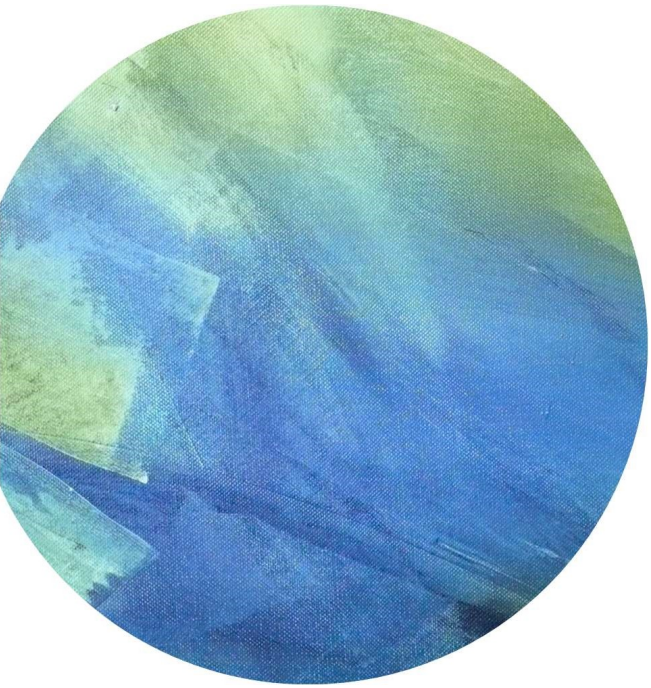
Lecture 3



Introduction

Stream Ciphers: Attacks

Block Ciphers: Definitions

DES

Administrative Details

- Course website: https://courses.grainger.illinois.edu/cs498ac3/fa2020/
- Has syllabus, instructor and TA info, office hours
- IMPORTANT: **Join Piazza!** piazza.com/illinois/fall2020/ececs498ac/home

I strongly encourage class participation.
If you don't understand something in class, please interrupt me and ask questions.
Please make abundant use of office hours.

Stream Ciphers: Examples and Attacks

Symmetric Ciphers

- A cipher defined over $(\mathcal{K}, \mathcal{M}, \mathcal{C})$

  is a pair of efficient algorithms **(E, D)** where

  E: $\mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$

  D: $\mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$

  that satisfies –

  <u>Correctness:</u> For all $m \in \mathcal{M}$, $k \in \mathcal{K}$, **D** (k, **E** (k, m)) = m

  <u>Semantic Security:</u> for all $m_0, m_1 \in \mathcal{M}$: **E** $(k,m_0) \approx_c$ **E** (k,m_1)

computationally bounded Adv $\Longleftrightarrow$ to any PPT Turing Machine

$\Updownarrow$

polynomial-sized circuit

Attack: Integrity

In the PRG-based cipher, $E(k,m) = PRG(k) \oplus m$

easy to convert $E(k,m) \rightarrow E(k,m \oplus 1)$ (mod 2)

$$E(k,m) = PRG(k) \oplus m$$

$$\oplus\, 000\ldots1 \quad = PRG(k) \oplus (m \oplus 0\ldots01)$$

$$= E(k, m \oplus 1)$$

Example: RC4 Cipher (deprecated)

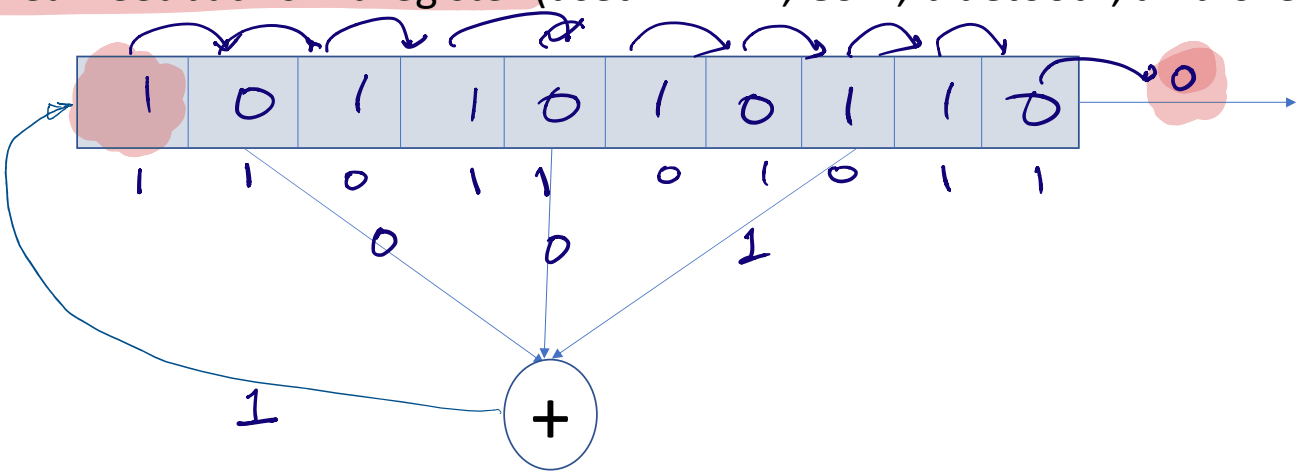
- Expand 128-bit seed to 2048 bits of pseudorandomness
- Use pseudorandomness to initialize internal state
- Used in HTTPS, WEP
- Weaknesses:
  1. Not pseudorandom: e.g. [0,0] appears more often than it should
  2. Related key attacks make it possible to recover the key

https://blog.cryptographyengineering.com/2013/03/12/attack-of-week-rc4-is-kind-of-broken-in/

Content Scramble System

Example: CSS Stream Cipher (badly broken)

- Used to encrypt DVDs. Was designed to be hardware friendly.
- Linear feedback shift register (used in DVD, GSM, bluetooth, all broken)

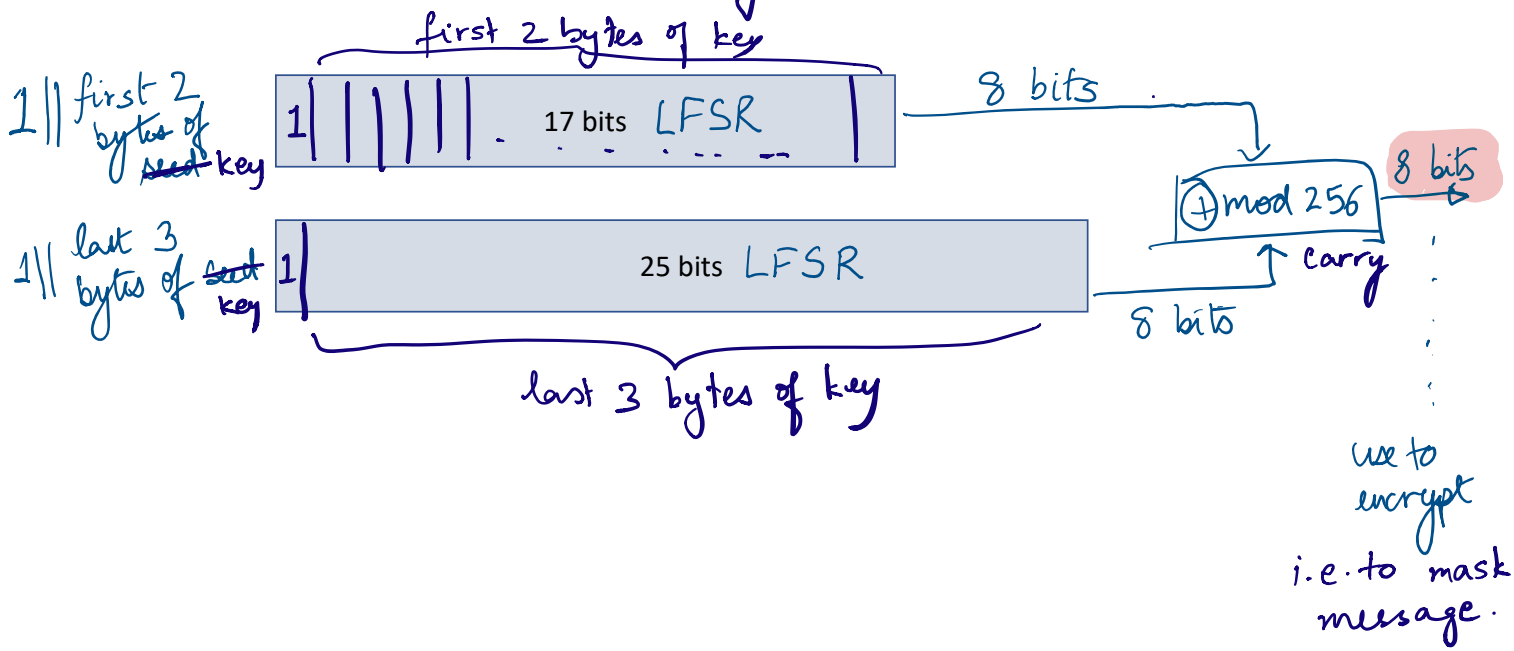


1 byte = 8 bits.

Example: CSS Stream Cipher (badly broken)

- ~~Seed~~ Key size = 5 bytes = 40 bits. = 5 bytes

2^{40} : typically expect

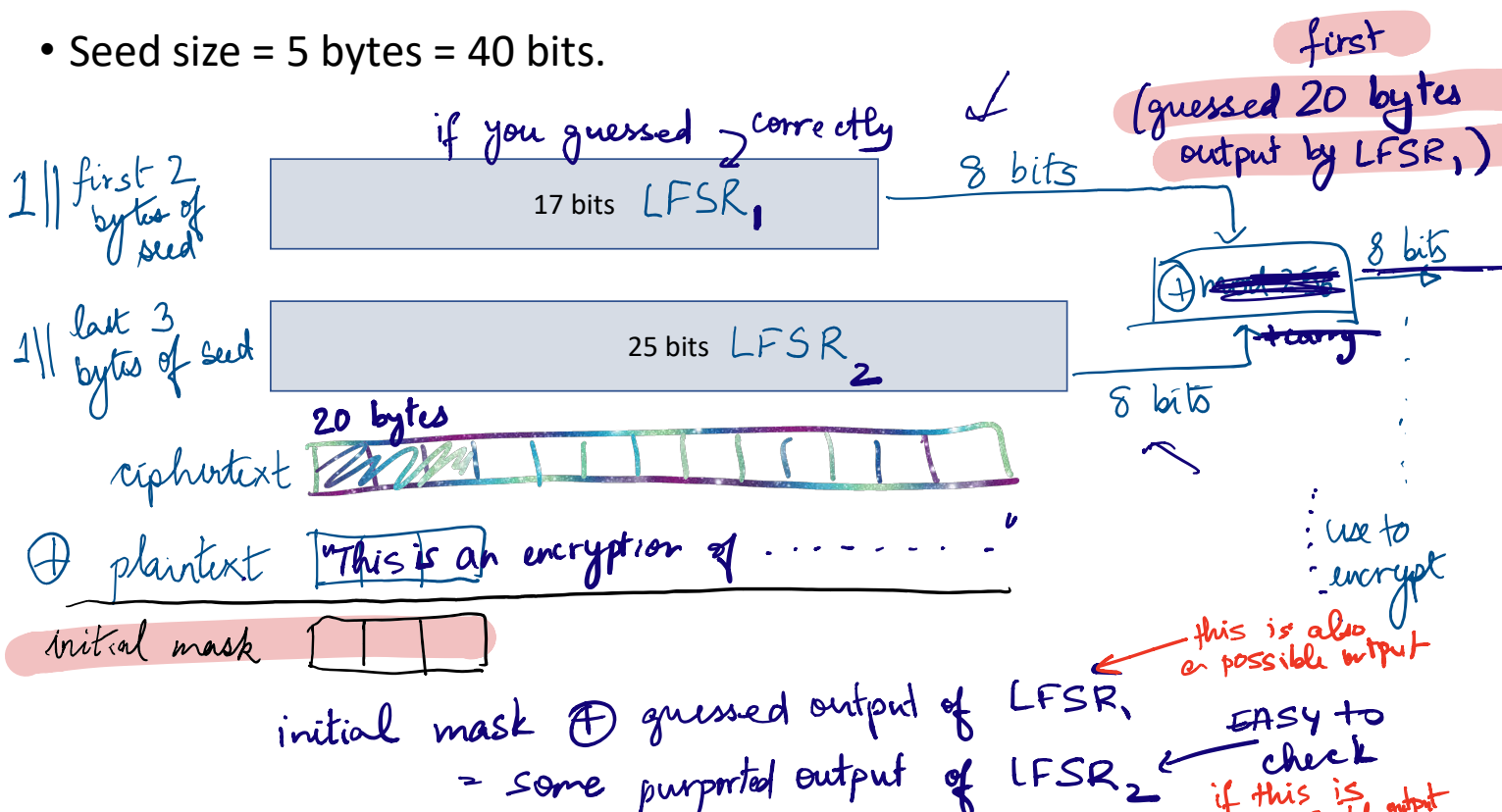


use to encrypt

i.e. to mask message.

Example: CSS Stream Cipher (badly broken)

- Seed size = 5 bytes = 40 bits.



Better Stream Ciphers

- Salsa20/12, Sosemanuk

  Nonce: a non-repeating value for a given key

  $E(k, m ; r) = m \oplus PRG(k ; r)$, r

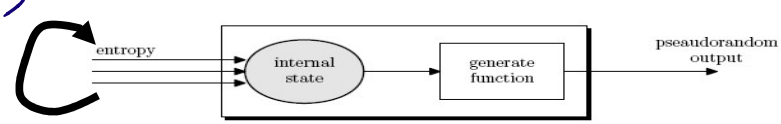
  The pair (k,r) is used at most once

- Pseudo random generators in practice: (e.g. /dev/random)

  Continuously add entropy to internal state

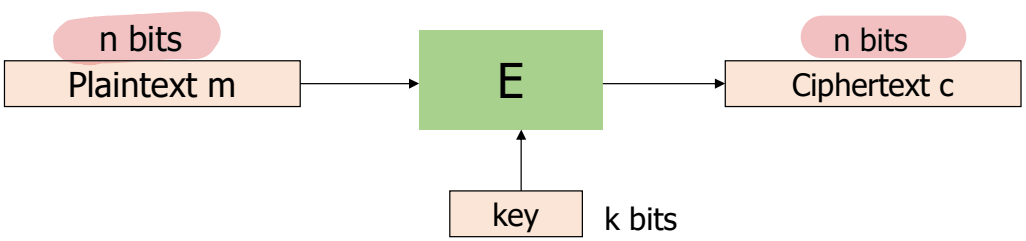
  Entropy sources:

  - Hardware RNG: Intel **RdRand** inst. (Ivy Bridge). 3Gb/sec.
  - Timing: hardware interrupts (keyboard, mouse)



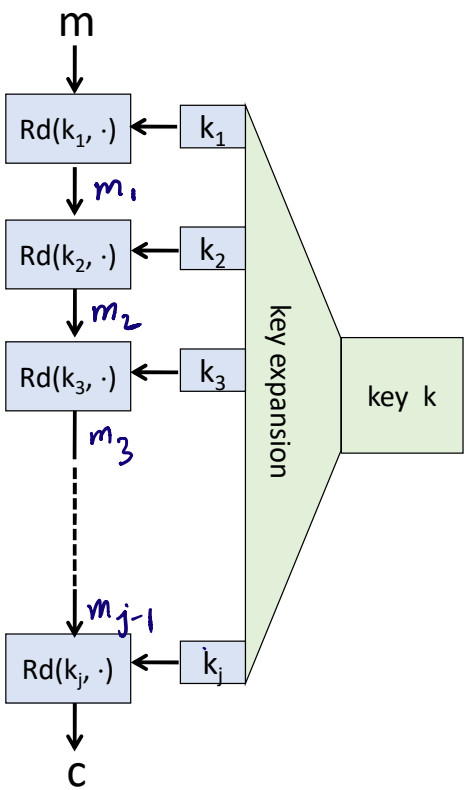
Block Ciphers

Block Ciphers



Examples:

1. 3DES: n= 64 bits, k = 168 bits output size = 64 bits
2. AES: n=128 bits, k = 128, 192, 256 bits output size = 128 bits



Examples:

1. 3DES: k = 168 bits, j = 16, each k_j = 48 bits
2. AES: k = 128/192/256 bits, j = 10, each k_j = 128 bits

$PRG : \mathcal{K} \rightarrow \mathcal{K} \times \mathcal{K}$

Defining Security for Block Ciphers

→ indistg. from a random function

- A **pseudorandom function (PRF)** is a function F from $\mathcal{K} \times \mathcal{X} \rightarrow \mathcal{C}$

  s.t. F (k, m) is efficiently computable for every k and m

- A **pseudorandom permutation (PRP)** is a function F from $\mathcal{K} \times \mathcal{X} \rightarrow \mathcal{X}$

  s.t. F (k, m) is efficiently computable for every k and m, and

  F (k, .) has domain = image and is one-to-one, and

  F^{-1} (k, y) is efficiently computable for every k and y

  where F^{-1} (k, F (k, x)) = x

$F(k, m) = c$

$F^{-1}(k, c) = m$

AES, DES are PRPs. In AES, $|\mathcal{X}| = 2^{128}$ and in DES, $|\mathcal{X}| = 2^{64}$.

permutation : $\{1, 2 \ldots N\} \rightarrow \{1, 2 \ldots N\}$

$1 \rightarrow 2,\ 2 \rightarrow 3,\ 3 \rightarrow 4 \ldots N-1 \rightarrow N,\ N \rightarrow 1$

one-one onto domain = image

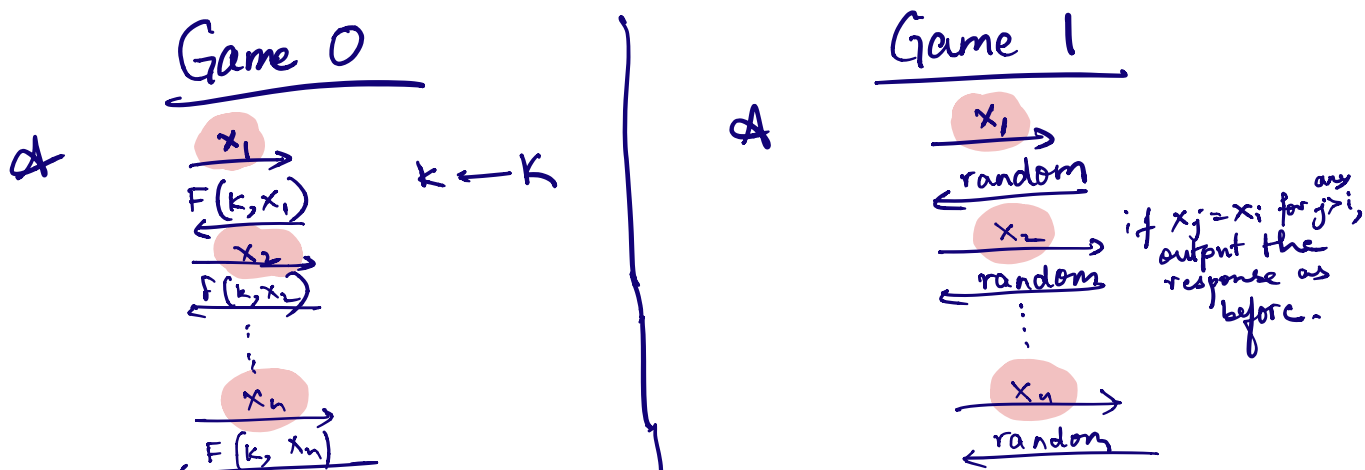
RANDOM : $1 \rightarrow 20,\ 2 \rightarrow 5, \ldots$

Defining Security for Block Ciphers

- A **pseudorandom function (PRF)** is a function from $\mathcal{K} \times \mathcal{X} \rightarrow \mathcal{C}$

  s.t. F (k, x) is efficiently computable for every k and m

  and is indistinguishable from a "random" function



$$\left| \Pr[\mathcal{A} = 1 \mid \text{Game } 0] - \Pr[\mathcal{A} = 1 \mid \text{Game } 1] \right| < \varepsilon$$

Question 1

Let $F: K \times X \rightarrow X$ be a secure PRP.

Is F a secure PRF?

1. Always
2. Never
3. Depends on F

Question 2

Let $F: K \times X \rightarrow \{0,1\}^{128}$ be a secure PRF.

Is the following ~~F~~ G a secure PRF?

$$G\cancel{F}(k, x) = \begin{cases} 1^{128} & \text{if } x=0 \\ F(k,x) & \text{otherwise} \end{cases}$$

1. Yes
2. ~~No~~
3. Depends on F

Game 0

$\mathcal{A}$ $\xrightarrow{x=0}$

$\xleftarrow{1^{128}}$

Game 1

$\xrightarrow{x=0}$

$\xleftarrow{\text{random}}$

if on input $x=0$, $\mathcal{A}$ see 1^{128}, then $\mathcal{A}$ will output 0, o/w output 1.

Question 3

Easy

Let $F: K \times X \to \{0,1\}^{128}$ be a secure PRF. ($X \in \{0,1\}^8$)

Can you build a PRG $G: K \to \{0,1\}^{4096}$ from F?

$$PRG(k) = \overbrace{F(k, 0^8) \,\|\, F(k, 0^7 1) \,\|\, F(k, 0^6 10) \ldots\ldots}^{2^5 \text{ terms}}$$

$$\frac{4096}{128} = \frac{2^{12}}{2^7} = 2^5.$$

Slightly Harder

Can you build a PRF from a PRG?

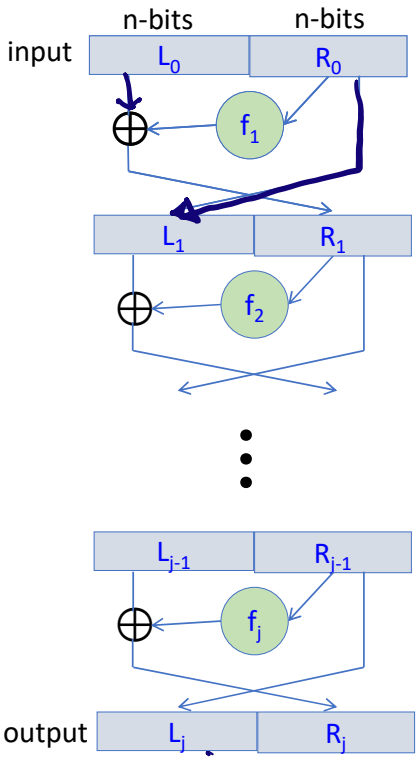
(Try on your own!)

DES

PRP : Pseudorandom permutation.

Data Encryption Standard

DES Core: Feistel Network



NOT necessarily invertible

Given functions $f_1, \ldots, f_j : \{0,1\}^n \longrightarrow \{0,1\}^n$

Build an invertible function $F : \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$

$x = (L_0, R_0)$

$L_1 = R_0$

$R_1 = L_0 \oplus f_1(R_0)$

$L_2 = R_1$

$R_2 = L_1 \oplus f_2(R_1)$ .....

L_1, R_1

$R_0 = L_1$

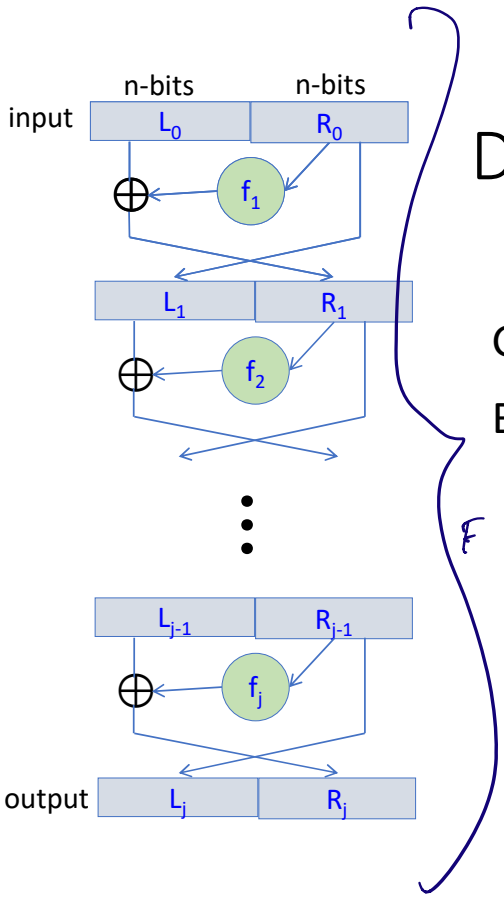
$L_0 = R_1 \oplus f_1(R_0)$

$= R_1 \oplus f_1(L_1)$

$E(k_1, (D(k_2, E(k_3, m))))$

Lucifer (precursor)

DES Core: Feistel Network



Given functions $f_1, \ldots, f_j : \{0,1\}^n \longrightarrow \{0,1\}^n$

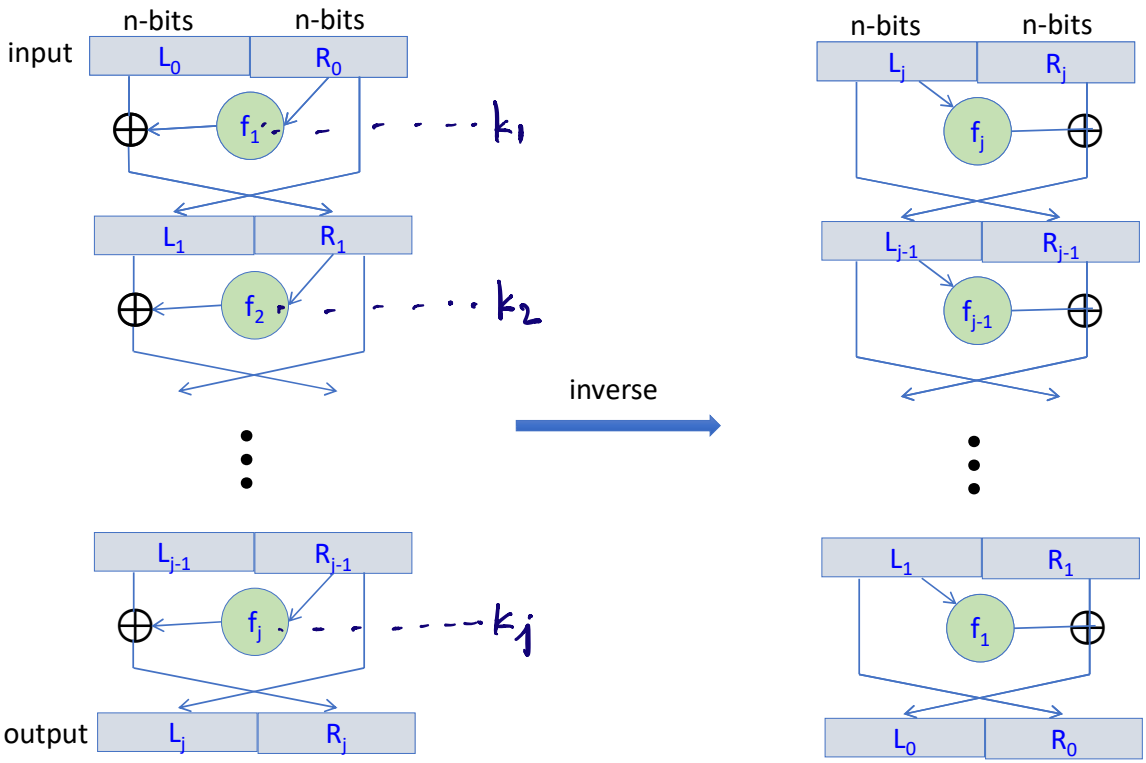
Build an invertible function $F : \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$

$(L_0, R_0) \qquad (L_j, R_j)$

$R_{j-1} = L_j$

$L_{j-1} = R_j \oplus f_{j-1}(R_{j-1})$

$= R_j \oplus f_{j-1}(L_j)$


n-bits
n-bits
input
L_0
R_0
f_1
k_1
L_1
R_1
f_2
k_2
L_{j-1}
R_{j-1}
f_j
k_j
output
L_j
R_j
inverse
n-bits
n-bits
L_j
R_j
f_j
L_{j-1}
R_{j-1}
f_{j-1}
L_1
R_1
f_1
L_0
R_0

PRFs => PRPs

Luby-Rackoff '85:

If f: $K \times \{0,1\}^n \longrightarrow \{0,1\}^n$ is a secure PRF

then 3-round Feistel F: $K^3 \times \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$ is a secure PRP

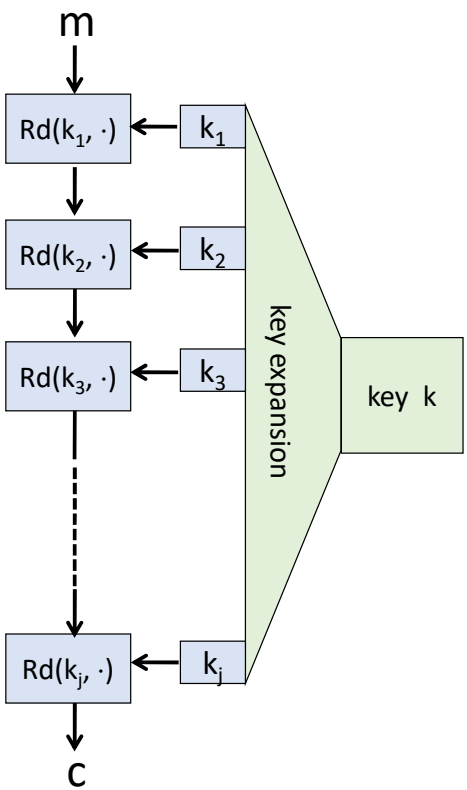
To prove this, show that :

① Feistel results in an invertible function

② If each f_i $\left[\begin{array}{l} f_1 = PRF(k_1, \cdot) \\ f_2 = PRF(k_2, \cdot) \\ f_3 = PRF(k_3, \cdot) \end{array}\right]$ is a secure PRF,

i.e.

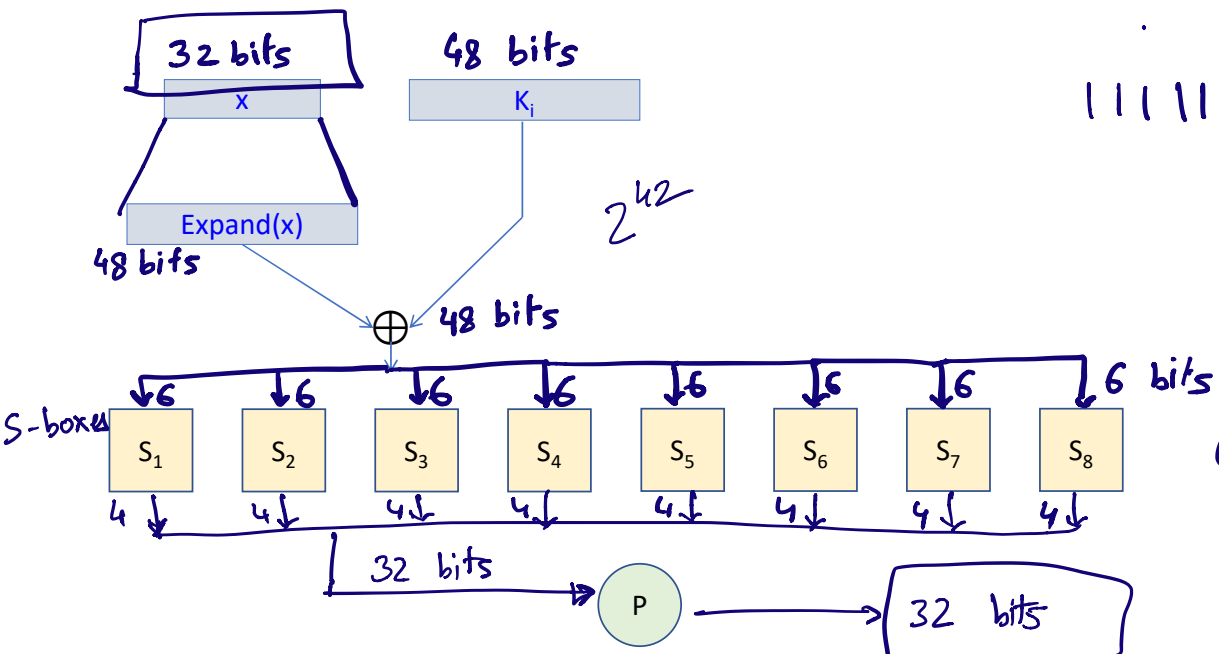
then so is the feistel.



DES

DES: n = 64 bits, k = 56 bits, j = 16, each k_j = 48 bits

Each function $f_i = F(k_i, x)$ is:



S_1

000000 → 0110

000001 → 1010

⋮ → 1010

111111 → 0101

confusion network

Requirements on S-boxes

- Implemented as lookup tables
- Must **not be** linear functions
- Note that all other functions are linear. If S-boxes are linear, it means that DES encryption can be represented as a matrix

$$\mathrm{DES}(k,m) = \overset{832}{64\,\boxed{B}} \cdot \begin{bmatrix} m \\ k_1 \\ k_2 \\ \vdots \\ k_{16} \end{bmatrix} = \boxed{c} \pmod 2$$

$$\overset{\mathrm{DES}(k,m_1)}{B\begin{bmatrix} m_1 \\ k \end{bmatrix}} \oplus \overset{\mathrm{DES}(k,m_2)}{B\begin{bmatrix} m_2 \\ k \end{bmatrix}} \oplus \overset{\mathrm{DES}(k,m_3)}{B\begin{bmatrix} m_3 \\ k \end{bmatrix}} = B\begin{bmatrix} m_1 \oplus m_2 \oplus m_3 \\ k \oplus k \oplus k \end{bmatrix}$$

$$B\begin{bmatrix} m_1 \oplus m_2 \\ 0 \end{bmatrix}$$

$k \oplus k = 0$

This is BAD!

Summary

- Block ciphers formalized as pseudorandom permutations (PRPs)

  special case of PRF

- Feistel Network, DES

- Next time, we will look at AES