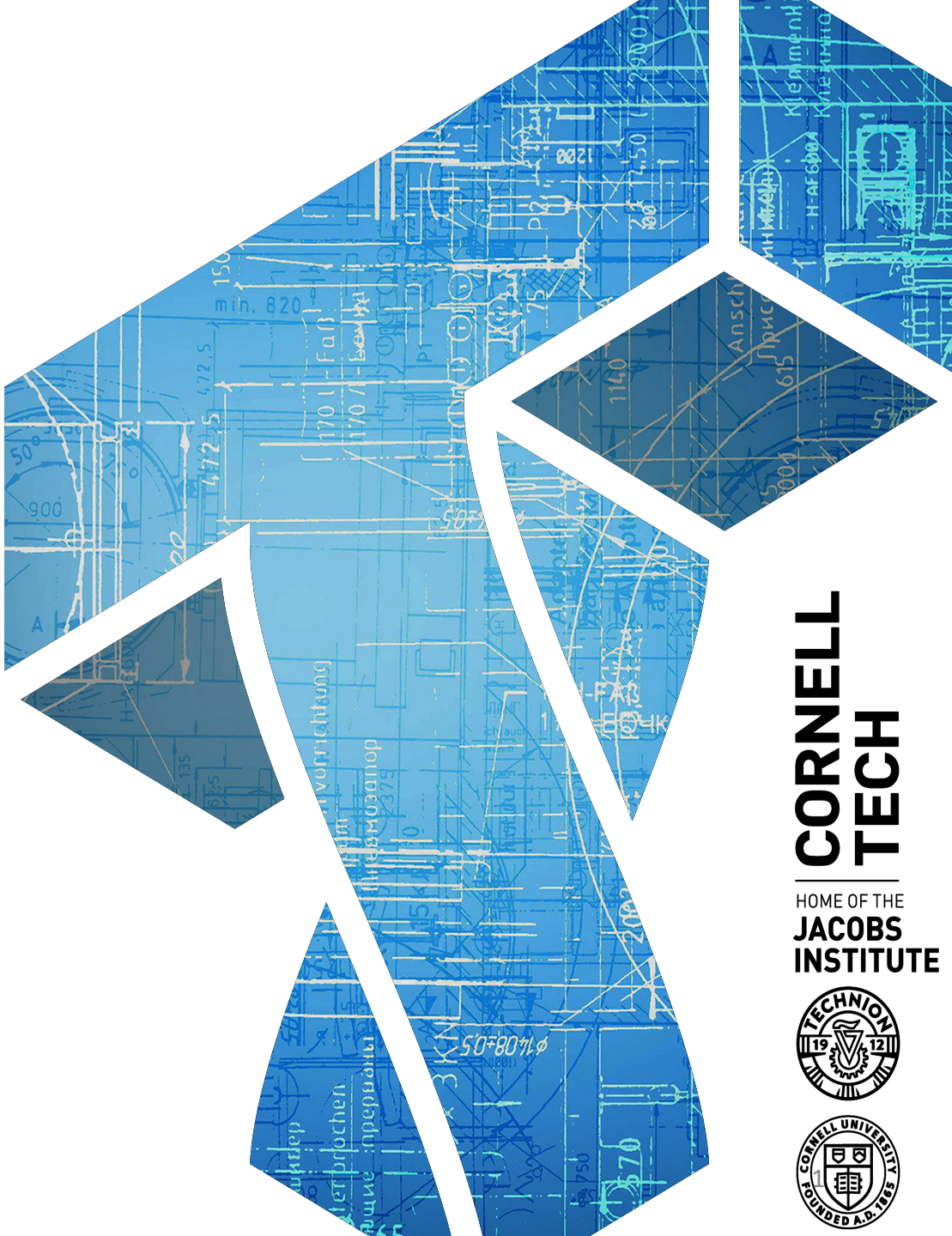
CS 5830
Cryptography

Instructor: Tom Ristenpart
TAs: Yan Ji, Sanketh Menda



CORNELL TECH
HOME OF THE JACOBS INSTITUTE

Recap and where we're at

- Can build stream ciphers from blockciphers
  - CTR mode encryption
- Blockciphers and their security goals
- Today: blockcipher design

Block ciphers built in practice

- Block ciphers workhorse of symmetric encryption
- Practical ones designed and analyzed via ***cryptanalysis***
  - Large number of techniques and design principles
  - Large community (mostly outside USA)
- Historically: closed door design
- Now: ciphers designed through public competition, analysis

Data encryption standard (DES)

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

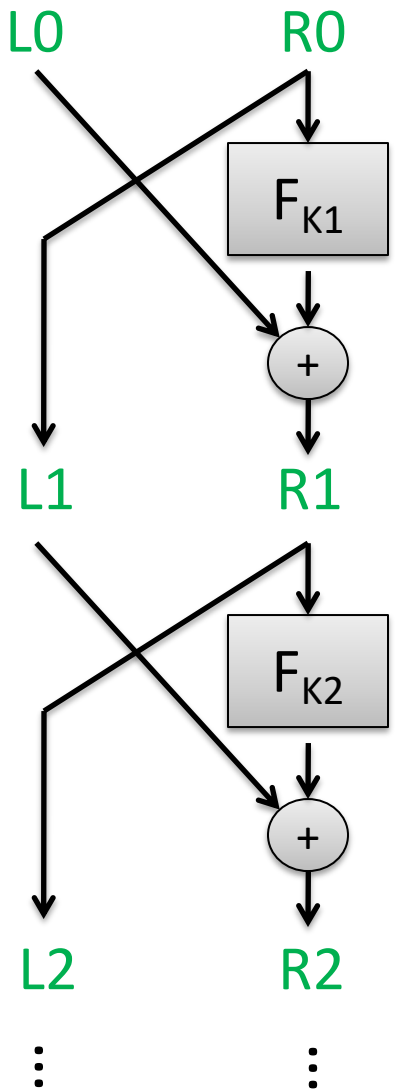
n = 64
k = 56

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



(Omitting initial and final permutations)

Key schedule expands 56-bit key into 16 48-bit round keys

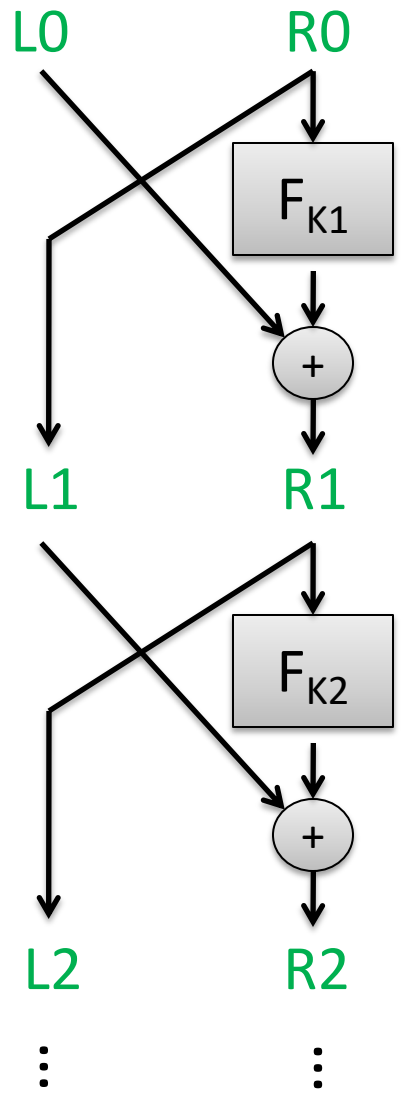
Feistel functions: intuition

Converts any function into a permutation

If we repeat enough Feistel rounds and F is a secure PRF, then block cipher is secure PRF

How many rounds do we need?

Luby-Rackoff proved that 3 rounds suffice for chosen-plaintext attacks, assuming at most $2^{n/4}$ uses under same secret key



The Birthday Bound

Throw q balls randomly into N bins. Let Coll(N,q) be event that some bin has (at least) two balls. Then:

$$\frac{0.3q(q-1)}{N} \leq \Pr[\text{Coll}(2^n,q)] \leq \frac{q(q-1)}{2N}$$

Proof of upper bound. Let Coll_i be event that collision occurs when throwing ball i

$$\Pr[\text{Coll}(2^n,q)] = \Pr[\bigvee_{i=1}^{q} \text{Coll}_i] \leq \Pr[\text{Coll}_1] + \Pr[\text{Coll}_2] + \ldots + \Pr[\text{Coll}_q]$$

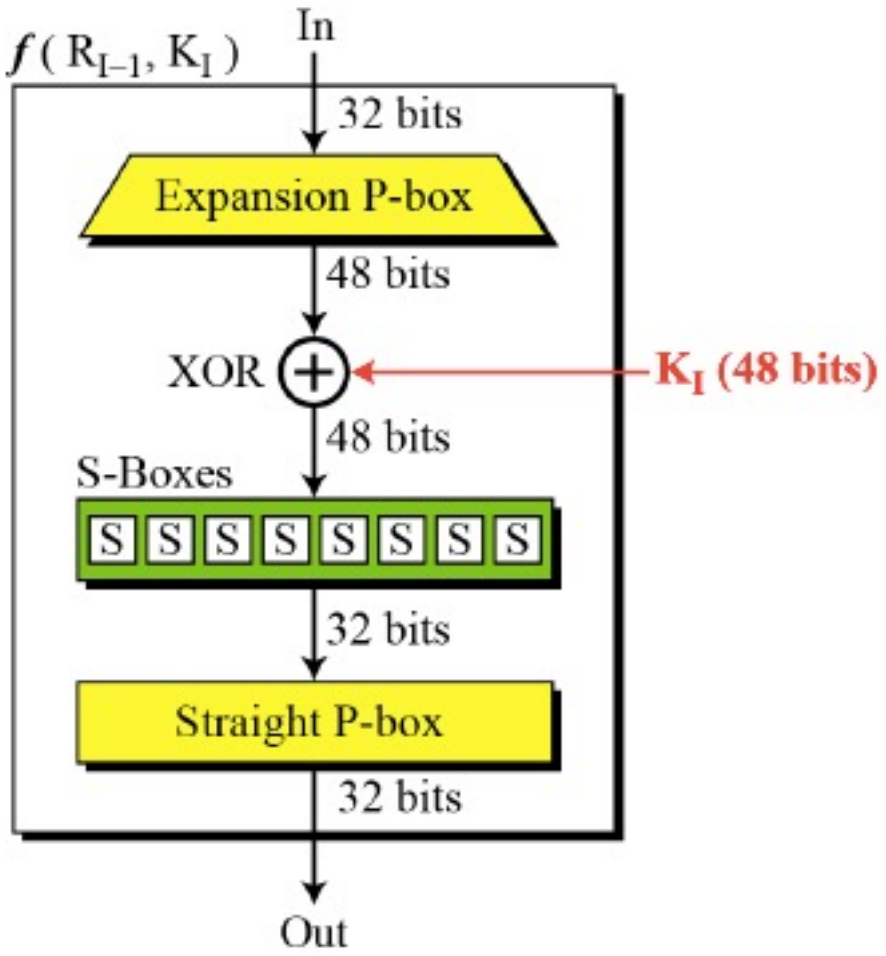
$$\leq 0/N + 1/N + \ldots + (q-1)/N$$

$$= q(q-1)/2N$$

See Appendix A of: https://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf

DES round functions

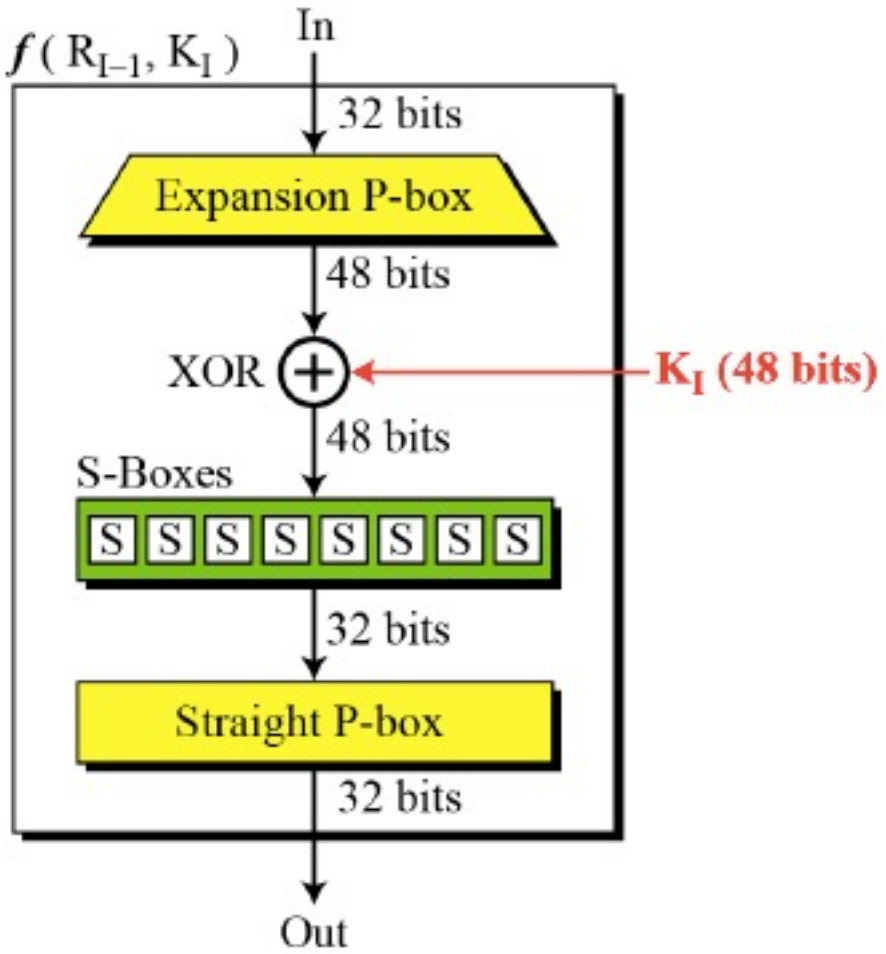
- P-box expands 32 bits to 48 bits and permutes
  - Copies over most bits, duplicates some
- 8 S-Boxes:
  - Each a 6-bit to 4-bit lookup table
- XOR in round key
  - 16 48-bit round keys derived via key schedule from 56 bit key deterministically
- How S-boxes chosen? Why particular permutations?
  - ***Confusion***: each output bit should depend on every bit of the key
  - ***Diffusion***: each output bit should depend on every input bit
  - Related term: ***avalanche effect***



Linear cryptanalysis

[Matsui 1993]

- Approximate S-box behavior by linear functions, e.g.,:
  - X1 + X2 + X6 = Y1 + Y2 + Y4
- S-boxes exhibit some biases, meaning some linear functions are satisfied with higher probability (over uniform inputs)
- Can carefully "pile up" linear approximations across multiple DES rounds
- Use linear approximations to efficiently narrow down search for key, given lots of encryptions of uniform messages



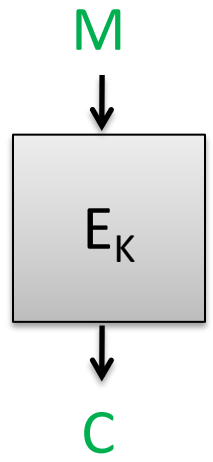
http://www.cs.bc.edu/~straubin/crypto2017/heys.pdf

Linear cryptanalysis

[Matsui 1993]

Approximate E_K by linear boolean function satisfied w/ probability different from ½ (over choice of random message)

For string X and set S ⊆ {0,…,n-1}, let X[S] = $\bigoplus_{i \in S} X_i$



Suppose we know S_m , S_c , S_k such that

$$\Pr\left[\, M[S_m] \oplus C[S_c] = K[S_k] \,\right] = \frac{1}{2} + \epsilon$$

where $C = E_K(M)$ and M chosen uniformly

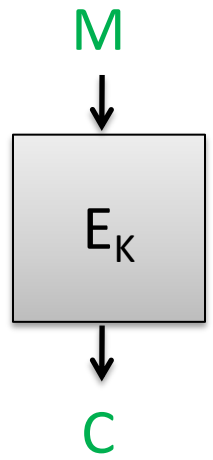
Then this allows recovering 1 bit of K using ***known plaintext attack***

Linear cryptanalysis: recovering one bit

$$\Pr\left[\, M[S_m] \oplus C[S_c] = K[S_k] \,\right] = \frac{1}{2} + \epsilon \qquad (1)$$

Let M_1 , … , M_q be uniform messages, $C_i = E_K(M_i)$

Let $K[S_k] = \text{Maj}(\{ M_i[S_m] \oplus C_i[S_c] \}_i)$



Theorem 1 *Let $\mathcal{E}$ be a cipher such that (1) holds with $\epsilon > 0$, and let $K \in \mathcal{K}$. Let $M_1, \ldots, M_q$ be sampled uniformly from $\{0,1\}^n$ and let $C_i = E_K(M_i)$ for $i \in \{1, \ldots, q\}$. Then*

$$\Pr\left[\, K[S_k] = \text{Maj}\left(\{M_i[S_m] \oplus C_i[S_c]\}_{i=1}^{q}\right) \,\right] \geq 1 - e^{-q\epsilon^2/2} \,.$$

Chernoff bound

Finding linear approximations

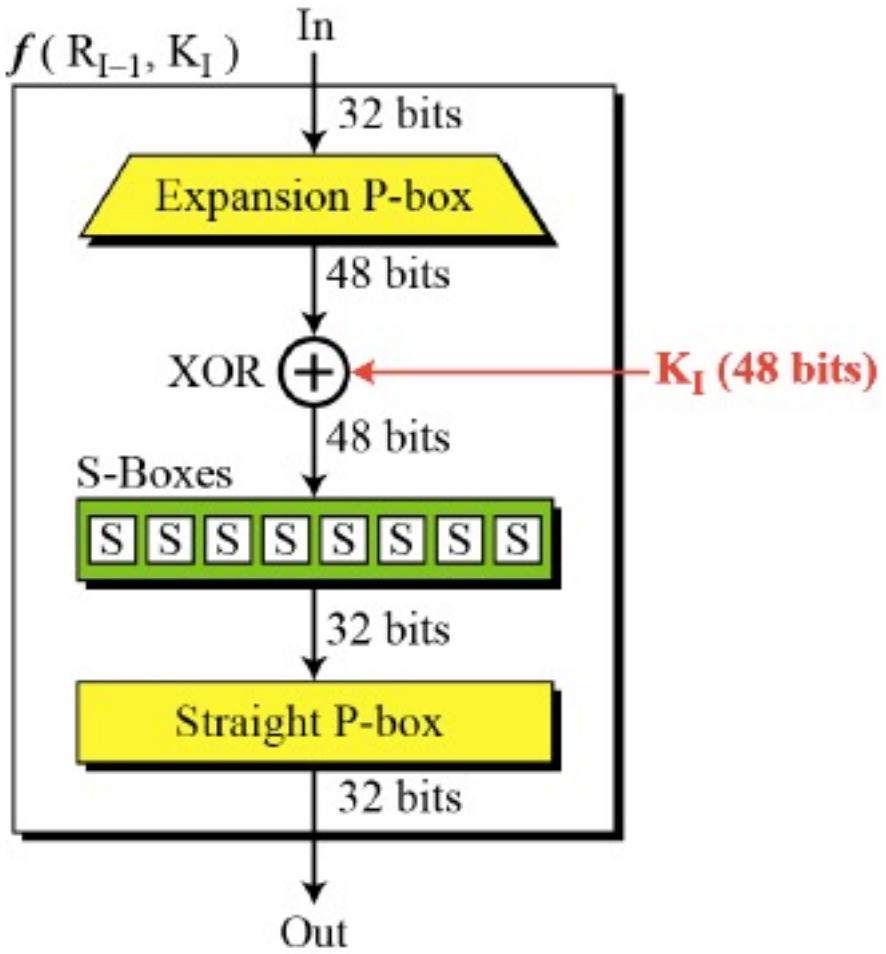
$$\Pr\left[\, M[S_m] \oplus C[S_c] = K[S_k] \,\right] = \frac{1}{2} + \epsilon$$

Need to determine a good linear approximation of cipher

For DES, only non-linearity is due to S-boxes

	S[5]															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S[5] : $(x_0, x_1, x_2, x_3, x_4, x_5) \rightarrow (y_0, y_1, y_2, y_3)$
$(1, 0, 1, 0, 0, 0)$: row 2, column 4, $S[5](1, 0, 1, 0, 0, 0) = 2 = (0, 0, 1, 0)$



Can we find linear approximation of S-Box?

Finding linear approximations

$$\Pr\left[\, X[S_x] \oplus \mathrm{Sbox}(X)[S_y] = 0 \,\right] = \frac{1}{2} + \epsilon$$

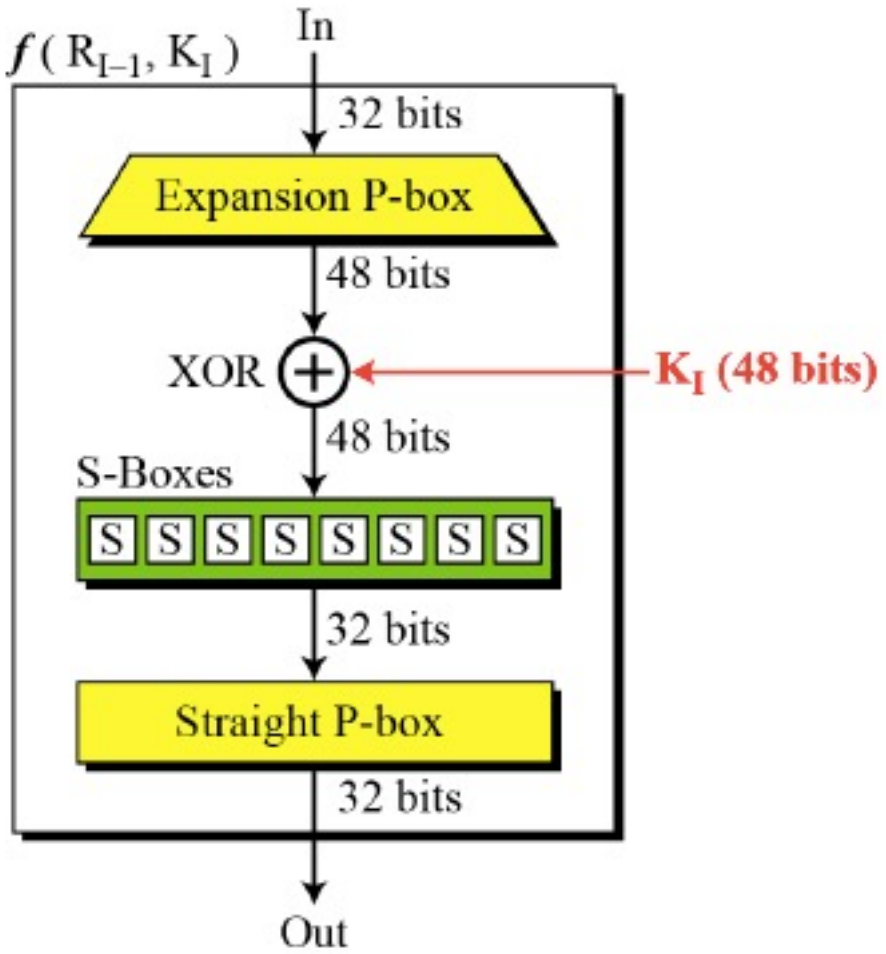
Can we find linear approximation of an SBox?

Number of possible S_x , S_y ? 64 x 16

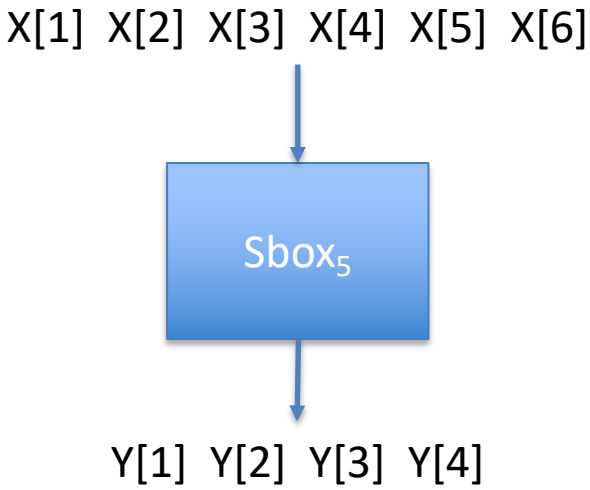
Directly compute bias of each possible S_x , S_y

# of X for which $X[S_x] \oplus \mathrm{Sbox}(X)[S_y] = 0$

Subtract 32 to get bias



S_x \ S_y	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	4	-2	2	-2	2	-4	0	4	0	2	-2	2	-2	0	-4
3	0	-2	6	-2	-2	4	-4	0	0	-2	6	-2	-2	4	-4
4	2	-2	0	0	2	-2	0	0	2	2	4	-4	-2	-2	0
5	2	2	-4	0	10	-6	-4	0	2	-10	0	4	-2	2	4
6	-2	-4	-6	-2	-4	2	0	0	-2	0	-2	-6	-8	2	0
7	2	0	2	-2	8	6	0	-4	6	0	-6	-2	0	-6	-4
8	0	2	6	0	0	-2	-6	-2	2	4	-12	2	6	-4	4
9	-4	6	-2	0	-4	-6	-6	6	-2	0	-4	2	-6	-8	-4
10	4	0	0	-2	-6	2	2	2	2	-2	2	4	-4	-4	0
11	4	4	4	6	2	-2	-2	-2	-2	-2	2	0	-8	-4	0
12	2	0	-2	0	2	4	10	-2	4	-2	-8	-2	4	-6	-4
13	6	0	2	0	-2	4	-10	-2	0	-2	4	-2	8	-6	0
14	-2	-2	0	-2	4	0	2	-2	0	4	2	-4	6	-2	-4
15	-2	-2	8	6	4	0	2	2	4	8	-2	8	-6	2	0
16	2	-2	0	0	-2	-6	-8	0	-2	-2	-4	0	2	10	-20
17	2	-2	0	4	2	-2	-4	4	2	2	0	-8	-6	2	4
18	-2	0	-2	2	-4	-2	-8	4	6	4	6	-2	4	-6	0
19	-6	0	2	-2	4	2	0	4	-6	4	2	-6	4	-2	0
20	4	-4	0	0	0	0	0	-4	-4	4	4	0	4	-4	0
21	4	0	-4	-4	4	-8	-8	0	0	-4	4	8	4	0	4
22	0	6	6	2	-2	4	0	4	0	6	2	2	2	0	0
23	4	-6	-2	6	-2	-4	4	4	-4	-6	2	-2	2	0	4
24	6	0	2	4	-10	-4	2	2	0	-2	0	2	4	-2	-4
25	2	4	-6	0	-2	4	-2	6	8	6	4	10	0	2	-4
26	2	2	-8	-2	4	0	2	-2	0	4	2	0	-2	-2	0
27	2	6	-4	-6	0	0	2	6	8	0	-2	-4	-6	-2	0
28	0	-2	2	4	0	-6	2	-2	6	-4	0	2	-2	0	0
29	4	-2	6	-8	0	-2	2	10	-2	-8	-8	2	2	0	4
30	-4	-8	0	-2	-2	-2	2	-2	2	-2	6	4	4	4	0
31	-4	8	-8	2	-6	-6	-2	-2	2	-2	-2	-8	0	0	-4
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



$$X[5] \oplus \mathrm{Sbox}(X)[1,2,3,4] = 0$$

Holds for for 12 / 64 of inputs X

Bias of 12-32 = -20

Finding linear approximations

$$\Pr[\, X[S_x] \oplus \mathrm{Sbox}(X)[S_y] = 0 \,] = \frac{1}{2} + \epsilon$$

Can we find linear approximation of S-Box?

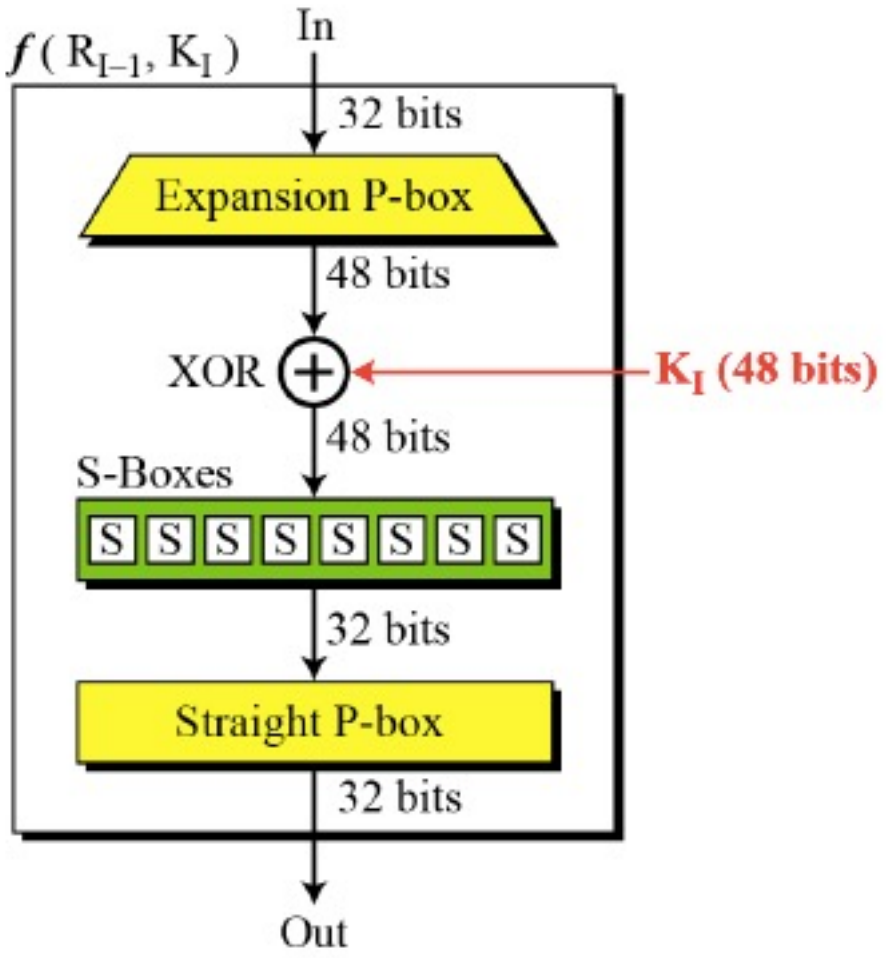
Number of possible S_x , S_y ? 64 x 16

Directly compute bias of each possible S_0 , S_1

# of X for which $X[S_x] \oplus \mathrm{Sbox}(X)[S_y] = 0$ minus 32

Use Sbox approximation within full round function to get ***one-round linear approx***:

$$\Pr[\, X[15] \oplus F_K(X)[7, 18, 24, 29] = K[22] \,] = \frac{1}{2} - \frac{20}{64}$$

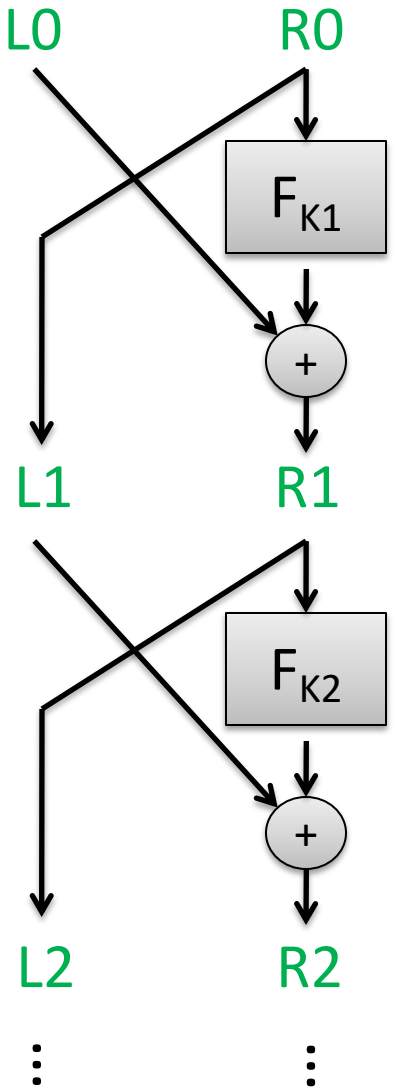


Finding linear approximations

Combine linear approximations for different rounds together, to get function of known values and key bits

Additive combination of different linear approximations

"Piling up" heuristic: calculate bias across multiple rounds by heuristically treating round estimators as independent



Recovering many key bits

Use a linear approximation of 15 rounds. Let Y_{15} = L15 || R15

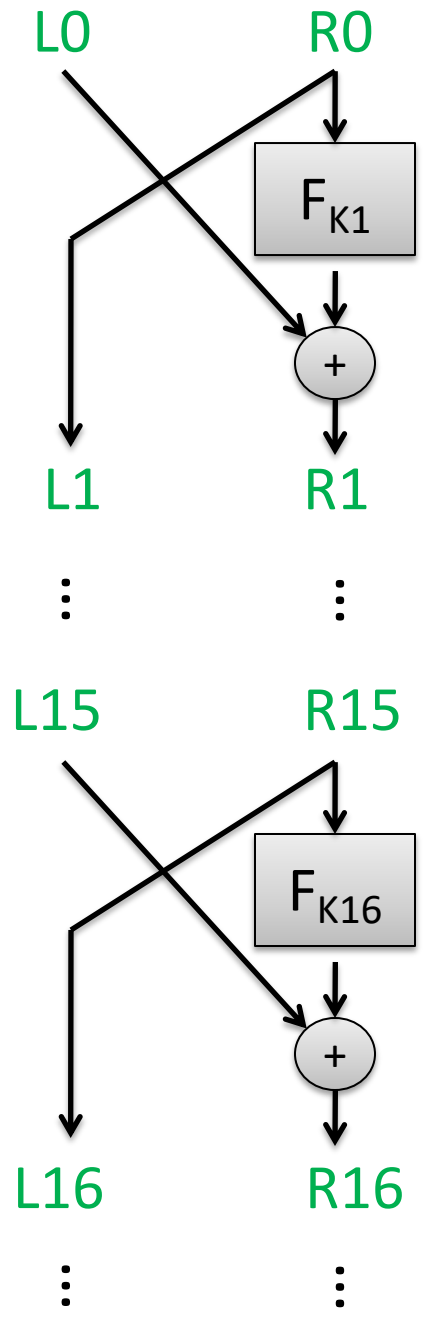
$$\Pr\left[\, M[S_m] \oplus Y_{15}[S_c] = K[S_k] \,\right] = \frac{1}{2} + \epsilon$$

Partially decrypt last round – subset of K16 key bits used

For correct K16 bits, $M[S_m] \oplus Y_{15}[S_c]$ will equal zero for either ½+ε or ½-ε fraction of M, Y_{15} pairs

For incorrect K16 bits, $M[S_m] \oplus Y_{15}[S_c]$ will equal zero for closer to ½ of M, Y_{15} pairs

Gives maximum-likelihood estimator of subset of K16 bits
Similar approach for 1st round, w/ estimator of last r-1 rounds
Combine this approach with brute-force of remaining key bits



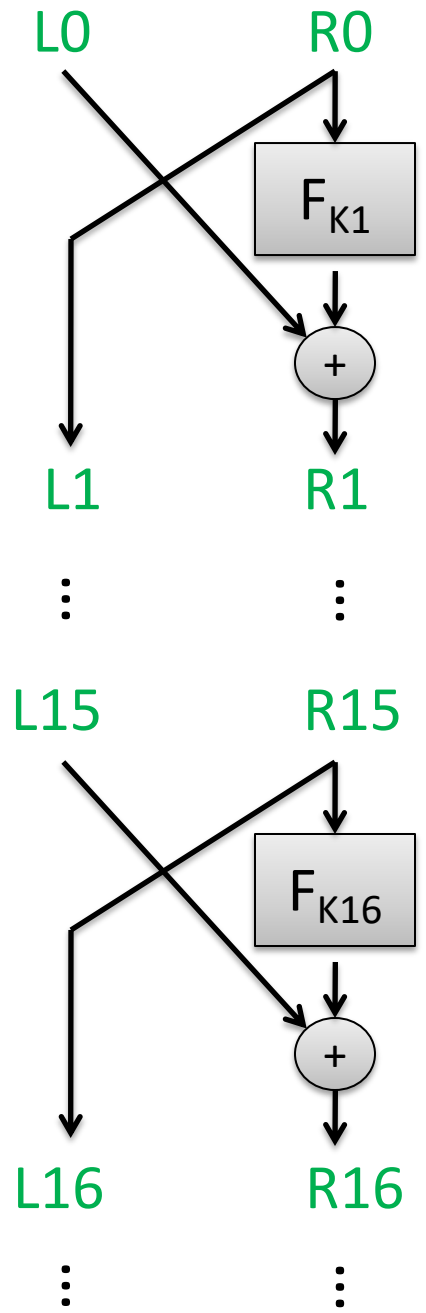
Recovering many key bits

Breaks 8-round DES with:

- 2^{21} known plaintext/ciphertext pairs
- 40 seconds of 1992 computation time (66 Mhz!)

Breaks 16-round DES with:

- 2^{43} known plaintext/ciphertext pairs
- ~2^{41} DES computations in expectation



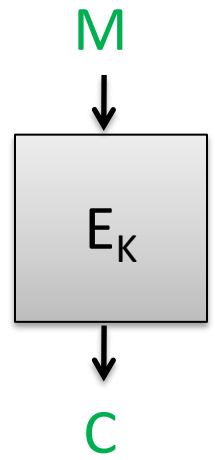
Differential cryptanalysis

[Biham, Shamir 1990]

Look for weaknesses based on pairs of inputs

$$\Delta_C = E_K(M + \Delta_m) + E_K(M)$$

Find Δ_m s.t. Δ_C holds with high probability (over choice of M)



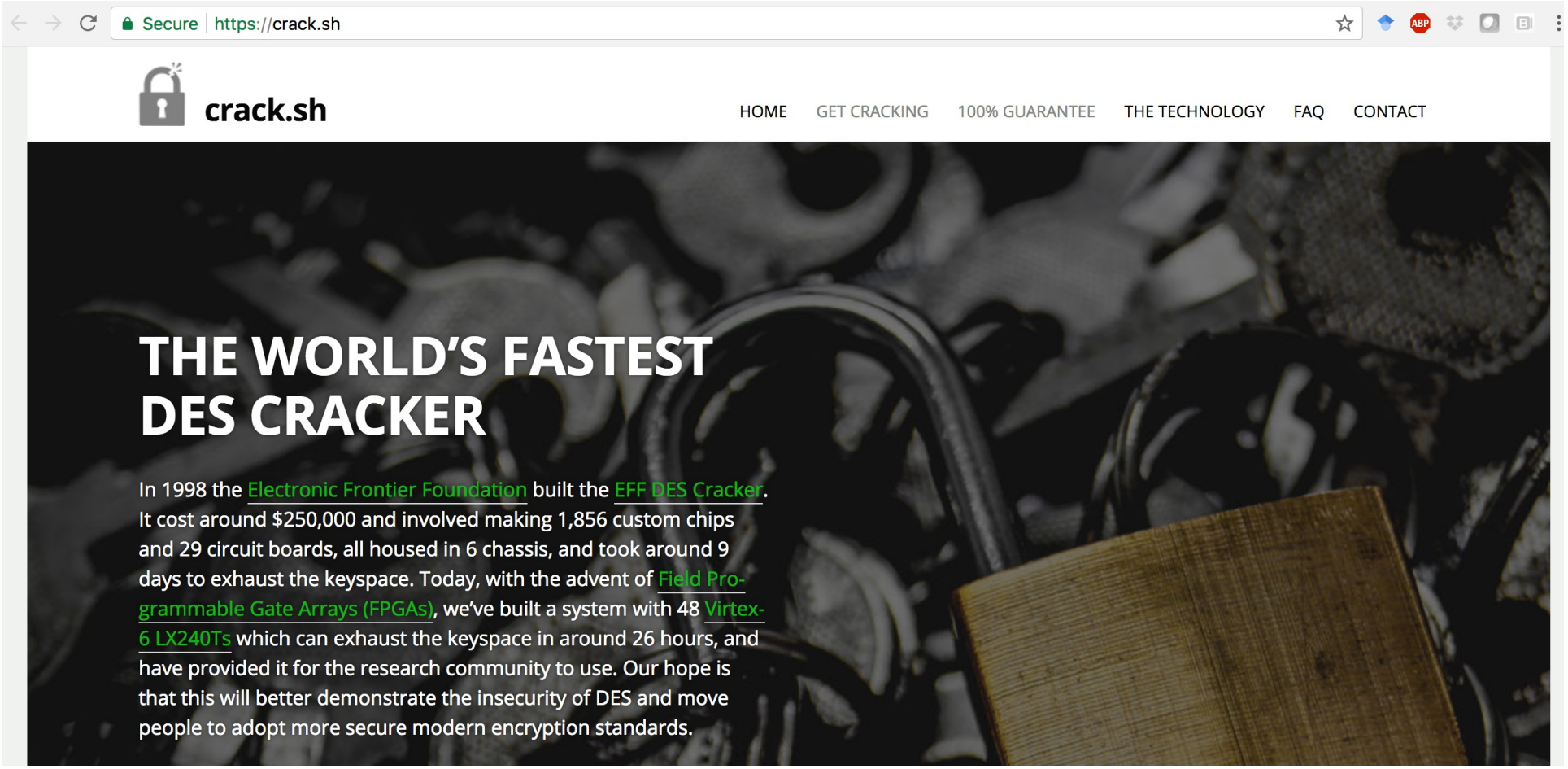
- Analyze S-boxes to find differentials Δ_x, Δ_y such that
  $\Delta_y = Sbox(X + \Delta_x) + Sbox(X)$
  holds with high probability over random choice of X
- Piece together compatible differentials across rounds, providing *differential characteristics*

Leads to ***chosen plaintext attack***

Best attacks against DES

Attack	Attack type	Complexity	Year
Biham, Shamir	Chosen plaintexts, recovers key (differential cryptanalysis)	2^{47} plaintext, ciphertext pairs	1992
Matsui	Known plaintexts, recovers key (linear cryptanalysis)	2^{43} plaintext, ciphertext pairs	1993
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

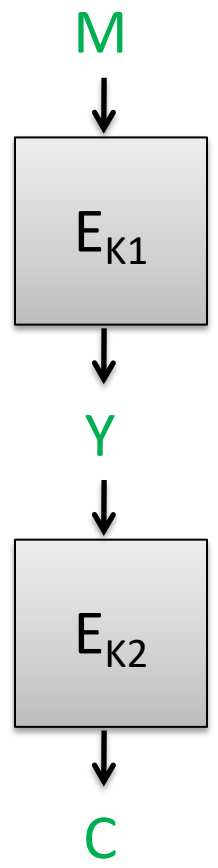
Best practical attack is brute-force key recovery



2DES and 3DES

Increase key length by composing cipher with separate keys

Security not as high as one would expect, due to
meet-in-the-middle attack

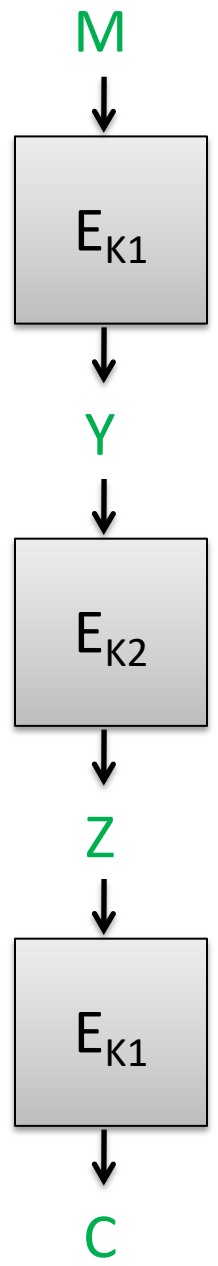


2DES and 3DES

Increase key length by composing cipher with separate keys

Security not as high as one would expect, due to
meet-in-the-middle attack

Triple-DES provides 112-bits of security



AES competition

- 4-year NIST competition to pick new cipher (1997-2000)
- 15 submissions
- 5 finalists
- Rijndael selected as winner
  - Joan Daemen
  - Vincent Rijmen

Advanced Encryption Standard (AES)

A form of key-alternating cipher

n = 128

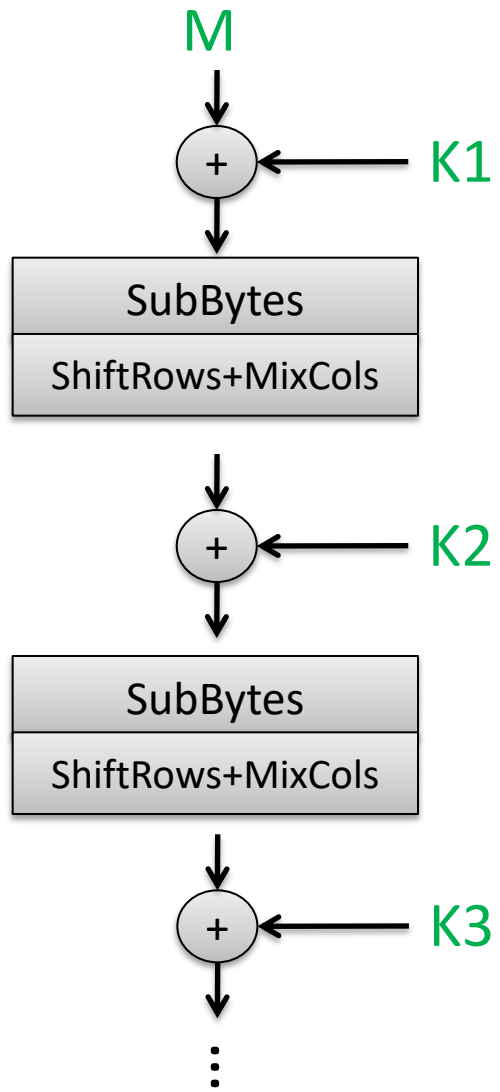
k = 128, 192, 256

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) SubBytes (non-linear 8-bit S-boxes)
2) ShiftRows & MixCols (linear permutation)
3) XOR'ing in a round key

(Last round skips MixCols)

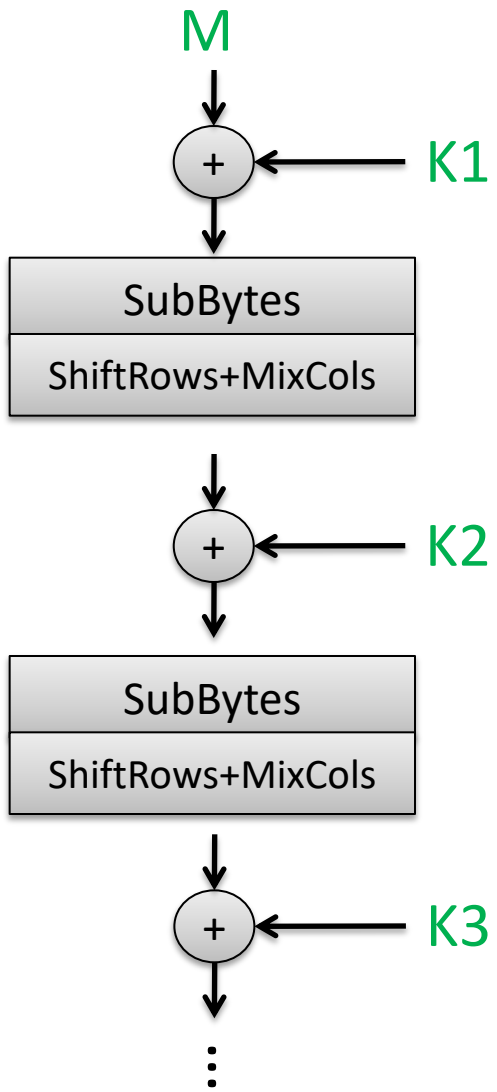


Advanced Encryption Standard (AES)

Designed to resist linear & differential cryptanalysis

"Wide-trail" strategy

- Ensure large # of Sboxes involved in any multi-round trail
- Use coding theory viewpoint to build permutations to ensure rapid ***diffusion***



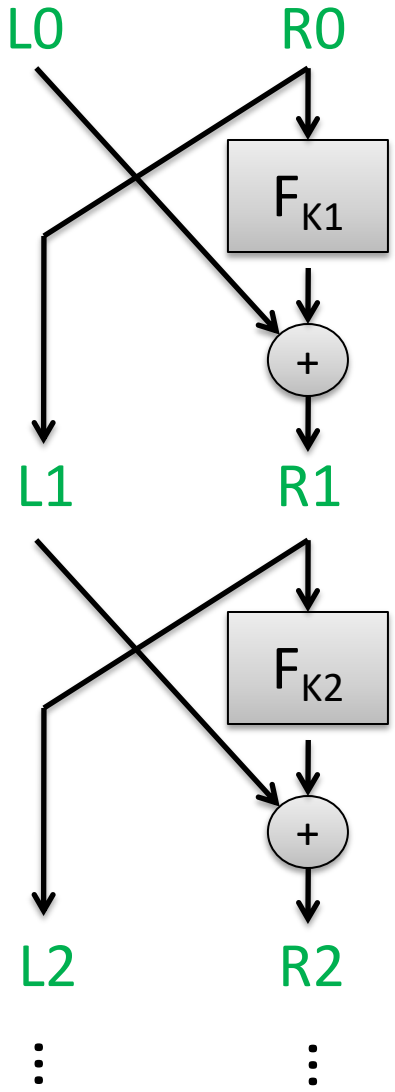
Best attacks against AES

Brute-force attack (try all keys): expected time ~2^{127}

Attack	Attack type	Complexity	Year
Bogdanov, Khovratovich, Rechberger	Key recovery	$2^{126.1}$ time + 2^{88} data	2011
Tao, Wu	Key recovery	$2^{126.01}$ time + 2^{72} data	2015

No algorithmic attacks of practical relevance known

Abstract block cipher designs

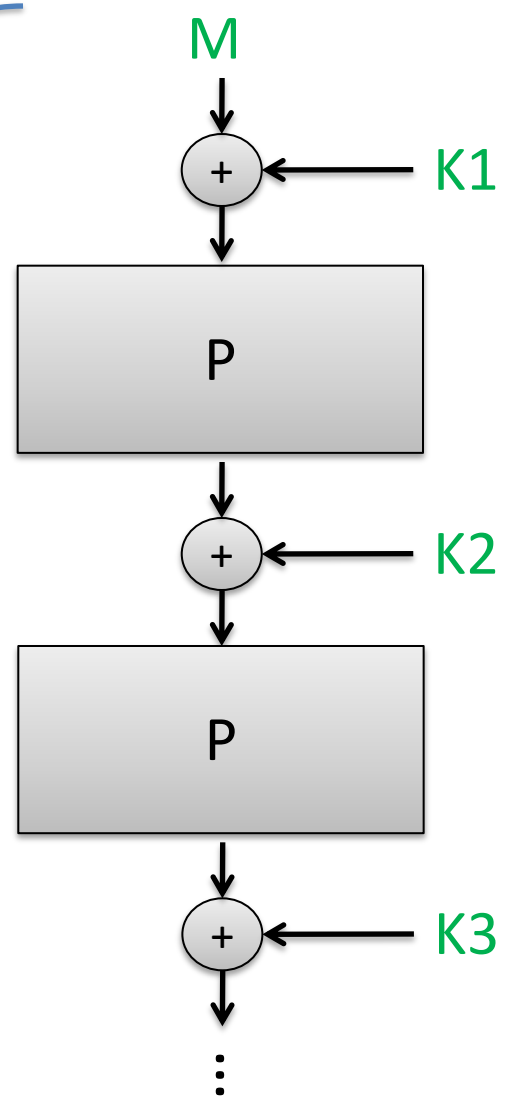


Feistel

- F is a function

Key-alternating ciphers

- P is a permutation
- "Iterated Even-Mansour"



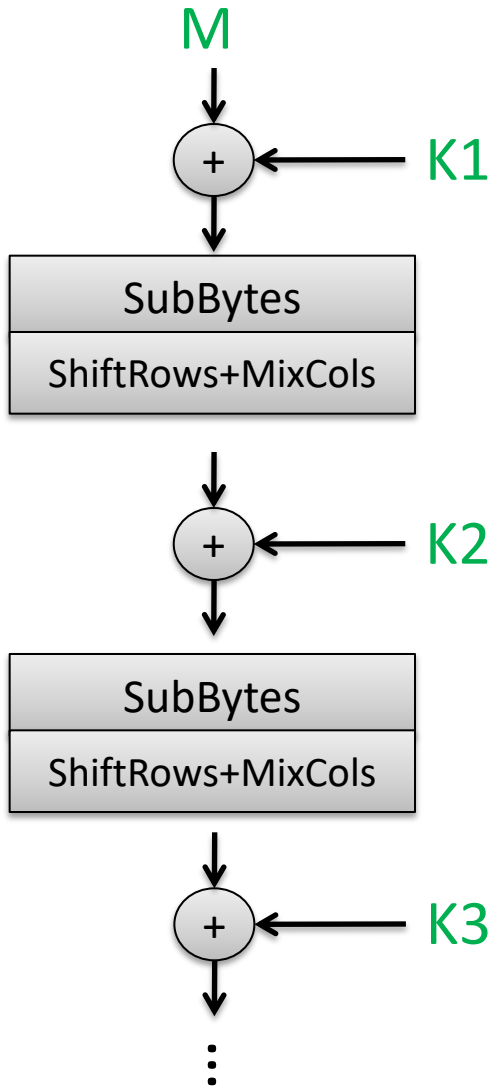
Side-channel attacks

Sbox's are look-up tables

Fastest AES software implementations do complete round via table look-ups of form T[M+K]

What table rows get accessed leaks information about key bits

- Timing (cache hits)
- Micro-architectural side channels



ARX ciphers

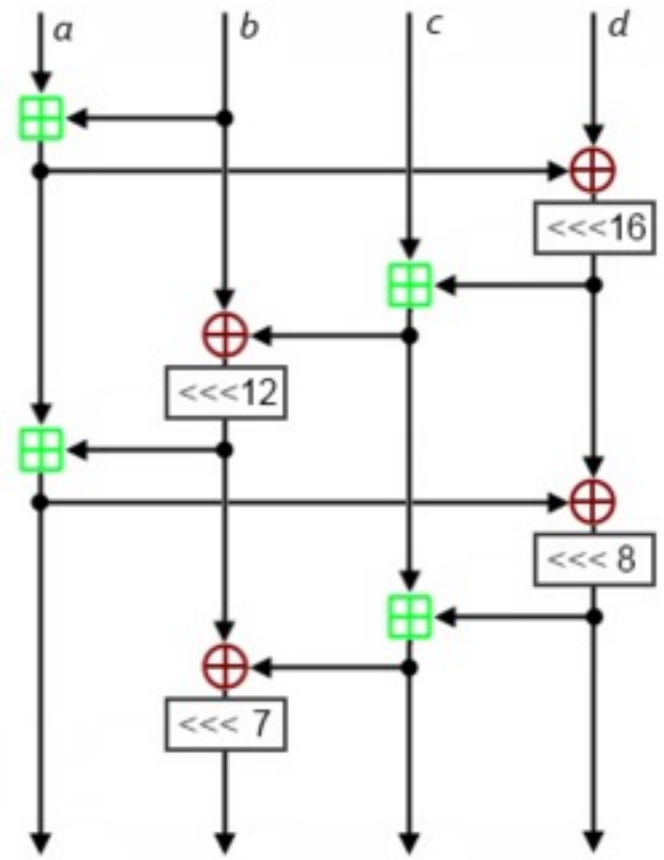
Additions, Rotations, XORs

Combination of operations leads to non-linearity

Easier to implement w/o side-channels

Examples:

- Salsa20 / ChaCha ciphers
- Speck, Simon (from NSA)
- …



ChaCha20 quarter-round function

Moving forward

- Assume good blockciphers that achieve PRF security up to implications of best-known generic attacks
  - ~ 2^k time (exhaustive key search)
  - ~ $2^{n/2}$ time (birthday attacks)