}[m] = \mathbf{E}_{k1}[\mathbf{E}_{k2}[m]]$
- The "Meet-in-the-middle" attack
  - Given a pair (m,c), we have $\mathbf{D}_{k1}[c] = \mathbf{E}_{k2}[m]$
  - Build table of all encryptions of *m*
  - Then for each possible k, test if $\mathbf{D}_k(c)$ is in the table
  - For 2DES, this takes about 2^{56} time
  - Requires $\approx 2^{56}$ space $\approx 10^{16}$
- Effective key length is 56, instead of 2*56=112

Triple Encryption

- Let $\mathbf{E}_k[M]$ be a symmetric block cipher
- Variant 1: $3\mathbf{E}_{k1,k2,k3}[M] = \mathbf{E}_{k1}[\mathbf{D}_{k2}[\mathbf{E}_{k3}[M]]]$
  - Observe: when k1=k2=k3, $3\mathbf{E}_{k1,k2,k3}[M]=\mathbf{E}_k[M]$
  - For triple DES, key=168 bits
  - Effective key length is only 112 bits because of the meet-in-the-middle attack.

- Variant 2: $3\mathbf{E}_{k1,k2}[M] = \mathbf{E}_{k1}[\mathbf{D}_{k2}[\mathbf{E}_{k1}[M]]]$
  - Given one pair (m,c), no known attack with less than 2^{2n} time
  - There exists a 2^n chosen-plaintext attack using 2^n chosen pairs

Strengthening DES to avoid Exhaustive Search: DES-X

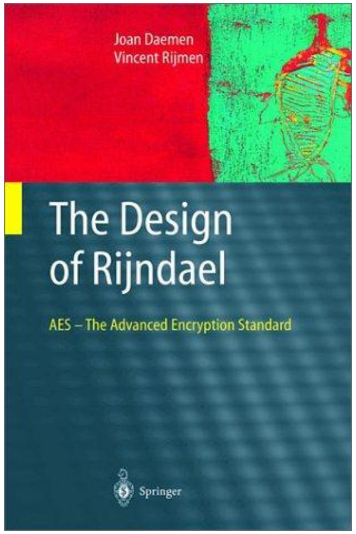
- Given block cipher $\mathbf{E}_k$
- Define $\mathbf{E}\mathrm{X}_{k1,k2,k3}(M)=\mathbf{E}_{k2}(M \oplus k3) \oplus k1$
- DESX: key-length=2*64+56=184 bits
- Increases effective key length
- Fast!

Advanced Encryption Standard

- In 1997, NIST made a formal call for algorithms stipulating that the AES would specify an unclassified, publicly disclosed encryption algorithm, available royalty-free, worldwide.
- Goal: replace DES for both government and private-sector encryption.
- The algorithm must implement symmetric key cryptography as a block cipher and (at a minimum) support block sizes of 128-bits and key sizes of 128-, 192-, and 256-bits.
- In 1998, NIST selected 15 AES candidate algorithms.
- On October 2, 2000, NIST selected **Rijndael** (invented by Joan Daemen and Vincent Rijmen) to as the AES.

AES Features

- Designed to be efficient in both hardware and software across a variety of platforms.
- Not a Feistel Network
- Block size: 128 bits
- Variable key size: **128, 192, or 256 bits.**
- Variable number of rounds (10, 12, 14):
  - 10 if K = 128 bits
  - 12 if K = 192 bits
  - 14 if K = 256 bits
- No known weaknesses



Overview of Rijndael/AES

- Essentially a Substitution-Permutation Network
- 128-bit round key used for each round:
  - 128 bits = 16 bytes = 4 words
  - needs N+1 round keys for N rounds
  - needs 44 words for 128-bit key (10 rounds)
- State: 4 by 4 array of bytes
  - 128 bits = 16 bytes

Rijandael: High-Level Description

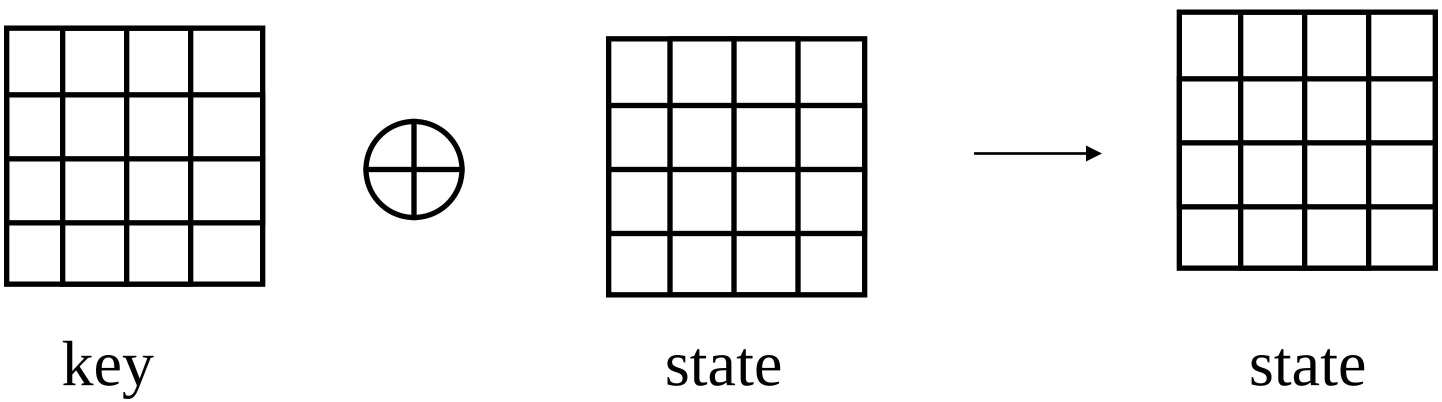
```
State = X
AddRoundKey(State, Key_0)                    (op1)
for r = 1 to Nr - 1
      SubBytes(State, S-box)                 (op2)
      ShiftRows(State)                       (op3)
      MixColumns(State)                      (op4)
      AddRoundKey(State, Key_r)
endfor
SubBytes(State, S-box)
ShiftRows(State)
AddRoundKey(State, Key_Nr)
Y = State
```

AddRound Key

State is represented as follows (16 bytes):

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

AddRoundKey(State, Key):



SubBytes

- Byte substitution using non-linear S-Box (independently on each byte).
- S-box is represented as a 16x16 array, rows and columns indexed by hexadecimal bits
- 8 bytes replaced as follows: 8 bytes defines a hexadecimal number rc, then $s_{r,c}$ = binary(S-box(r, c))

Rijandael S-box

- How is AES S-box different from DES S-box?
  - Only one S-box
  - The S-box is not random; rather it is based on modular arithmetic with polynomials in the field

$$\mathrm{F}_{2^x} = \mathrm{Z}_2[x] \big/ (x^8 + x^4 + x^3 + x + 1)$$

  - as it can be defined algebraically, it can be easily analyzed, can be proven that linear and differential cryptanalysis fail

S-box Table

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	**ED**	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	3	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Example: hexa 53 is replaced with hexa ED

Rijandael: High-Level Description

Diffusion Step

State = X

AddRoundKey(State, Key_0) (op1)

for r = 1 to Nr - 1

    SubBytes(State, S-box) (op2)

    ShiftRows(State) (op3)

    MixColumns(State) (op4)

    AddRoundKey(State, Key_r)

endfor

SubBytes(State, S-box)

ShiftRows(State)

AddRoundKey(State, Key_{Nr})

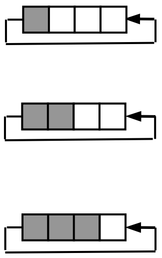
Y = State

ShiftRows

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

$\longrightarrow$

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,0}$
$S_{2,2}$	$S_{2,3}$	$S_{2,0}$	$S_{2,1}$
$S_{3,3}$	$S_{3,0}$	$S_{3,1}$	$S_{3,2}$



MixColumns

- Interpret each column as a vector of length 4.
- Each column of State is replaced by another column obtained by multiplying that column with a matrix in F_{2^x}

Coming Attractions …

- Block cipher encryption modes
- Reading: Katz & Lindell: 3.6.3, 3.6.4, 3.7