CIS 700 Lecture 4: Pseudorandom permutations and block ciphers

"chosen plaintext attack"

Def A adversary — V verifier

oracle access to Enc_k — $k = Gen(1^n)$

$\xrightarrow{m_0, m_1}$

$\xleftarrow{c \leftarrow Enc_k(m_b)}$ — $b \in \{0,1\}$

oracle access to Enc_k

$\xrightarrow{b'}$ — A succeeds if $b = b'$

- Enc must be randomized. (Why?)

Def "indistinguishable encryptions under a chosen plaintext attack" "CPA-secure"

$\forall$ p.p.t A $\Pr[A \text{ succeeds}] \leq \frac{1}{2} + \varepsilon(n)$

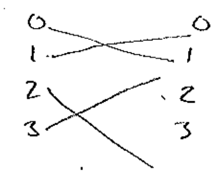
Def "pseudorandom function"

$F: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ efficient length-preserving function

key, input

want truly random function

$x \to \boxed{f} \to f(x)$



not efficient to store

$\forall$ p.p.t. distinguisher D

$|\Pr[D^{F_k(\cdot)}(1^n) = 1] - \Pr[D^{f(\cdot)}(1^n) = 1]| \leq \varepsilon(n)$

oracle access to PRF

oracle access to f truly random function (defined using a lookup table)

Using a PRF to do encryption:

Doesn't work: $Enc_k(m) = F_k(m)$ (Deterministic, not CPA-secure.)

Gen: $k = \{0,1\}^n$ uniformly at random

Enc: choose $r \in \{0,1\}^n$ u.a.r.

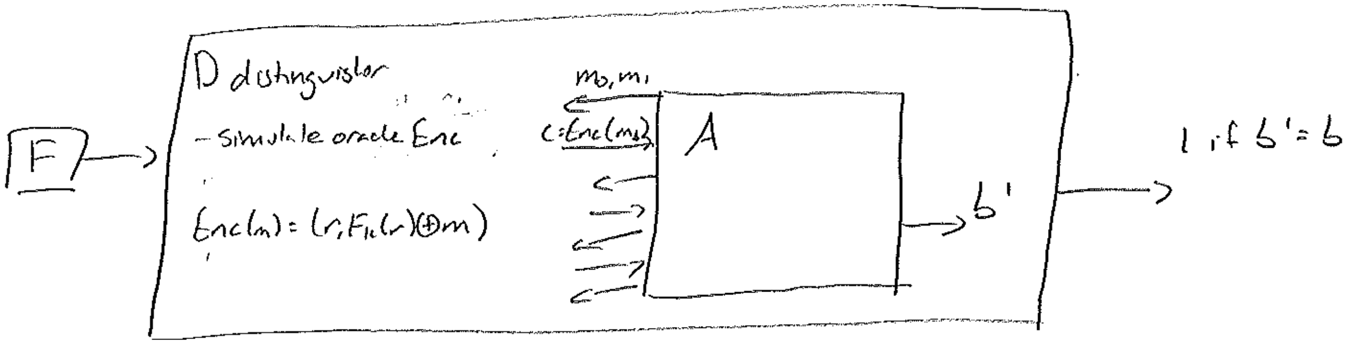
$c = (r, F_k(r) \oplus m)$

Dec: $c = (r, s)$

$m = F_k(r) \oplus s$

Thm F is PRF $\Rightarrow$ construction is CPA-secure

Pf By reduction. Assume A can distinguish encryptions. (Enc is not CPA-secure.) w.p. $\frac{1}{2} + d(n)$, $d(n) >$ negl



If F is true random function f:

A makes $g(n)$ oracle queries

- If r_c used in challenge is repeated, F learns value of $F_k(r_c)$ and succeeds w.p. 1

$$\Pr(r_c \text{ repeated across oracle queries}) \le \frac{g(n)}{2^n}$$

$$\Rightarrow \Pr(A \text{ distinguishes}) \le \frac{1}{2} + \underbrace{\frac{g(n)}{2^n}}_{\text{negligible}}$$

- If r_c not used in challenge, no information leaked.

$$\Pr(\text{success}) = \frac{1}{2}$$

$$\left| \Pr[D^{F_k(\cdot)}(1^n) = 1] - \Pr[D^{f(\cdot)}(1^n) = 1] \right| = \left| \frac{1}{2} + d(n) - \left(\frac{1}{2} + \frac{g(n)}{2^n}\right) \right| = d(n) - \frac{g(n)}{2^n} > \text{negligible}$$

Def "strong pseudorandom permutation"

$F: \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ efficient keyed permutation - one-to-one

$\forall$ p.p.t. distinguishers D: truly random permutations (already indistinguishable from random fn w/ polynomially many queries)

$$\left| \Pr[D^{F_k(\cdot), F_k^{-1}(\cdot)}(1^n) = 1] - \Pr[D^{f(\cdot), f^{-1}(\cdot)}(1^n) = 1] \right| \le \varepsilon(n)$$

(We give access to inverse function too.)

PRPs = abstraction of block ciphers

DES = Data Encryption Standard

IBM, NSA "We sent the S-boxes off to Washington. They came back and were all different."

NSA convinced IBM to reduce keysize from 64 to 56 bits.

FIPS standard in 1977

56-bit key length (already a worry in 1977)

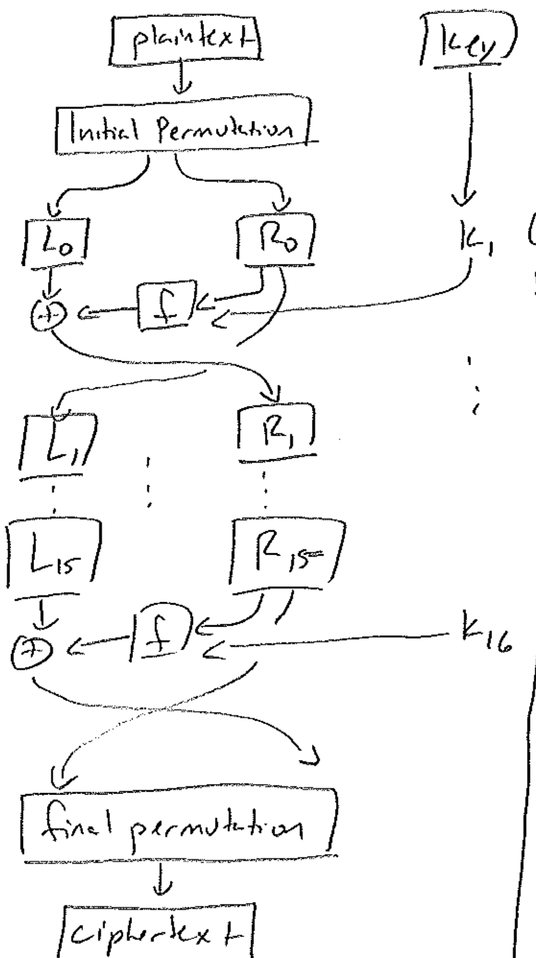
1997 - DES challenges solved by distributed computation

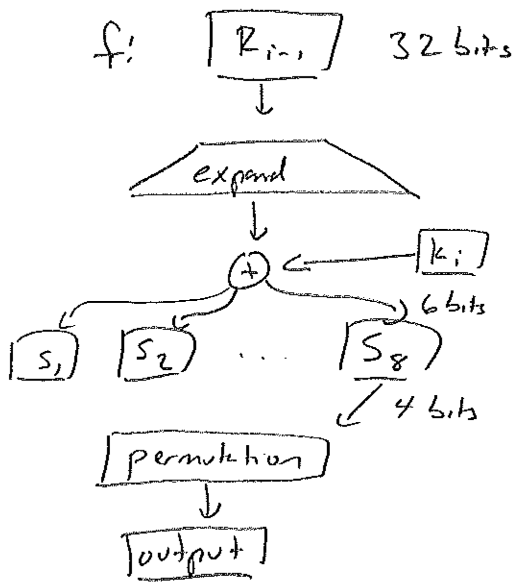
1998 - Deep Crack EFF \$250,000 + 56 hours

DES construction ideal properties "diffusion" - mix input bits together w/ permutations, xoring

"confusion" - nonlinearity s-boxes

Feistel Network





Adapting DES:

- 2DES?

  $F'_{k_1,k_2}(x) = F_{k_2}(F_{k_1}(x))$ (Exercise: Why is this a bad idea?)

- 3DES

  $F'_{k_1,k_2,k_3} = F_{k_3}(F^{-1}_{k_2}(F_{k_1}(x)))$

  why alternate? If $k_1 = k_2 = k_3 \Rightarrow Enc' = Enc$

  with 2 keys:

  $F'_{k_1,k_2} = F_{k_1}(F^{-1}_{k_2}(F_{k_1}(x)))$ popular in finance

- DES-X "whitening"

  $F'_{k_1,k_2,k_3}(x) = k_3 \oplus F_{k_2}(x \oplus k_1)$

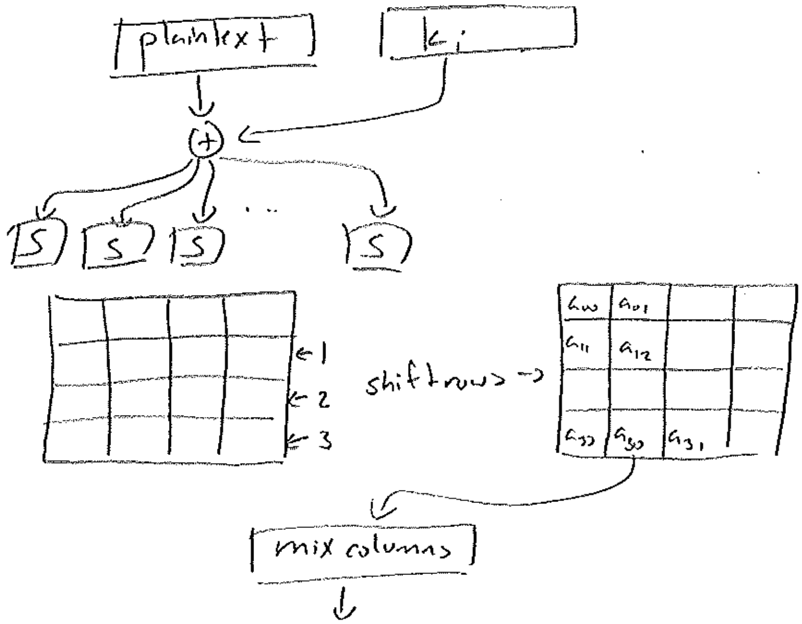
Current standard: AES

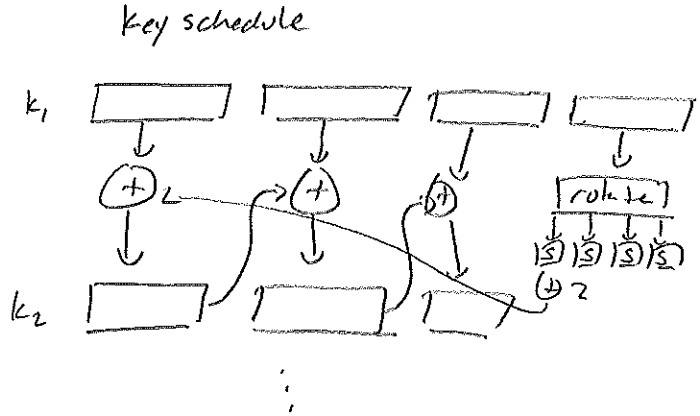
Chosen in 2000 after NIST-run competition Rijndael Joan Daemen, Vincent Rijmen

128, 192, 256-bit versions

also designed keccak

10 rounds





Modes of operation

- how to encrypt an arbitrary-length message with a block cipher

ECB mode "Electronic Code Book"

$m = m_1, m_2, \ldots, m_\ell$

$c = F_k(m_1), F_k(m_2), \ldots, F_k(m_\ell)$

deterministic, messages can be replayed and distinguished

commonly used in practice, should never be used

CTR mode "counter"

Turns a block cipher into a stream cipher

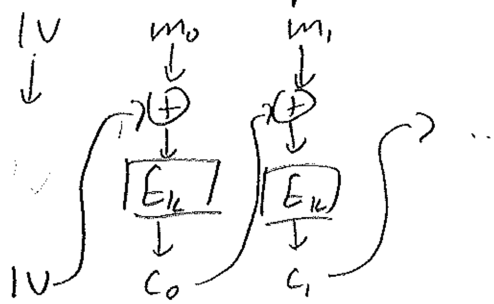
1. Choose random IV ctr.
2. $r_i = F_k(ctr + i)$
3. $c_i = r_i \oplus m_i$

- ctr must be random - why?

CPA-secure, easy to parallelize, allows decryptions of single blocks

subject to problems if randomness failure

CBC mode "Cipher Block Chaining" ← most recommended



1. IV has length n
2. $c_i = F_k(c_{i-1} \oplus m_i)$
3. output $(IV, c_0, c_1, \ldots, c_\ell)$

IV needs to be random

or use E_k(message number) "nonce-generated IV"

CPA-secure