Cryptography (5830)

Overview of reduction argument for CTR[E]
Block cipher history
Feistel and DES
Differential cryptanalysis
AES high level

Block ciphers

Family of permutations, one permutation for each key

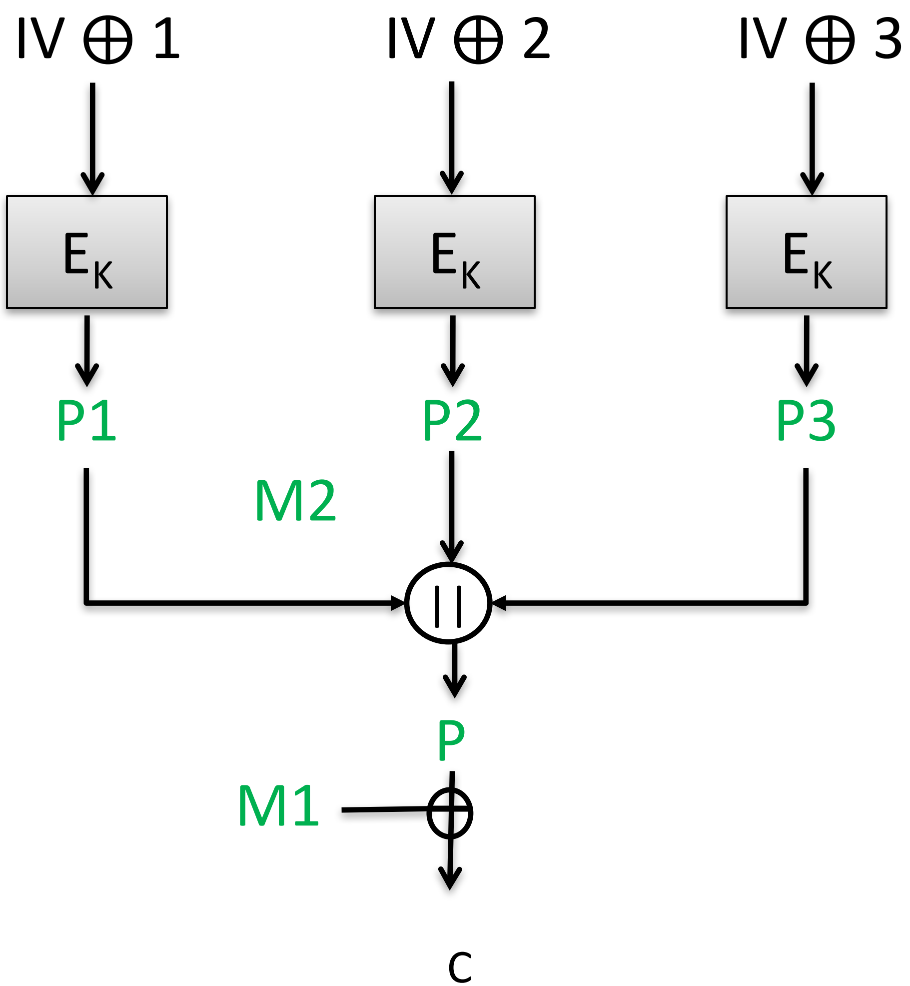
$$E : \{0,1\}^k \times \{0,1\}^n \dashrightarrow \{0,1\}^n$$

Use notation $E(K,X) = E_K(X) = Y$
Define inverse $D(K,Y) = D_K(Y) = X$ such that $D_K(E_K(X)) = X$
E,D must be efficiently computable

Key generation: pick K uniformly at random from $\{0,1\}^k$

Nowadays $k \geq 128$

CTR[E] encryption



Output IV || C

CTR-mode SE scheme

CTR[E] (counter-mode using block cipher E) is the following scheme:

<u>Kg():</u>
K <-\$ $\{0,1\}^k$

Pick a random key

<u>Enc(K,M):</u>
L <- |M| ; m <= ceil(L/n)
IV <-\$ $\{0,1\}^n$
P <- $\mathrm{trunc}_L(E_K(IV \oplus 1) \| \cdots \| E_K(IV \oplus m))$
Return (IV, $P \oplus M$)

$\mathrm{trunc}_L(\ )$ outputs first L bits of input

<u>Dec(K,(IV,C)):</u>
L <- |C| ; m <= ceil(L/n)
P <- $E_K(IV \oplus 1) \| \cdots \| \mathrm{trunc}(E_K(IV \oplus m))$
Return (IV, $P \oplus C$)

Assume ciphertext can be parsed into IV and remaining ciphertext bits

CTR mode using stream cipher abstraction

CTR[E] (counter-mode using block cipher E) is the following scheme:

<u>Kg():</u>
K <-\$ $\{0,1\}^k$

Pick a random key

<u>Enc(K,M):</u>
L <- |M|
IV <-\$ $\{0,1\}^n$
P <- G(K,IV,L)
Return (IV, P $\oplus$ M)

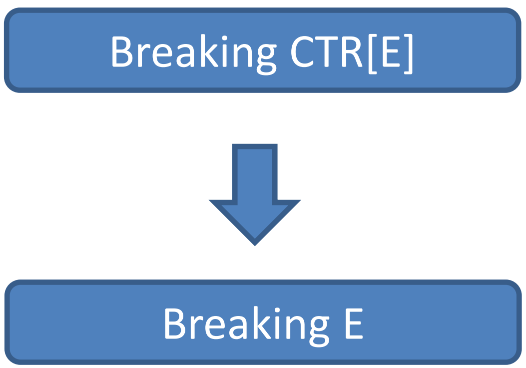
<u>Dec(K,(IV,C)):</u>
L <- |C|
P <- G(K,IV,L)
Return (IV, P $\oplus$ C)

Assume ciphertext can be parsed into IV and remaining ciphertext bits

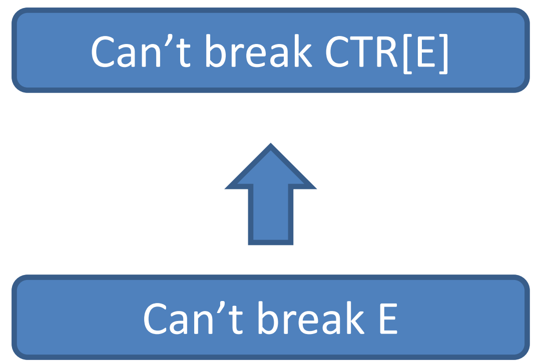
Reduction-based security analysis

Goal: show that if stream cipher is secure, then encryption is secure

Reduction targets showing:



Logical implication:



Reduces security analysis task to analyzing block cipher

Confidence in block cipher security gives confidence in scheme’s security

Two-step proof game plan

Goal: show break against CTR[E] implies break against E

<u>Split into two steps</u>

Breaking scheme

$\Downarrow$

Breaking stream cipher

$\Downarrow$

Breaking block cipher

Attacker $\mathcal{A}$ violates
(t, ϵ)-indistinguishability security

$\Downarrow$

For any $\mathcal{A}$ need to give $\mathcal{B}$ for which $t' \approx t$ and $\epsilon' \approx \epsilon$

Attacker $\mathcal{B}$ that violates
(t', ϵ')-pseudorandom generator security of G

$\Downarrow$

For any $\mathcal{B}$ need to give C with $t'' \approx t'$, $\epsilon'' \approx \epsilon'$, q smallish

Attacker C that violates
(t'', q, ϵ'')-pseudorandom function security of E

Breaking CTR[E] ⇨ Breaking G

$\underline{\text{IND}(\text{SE},\mathcal{A})\text{:}}$
$(st, M_0, M_1) \leftarrow\$ \mathcal{A}_1$
$K \leftarrow\$ Kg$; $b \leftarrow\$ \{0,1\}$
$C \leftarrow\$ Enc(K, M_b)$
$b' \leftarrow\$ \mathcal{A}_2(st, C)$
Return $(b = b')$

$\Pr[\text{IND}(\text{SE},\mathcal{A}) = 1] = 1/2 + \epsilon$

$\underline{\text{PRG1}(G,L,\mathcal{B})\text{:}}$
$K \leftarrow\$ Kg$
$IV \leftarrow\$ \{0,1\}^n$
$P \leftarrow G(K, IV, L)$
$d \leftarrow\$ \mathcal{B}(IV, P)$
Return d

$\underline{\text{PRG0}(G,L,\mathcal{B})\text{:}}$
$K \leftarrow\$ Kg$
$IV \leftarrow\$ \{0,1\}^n$
$P \leftarrow\$ \{0,1\}^L$
$d \leftarrow\$ \mathcal{B}(IV, P)$
Return d

$\epsilon' = |\ \Pr[\text{PRG1}(G,L,\mathcal{B}) = 1] - \Pr[\text{PRG0}(G,L,\mathcal{B}) = 1]\ |$

$\underline{\mathcal{B}(IV,P)\text{:}}$
$(st, M_0, M_1) \leftarrow\$ \mathcal{A}_1$
$b \leftarrow\$ \{0,1\}$
$b' \leftarrow\$ \mathcal{A}_2(st, IV, M_b \oplus P)$
If $(b = b')$ then
  Return 1
Return 0

$\mathcal{B}$ runs in time that of $\mathcal{A}$ plus small overhead

$\Pr[\text{PRG1}(G,L,\mathcal{B}) = 1] = \Pr[\text{IND}(\text{SE},\mathcal{A}) = 1]$
$\Pr[\text{PRG0}(G,L,\mathcal{B}) = 1] = \Pr[b = b'] = 1/2$

$$\begin{aligned}\epsilon' &= \Pr[\text{IND}(\text{SE},\mathcal{A}) = 1] - 1/2 \\ &= \epsilon + 1/2 - 1/2 \\ &= \epsilon\end{aligned}$$

If $\mathcal{A}$ can learn anything about message encrypted, then G's output is not random-looking

Breaking G ⇨ Breaking E

$\underline{\text{PRG1}(G,L,\mathcal{B})\text{:}}$
K <-\$ Kg
IV <-\$ $\{0,1\}^n$
P <- G(K,IV,L)
d <-\$ $\mathcal{B}$(IV,P)
Return d

$\underline{\text{PRG0}(G,L,\mathcal{B})\text{:}}$
K <-\$ Kg
IV <-\$ $\{0,1\}^n$
P <-\$ $\{0,1\}^L$
d <-\$ $\mathcal{B}$(IV,P)
Return d

$\underline{\text{PRF1}(E,\mathcal{C})\text{:}}$
K <-\$ $\{0,1\}^k$
b' <-\$ $\mathcal{C}^{E_K}()$
Return b'

$\underline{\text{PRF0}(E,\mathcal{C})\text{:}}$
F <-\$ Func(n)
b' <-\$ $\mathcal{C}^F$ ()
Return b'

$\epsilon' = |\ \Pr[\text{PRG1}(G,L,\mathcal{B}) = 1] - \Pr[\text{PRG0}(G,L,\mathcal{B}) = 1]\ |$

$\epsilon'' = |\ \Pr[\text{PRF1}(E,\mathcal{C}) = 1] - \Pr[\text{PRF0}(E,\mathcal{C}) = 1]\ |$

$\underline{\mathcal{C}^{O}\text{:}}$
IV <-\$ $\{0,1\}^n$
P = $\text{trunc}_L(\ O(IV \oplus 1) \,\|\, \cdots \,\|\, O(IV \oplus m)\ )$
b' <-\$ $\mathcal{B}$(IV, P)
Return b'

$\Pr[\text{PRF1}(G,L,\mathcal{C}) = 1] = \Pr[\text{PRG1}(SE,\mathcal{B}) = 1]$
$\Pr[\text{PRF0}(G,L,\mathcal{C}) = 1] = \Pr[\text{PRG0}(SE,\mathcal{B}) = 1]$

$$\epsilon'' = \epsilon'$$

$\mathcal{C}$ runs in time that of $\mathcal{B}$ plus small overhead
$\mathcal{C}$ makes m queries to O

If G's output is not random-looking, then blockcipher's output not random-looking

Two-step proof game plan

Goal: show break against CTR[E] implies break against E

Split into two steps

Breaking scheme

↓

Breaking stream cipher

↓

Breaking block cipher

Attacker $\mathcal{A}$ violates
(t, ϵ)-indistinguishability security

↓

For any $\mathcal{A}$ need to give $\mathcal{B}$ for which $t' \approx t$ and $\epsilon' \approx \epsilon$

Attacker $\mathcal{B}$ that violates
(t', ϵ')-pseudorandom generator security of G

↓

For any $\mathcal{B}$ need to give C with $t'' \approx t'$, $\epsilon'' \approx \epsilon'$, q smallish

Attacker C that violates
(t'', q, ϵ'')-pseudorandom function security of E

Reduces security analysis task to analyzing block cipher

Confidence in block cipher security gives confidence in scheme's security

Two-step proof game plan

Goal: show break against CTR[E] implies break against E

Split into two steps

Breaking scheme

↓

Breaking stream cipher

↓

Breaking block cipher

Attacker $\mathcal{A}$ violates
(t, ϵ)-indistinguishability security

↓

For any $\mathcal{A}$ need to give $\mathcal{B}$ for which $t' = t$ and $\epsilon' = \epsilon$

Attacker $\mathcal{B}$ that violates
(t', ϵ')-pseudorandom generator security of G

↓

For any $\mathcal{B}$ need to give C with $t'' = t'$, $\epsilon'' = \epsilon'$, $q = m$

Attacker C that violates
(t'', q, ϵ'')-pseudorandom function security of E

Reduces security analysis task to analyzing block cipher

Confidence in block cipher security gives
confidence in scheme's security

Blockcipher History

- DES (under name Lucifer) designed by IBM in 1970s
- NIST standardized it
  - NSA evaluated it and made suggested changes to shorten key length to 56 bits and other slight changes
  - Many public criticisms of these changes, though some changes actually strengthened DES against differential cryptanalysis
- AES competition run by NIST (1997-2000)
  - Many good submissions (15 total submissions)
  - Rijndael (Rijmen & Daemon) chosen as winner

Data encryption standard (DES)

Originally called Lucifer
- team at IBM
- input from NSA
- standardized by NIST in 1976

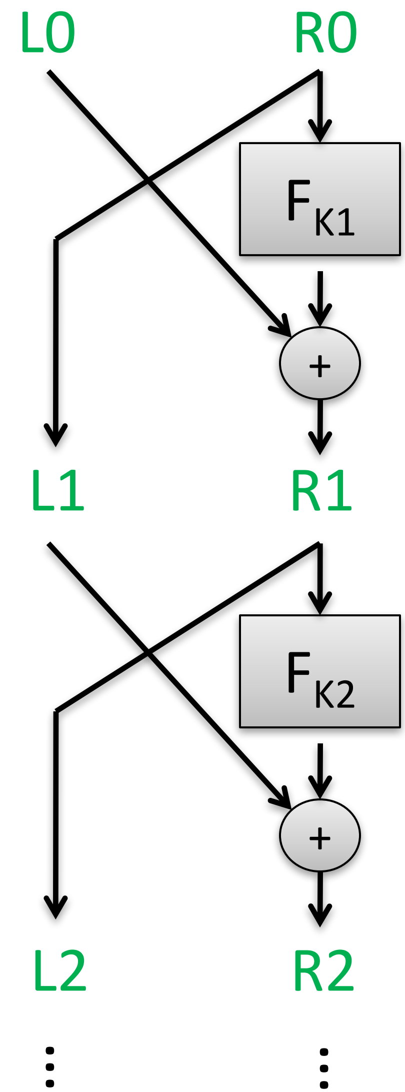
$n = 64$
$k = 56$

Number of keys:
72,057,594,037,927,936

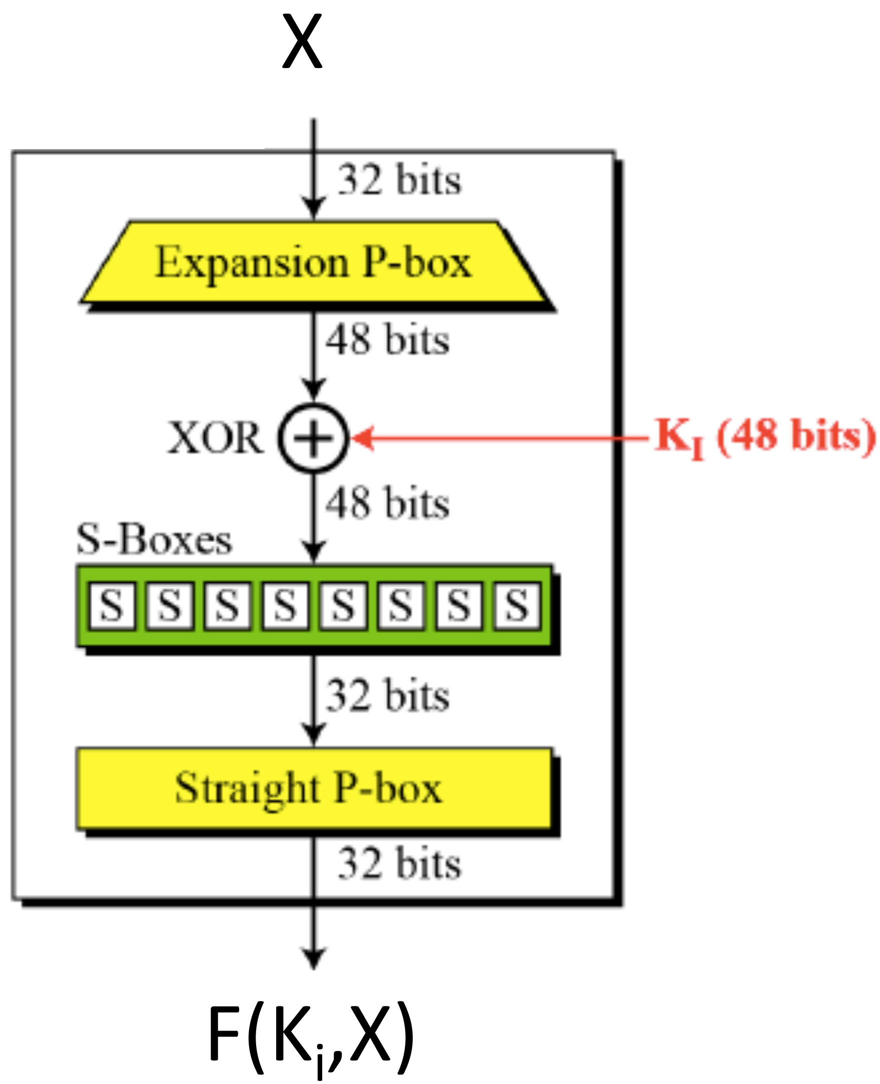
Split 64-bit input into L0,R0 of 32 bits each

Repeat Feistel round 16 times

Each round applies function F using separate round key



Round functions in DES



S boxes stands for "Substitution boxes"

Each S-box a look-up table
6-bit input, 4 bit output

Input	Output
000000	1110
000001	0100
…	
111111	1101

$2^6 = 64$ rows

Attacking DES with brute-force

Attacker given $C = DES_K(M)$ for some known M

How can attacker recover K?

```
BruteForceAttack(M,C):
For  i = 1  to  2^56 do
      C <-  DES_K[i](M)
      If C = M then Return K[i]
```

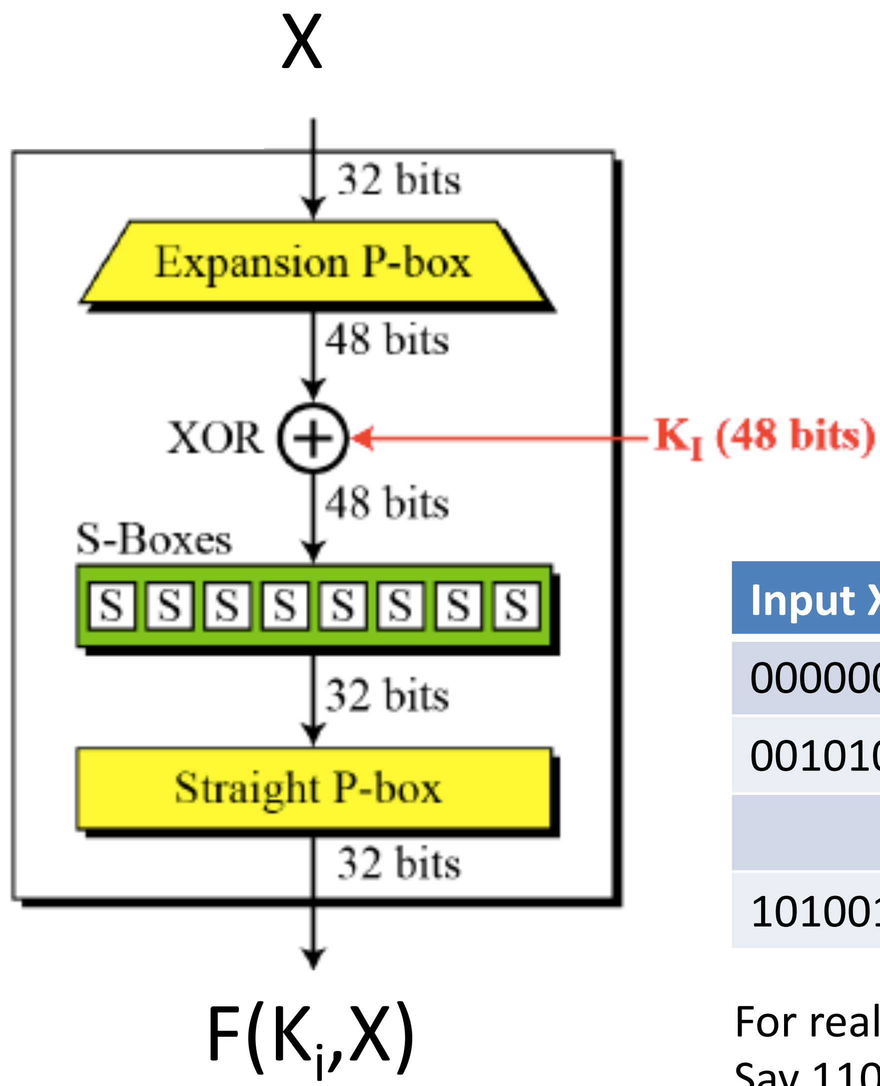
Small chance that we get “false positive”:
$K[i] \neq K$ s.t. $DES_{K[i]}(M) = C$

Low probability event.
Can reduce probability further if given multiple input-output examples

Lesson: Security of block cipher never better than # of possible keys 2^k

Cryptanalysis tries to give attacks much faster than 2^k

Differential cryptanalysis



Idea: find non-uniform behavior of S-boxes under two different inputs

Specific input pairs X1,X2 s.t.

$X1 \oplus X2 = X^*$

$S(X1) = Y1 \quad S(X2) = Y2$

What values $Y^* = Y1 \oplus Y2$ can arise?

Input X1	Input X2	$Y^* = Y1 \oplus Y2$
000000	001010	1101
001010	110101	1101
	…	
101001	100111	0000

$2^6 = 64$ rows

For real S-boxes, we find repeat Y* in table

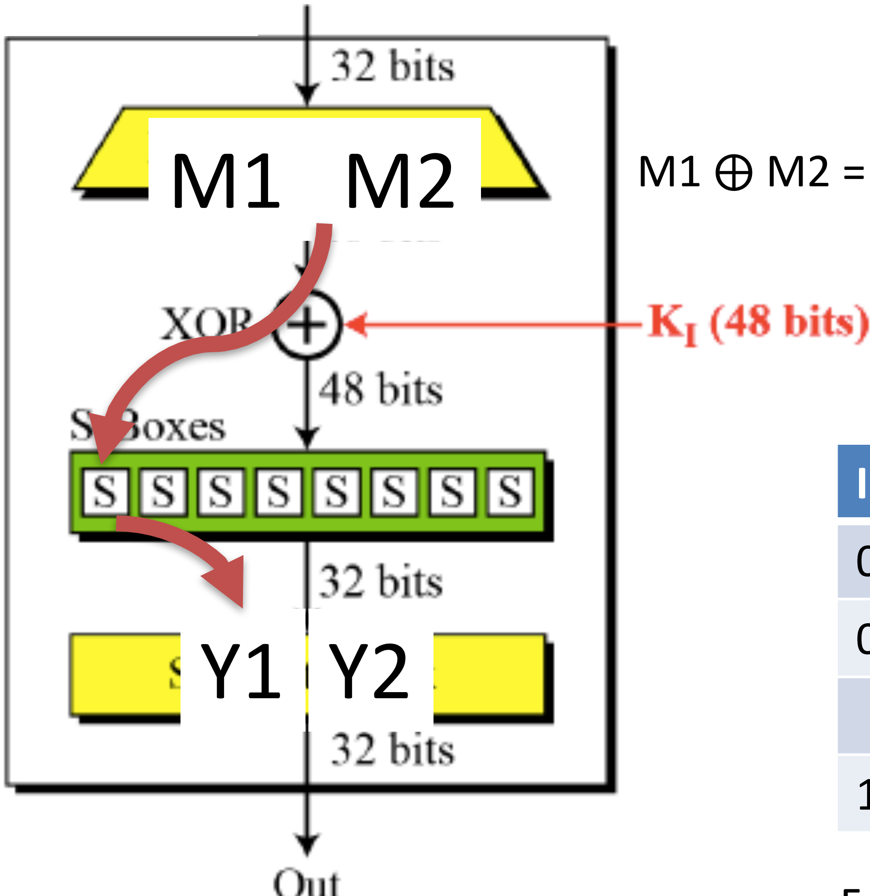
Say 1101 only appears in first two rows

Dif

If Y1 ⊕ Y2 = Y* = 1101, narrows down possibilities for K. Either:

000000 = K ⊕ M1   000000 = K ⊕ M2
001010 = K ⊕ M2   001010 = K ⊕ M1

001010 = K ⊕ M1   001010 = K ⊕ M2
110101 = K ⊕ M2   110101 = K ⊕ M1



M1 ⊕ M2 = X*

Specific input pairs X1,X2 s.t.
X1 ⊕ X2 = X*

S(X1) = Y1   S(X2) = Y2

What values Y* = Y1 ⊕ Y2 can arise?

Input X1	Input X2	Y* = Y1 ⊕ Y2
000000	001010	1101
001010	110101	1101
	…	
101001	100111	0000

$2^6 = 64$ rows

For real S-boxes, we find repeat Y* in table
Say 1101 only appears in first two rows

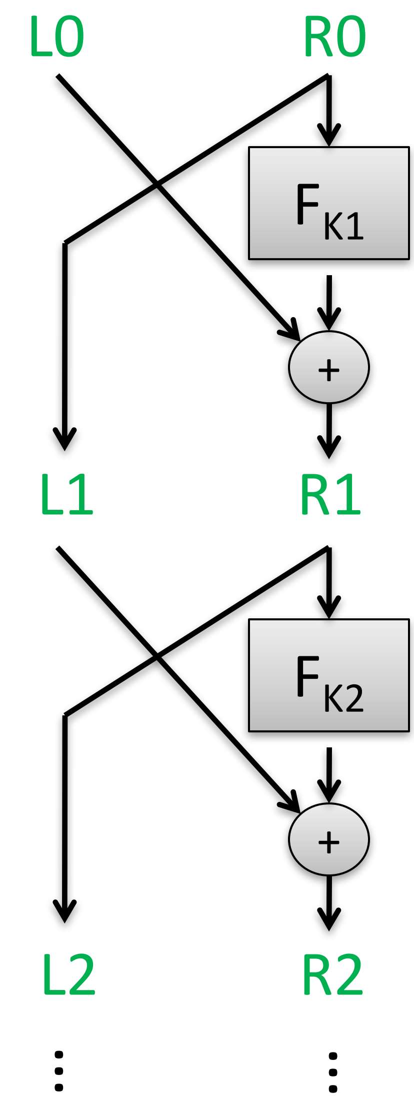
Differential cryptanalysis

Can extend analysis of individual S-box to round function, then to multiple rounds

"Differential trails" track probability of seeing differences at intermediate values

Query many input-output differential pairs to narrow down probable keys

Breaks many weaker ciphers. For DES it is only a theoretical attack, requring 2^{47} pairs



Best attacks against DES

Attack	Attack type	Complexity	Year
Biham, Shamir	Chosen plaintexts, recovers key	2^{47} plaintext, ciphertext pairs	1992
DESCHALL	Brute-force attack	$2^{56/4}$ DES computations 41 days	1997
EFF Deepcrack	Brute-force attack	~4.5 days	1998
Deepcrack + DESCHALL	Brute-force attack	22 hours	1999

- DES is still used in some places
- 3DES (use DES 3 times in a row with more keys) expands keyspace and still used widely in practice

Advanced Encryption Standard (AES)

Rijndael (Rijmen and Daemen)

n = 128

k = 128, 192, 256

Number of keys for k=128:
340,282,366,920,938,463,463,374,607,431,768,211,456

Substitution-permutation design.
For k=128 uses 10 rounds of:

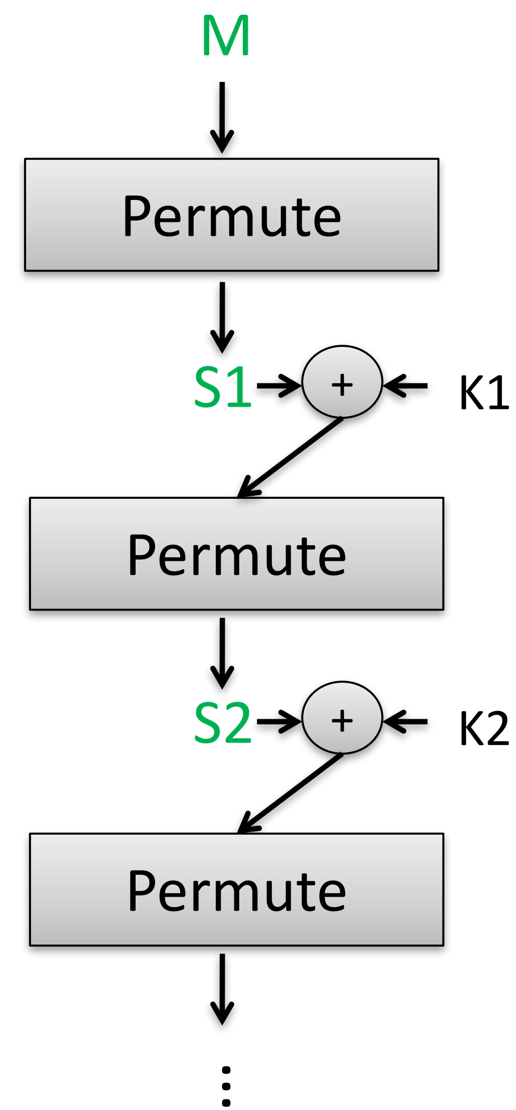
1) Permute:

SubBytes (non-linear S-boxes)

ShiftRows + MixCols (invertible linear transform)

2) XOR in a round key derived from K

(Actually last round skips MixCols)



Best attacks against AES

Brute-force attack (try all keys): worst case time about 2^{128}

Attack	Attack type	Complexity	Year
Bogdanov, Khovratovich, Rechberger	chosen ciphertext, recovers key	$2^{126.1}$ time + some data overheads	2011

No direct attacks of practical interest known

Effective side-channel attacks do exist,
need to implement very carefully

OpenSSL (underlying cryptography.io) does pretty good job

Recap

- Can formally reduce CTR mode security to block cipher security
- Block ciphers
  - DES is based on Feistel network
  - AES based on substitution-permutation network
  - Confidence in blockcipher security via cryptanalysis
    - Differential cryptanalysis widely used tool
    - Modern ciphers (including AES) designed to withstand differential cryptanalysis