Lecture 2 – Block ciphers, PRFs/PRPs, DES, AES

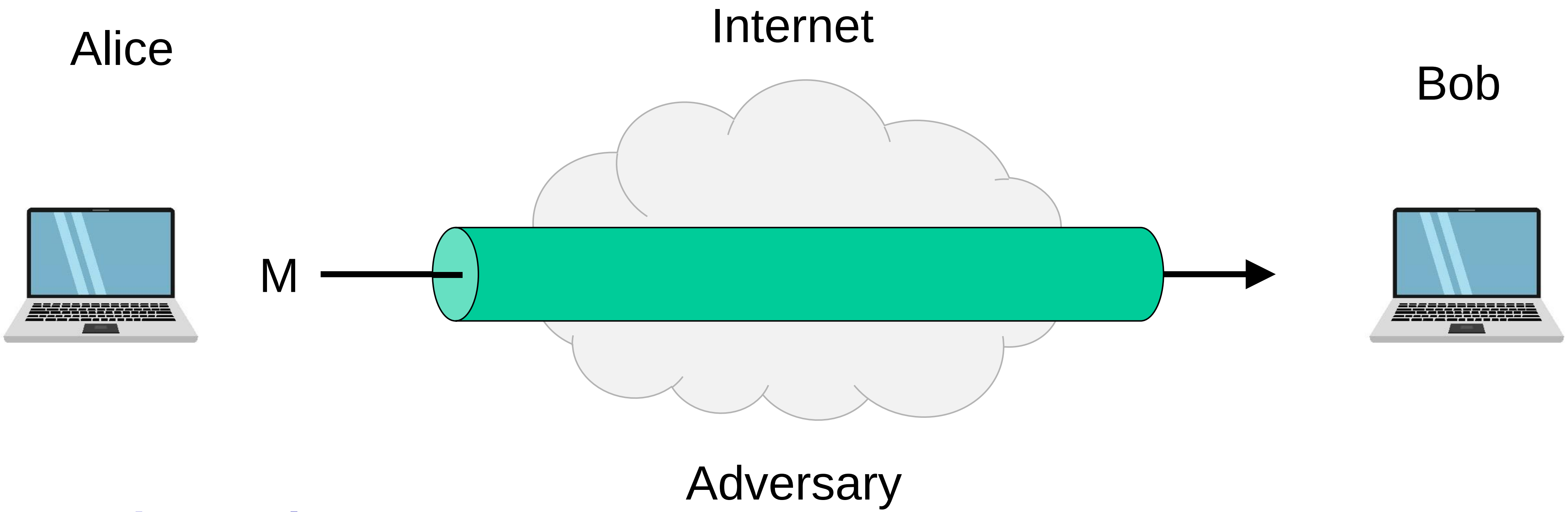
TEK4500

01.09.2020

Håkon Jacobsen

hakon.jacobsen@its.uio.no

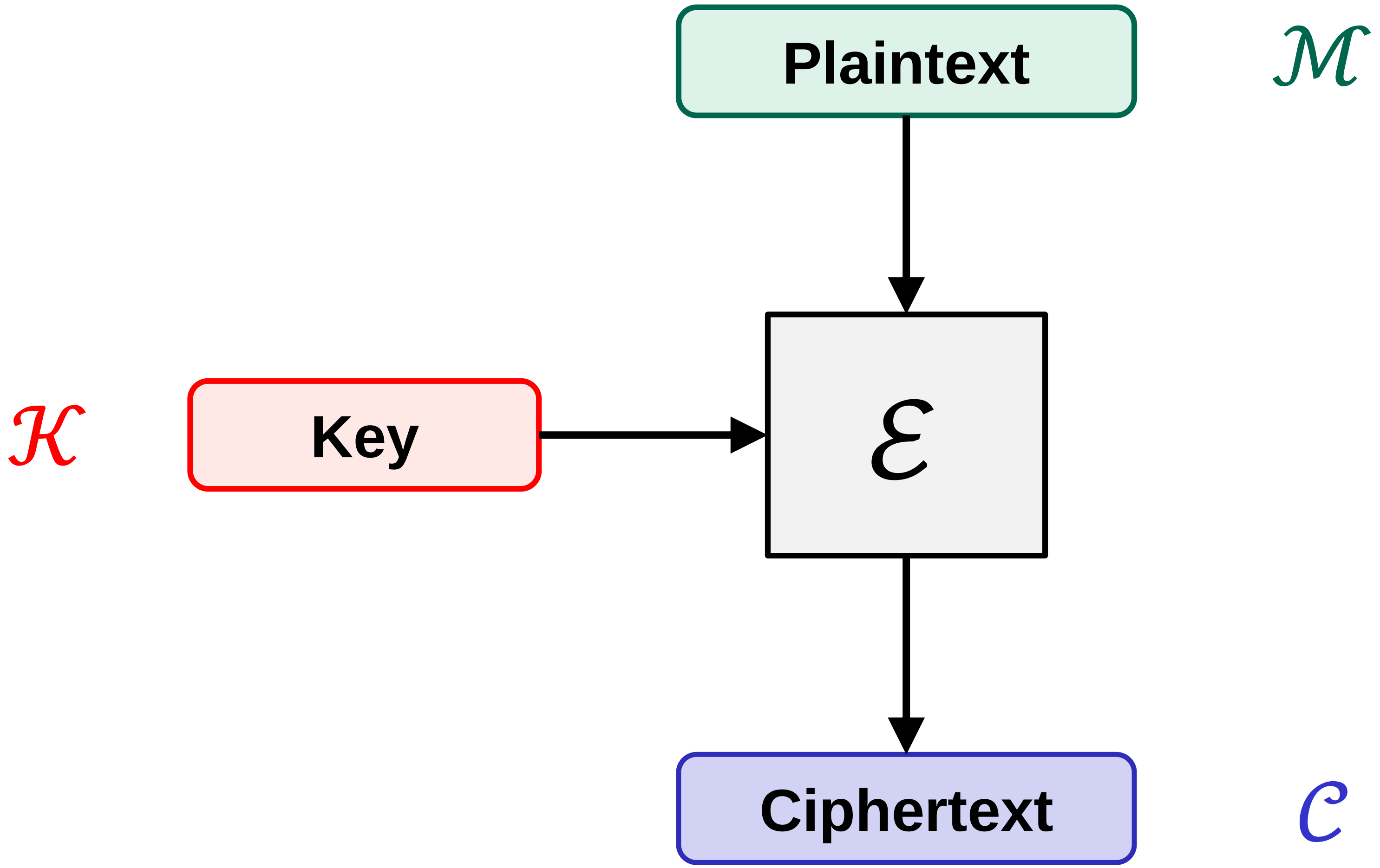
Ideal solution: secure channels



Security goals:

- **Data privacy:** adversary should not be able to read message M ✓
- **Data integrity:** adversary should not be able to modify message M ✓
- **Data authenticity:** message M really originated from Alice ✓

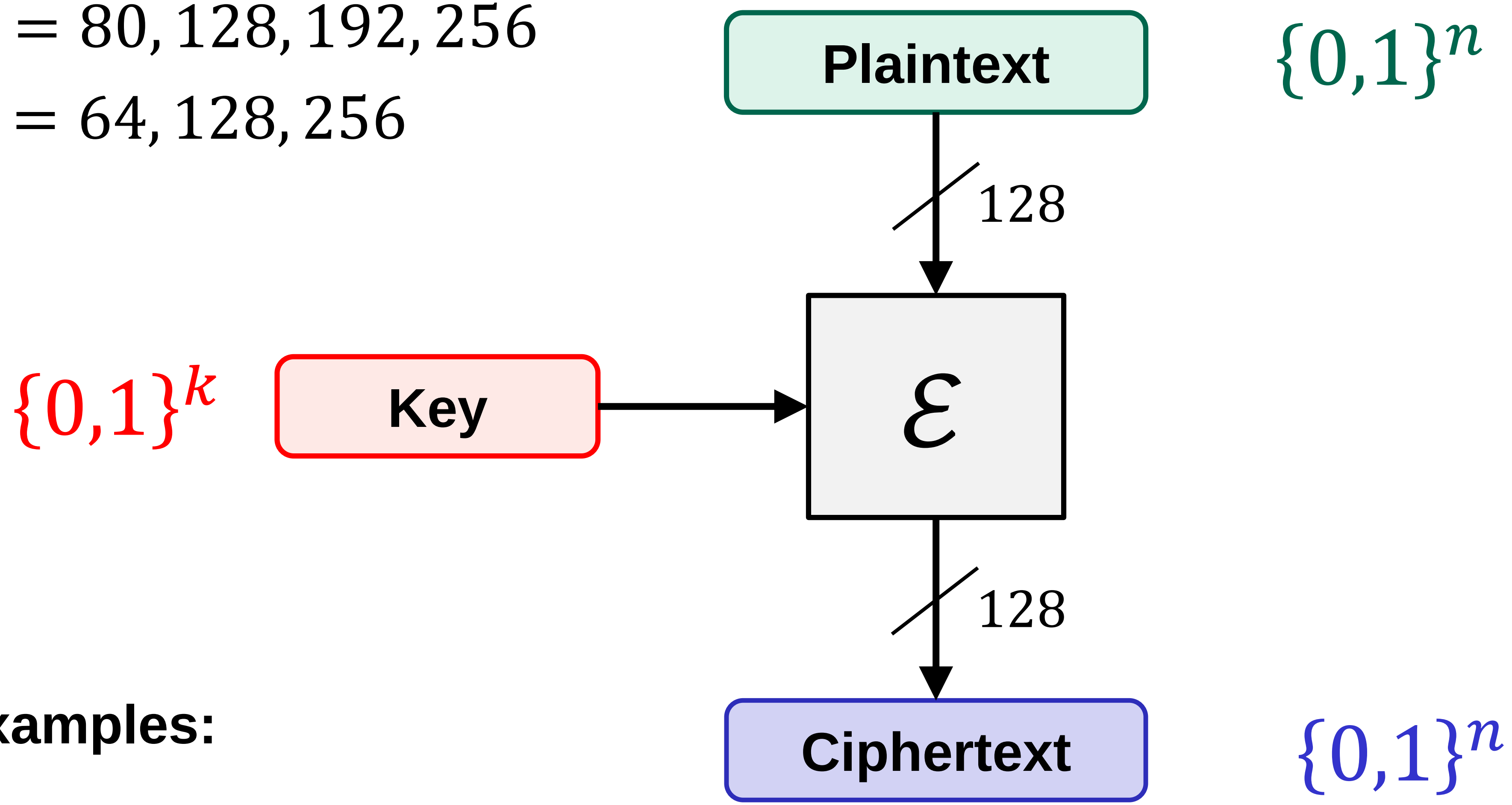
Encryption schemes



Block ciphers

$k = 80, 128, 192, 256$

$n = 64, 128, 256$

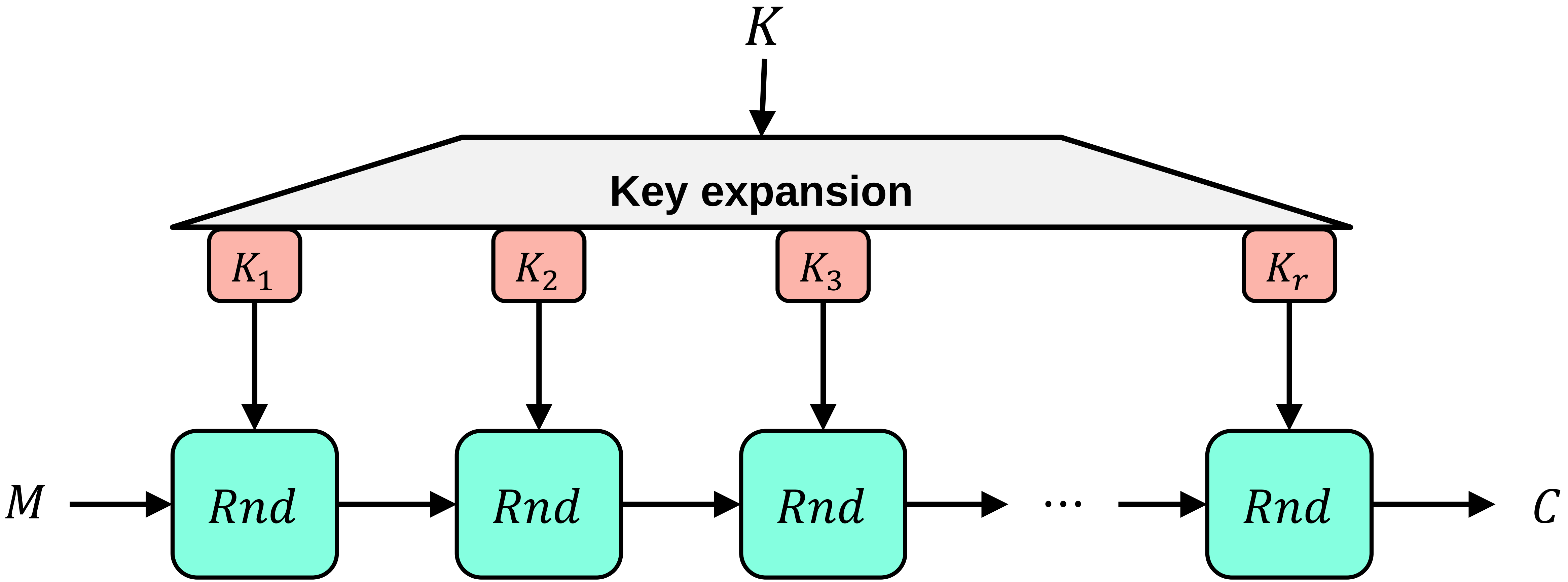


Examples:

DES: $k = 56, \; n = 64$

AES: $k = 128, 192, 256, \; n = 128$

Block ciphers



$Rnd(K_i, M)$ is called a **round function**

DES: $r = 16$

AES-128/192/256: $r = 10/12/14$

Block cipher applications (1)

- Encryption of messages of 128 bits (block length)

K

$Y_1, Y_2, \ldots, Y_t$

K

$Y_i \leftarrow E_K(M_i)$

$M_i \leftarrow E_K^{-1}(Y_i)$

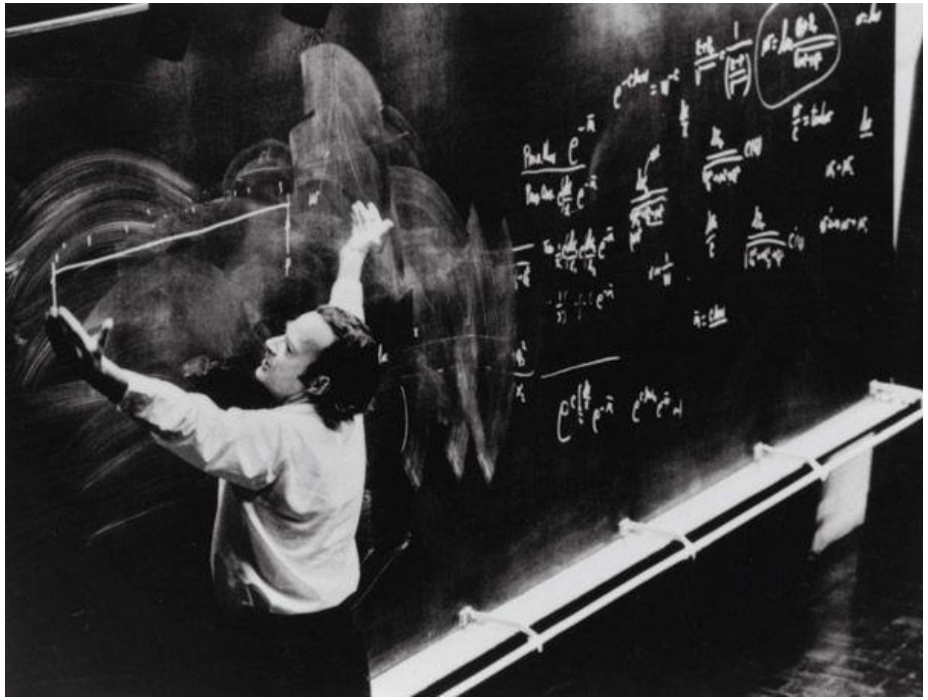
- **However:** we usually want to encrypt messages of *arbitrary* length!
  - Splitting the message into multiple 128 bit blocks (like above) is **not secure!**
  - Need to use them in a proper **mode-of-operation** (covered later in the course)

- Correct viewpoint: block ciphers are **not** encryption schemes!
  - Block ciphers are **primitives** used to construct other things

Block cipher applications (2)

- The “work horse” of crypto

- Can be used to build:
  - Encryption of arbitrary length messages (including stream ciphers)
  - Message authentication codes
  - Authenticated encryption
  - Hash functions
  - (Cryptographically secure) pseudorandom generators
  - Key derivation functions

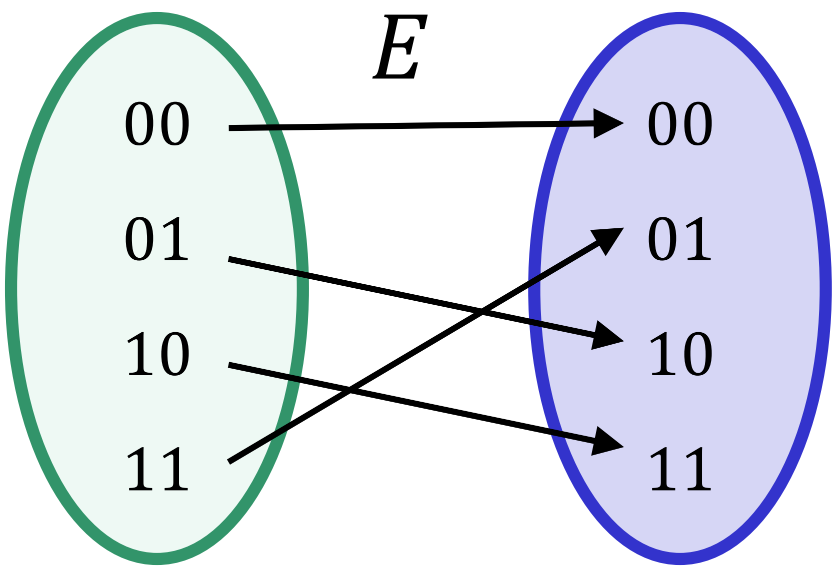


DEFINING BLOCK CIPHERS

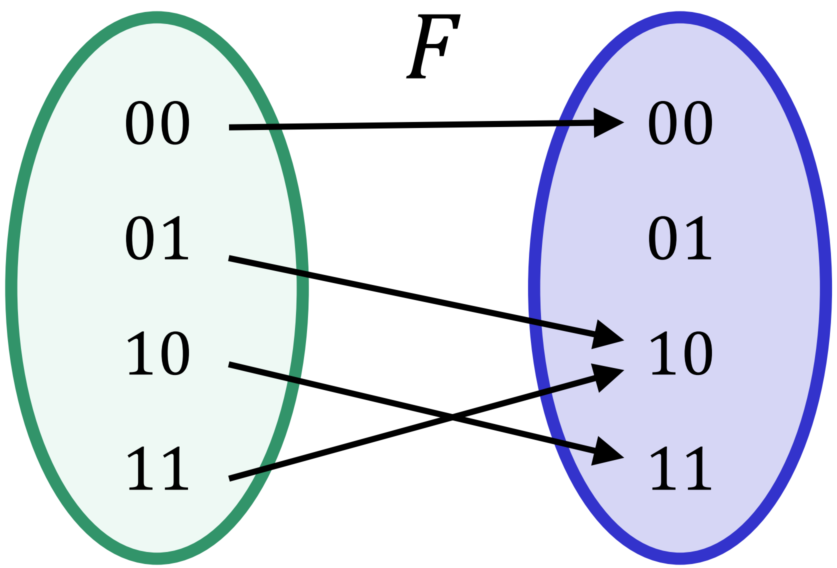
Permutations

Definition: A function $E : \{0,1\}^n \rightarrow \{0,1\}^n$ is a **permutation** if there exists an inverse function E^{-1} such that

$$E^{-1}\big(E(X)\big) = X$$



Permutation

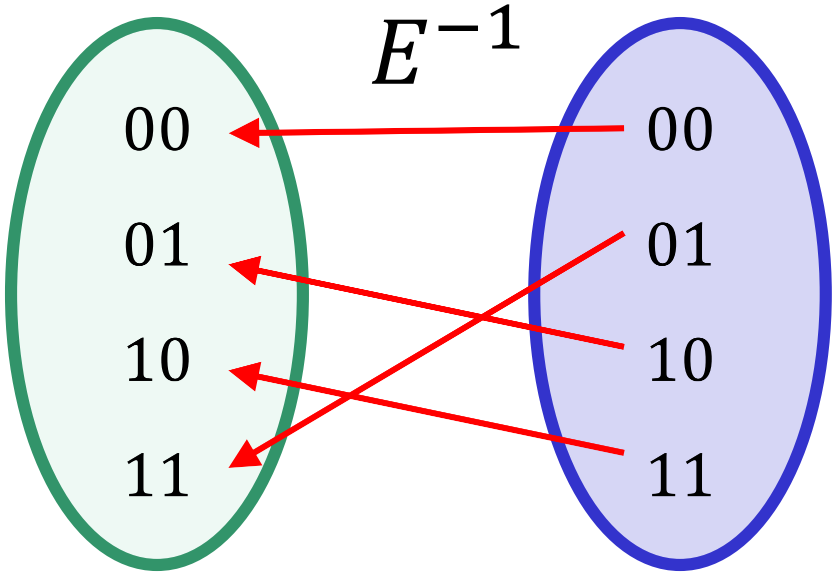


Not a permutation

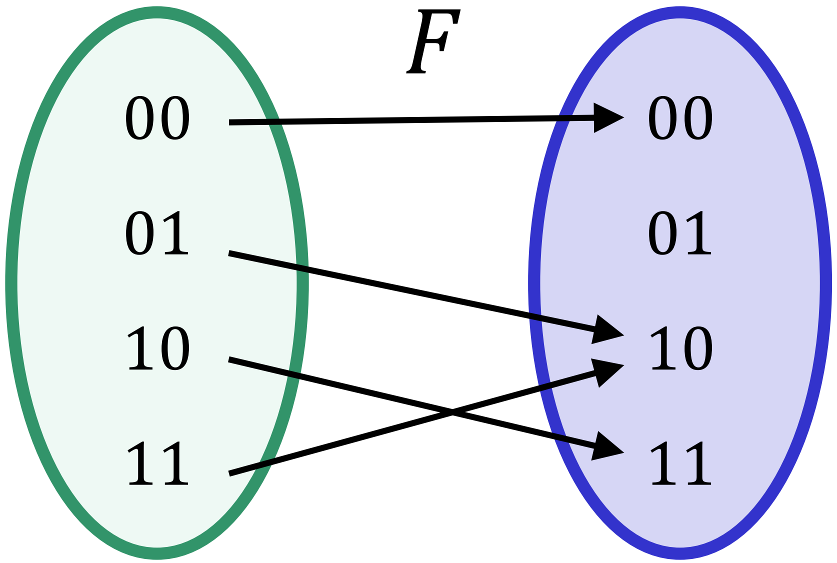
Permutations

Definition: A function $E : \{0,1\}^n \to \{0,1\}^n$ is a **permutation** if there exists an inverse function E^{-1} such that

$$E^{-1}\big(E(X)\big) = X$$



Permutation



Not a permutation

Block cipher security

- Which security properties should a block cipher satisfy?
  - I.e., what should the **security definition** of a block cipher look like?

- Some suggestions:

  - **P1:** Should be hard to obtain K from $Y \leftarrow E_K(X)$ for secret K
  - **P2:** Should be hard to obtain K from $Y_1, Y_2, \ldots$ where $Y_i \leftarrow E_K(X_i)$
  - **P3:** Should be hard to obtain X from $Y \leftarrow E_K(X)$
  - **P4:** Should be hard to obtain *any* X_i from $Y_1, Y_2, \ldots$ where $Y_i \leftarrow E_K(X_i)$
  - **P5:** Should be hard to learn *any* bit of X from $Y \leftarrow E_K(X)$
  - **P6:** Should be hard to detect *repetitions* among $X_1, X_2, \ldots$ from $Y_1, Y_2, \ldots$
  - **P7:** …

Block cipher security

- Which security properties should a block cipher satisfy?
  - I.e., what should the **security definition** of a block cipher look like?

- Some suggestions:

  - **P1:** Should be hard to obtain K from $Y \leftarrow E_K(X)$ for secret K
  - **P2:** Should be hard to obtain K from $Y_1, Y_2, \dots$ where $Y_i \leftarrow E_K(X_i)$
  - **P3:** Should be hard to obtain X from $Y \leftarrow E_K(X)$
  - **P4:** Should be hard to obtain *any* X_i from $Y_1, Y_2, \dots$ where $Y_i \leftarrow E_K(X_i)$
  - **P5:** Should be hard to learn *any* bit of X from $Y \leftarrow E_K(X)$
  - ~~**P6:** Should be hard to detect *repetitions* among $X_1, X_2, \dots$ from $Y_1, Y_2, \dots$~~
  - **P7:** …

Not good enough! (P1–P5)

Impossible! (P6)

Pseudorandom functions (PRFs) and permutations (PRP)

Definition: A **pseudorandom function (PRF)** is a function

$$F : \{0,1\}^k \times \{0,1\}^{in} \rightarrow \{0,1\}^{out}$$

- k, in, out are called the **key-length**, i

- Think of a PRF as a *family* of functio
  - For each $K \in \{0,1\}^k$ we get a function $F_K : \{0,1\}^{in}$ … defined by $F_K(X) = F(K, X)$

PRP = block cipher!

also: all PRPs are PRFs
(but not the other way around)

Definition: A **pseudorandom permutation (PRP)** is a function

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

such that the function $E_K : \{0,1\}^n \rightarrow \{0,1\}^n$ defined by $E_K(X) = E(K, X)$ is a *permutation* for all $K \in \{0,1\}^k$

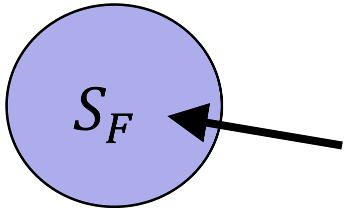
Secure PRFs

- Let $F : \{0,1\}^k \times \{0,1\}^{in} \rightarrow \{0,1\}^{out}$

  $\mathrm{Func}[in, out]$: the set of *all* functions from $\{0,1\}^{in}$ to $\{0,1\}^{out}$

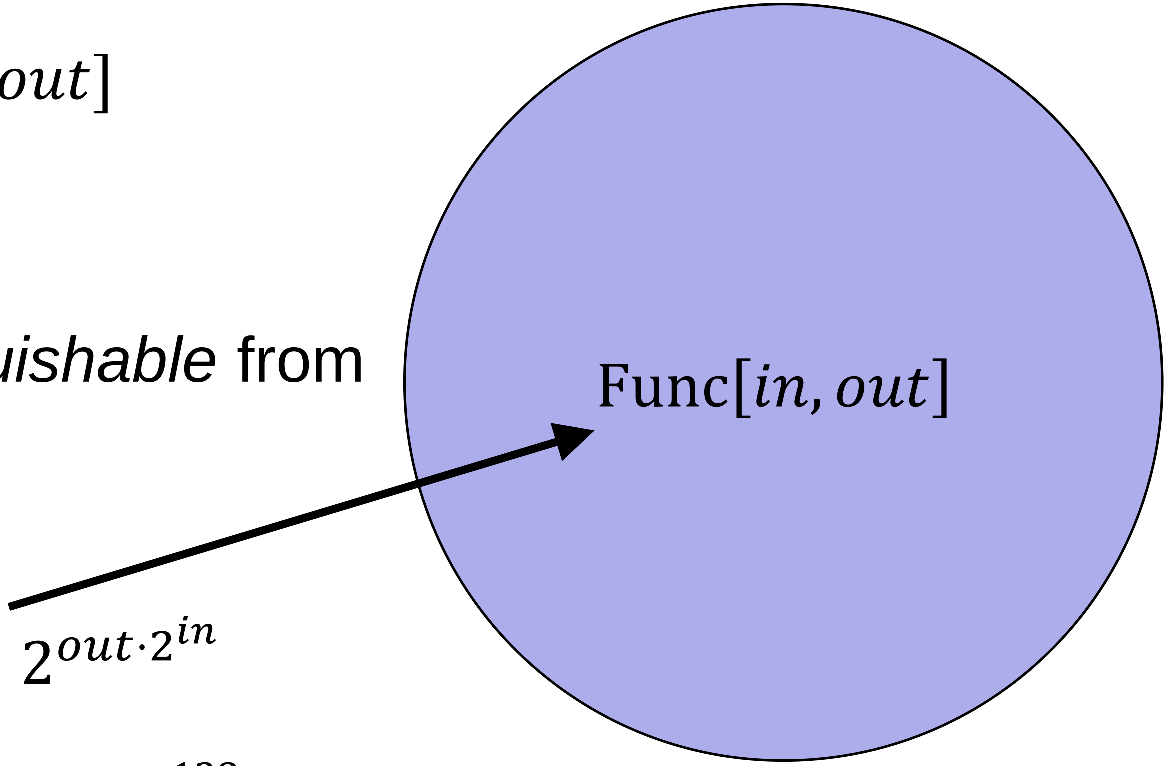
  $S_F = \{\ F_K \mid K \in \{0,1\}^k\} \subseteq \mathrm{Func}[in, out]$

- Intuition: F is **secure** if
  a random function in S_F is *indistinguishable* from
  a random function in $\mathrm{Func}[in, out]$



size 2^k

AES-128: 2^{128}



size $2^{out \cdot 2^{in}}$

$(2^{128})^{2^{128}}$

Random functions

- Let $\tilde{F} \in \mathrm{Func}[in, out]$

X	$\tilde{F}(X)$
$000 \ldots 000$	$101 \ldots 111$
$000 \ldots 001$	$001 \ldots 001$
$000 \ldots 010$	$111 \ldots 100$
$000 \ldots 011$	$101 \ldots 000$
$\vdots$	$\vdots$
$111 \ldots 111$	$100 \ldots 010$

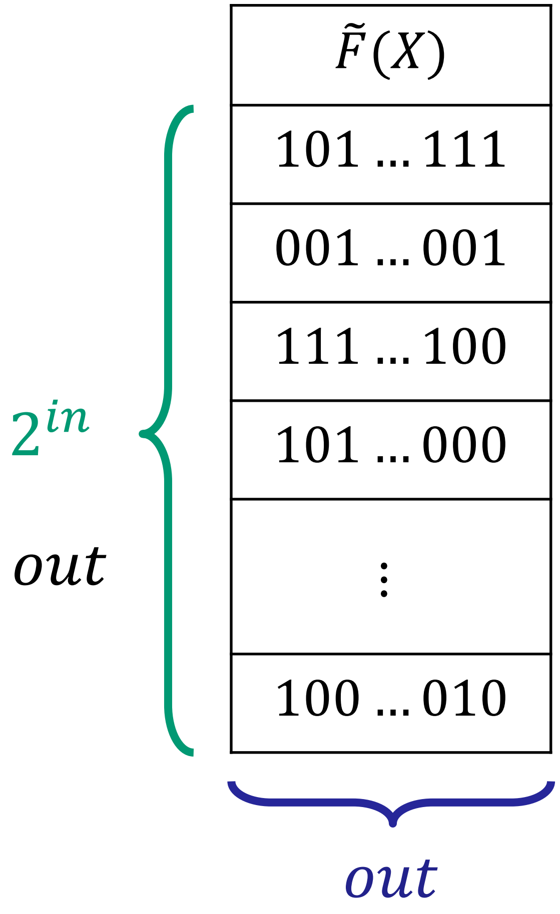
2^{in}

out

Random functions

- Let $\tilde{F} \in \mathrm{Func}[in, out]$
- Bits needed to specify *one* function $\tilde{F}$: $2^{in} \cdot out$
- Each bit string of length $2^{in} \cdot out$ specifies a unique function $\Longrightarrow$

$$|\mathrm{Func}[in, out]| = \text{the number of bitstrings of length } 2^{in} \cdot out$$
$$= 2^{(2^{in} \cdot out)}$$



Random functions – alternate view

$\widetilde{F}$
$T \leftarrow [\,]$
$\text{LookUp}(X \in \{0,1\}^{in})$
----
if $T[X]$ undefined **then**:
$\quad T[X] \xleftarrow{\$} \{0,1\}^{out}$
return $T[X]$

PRF security

$\approx$

$\boldsymbol{F}$
$K \xleftarrow{\$} \{0,1\}^{k}$
$\text{LookUp}(X \in \{0,1\}^{in})$
----
return $F(K, X)$

PRF security definition

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$ $A =$ 

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$b = 0$ $b = 1$

$\widetilde{F}$

$T \leftarrow [\,]$

$\mathrm{LookUp}(X \in \{0,1\}^{in})$

---

if $T[X]$ undefined **then**:

$T[X] \xleftarrow{\$} \{0,1\}^{out}$

return $T[X]$

X_1, X_2, $\vdots$ (↑) Y_1, Y_2, $\vdots$ (↓)

F

$K \xleftarrow{\$} \{0,1\}^k$

$\mathrm{LookUp}(X \in \{0,1\}^{in})$

---

return $F(K, X)$

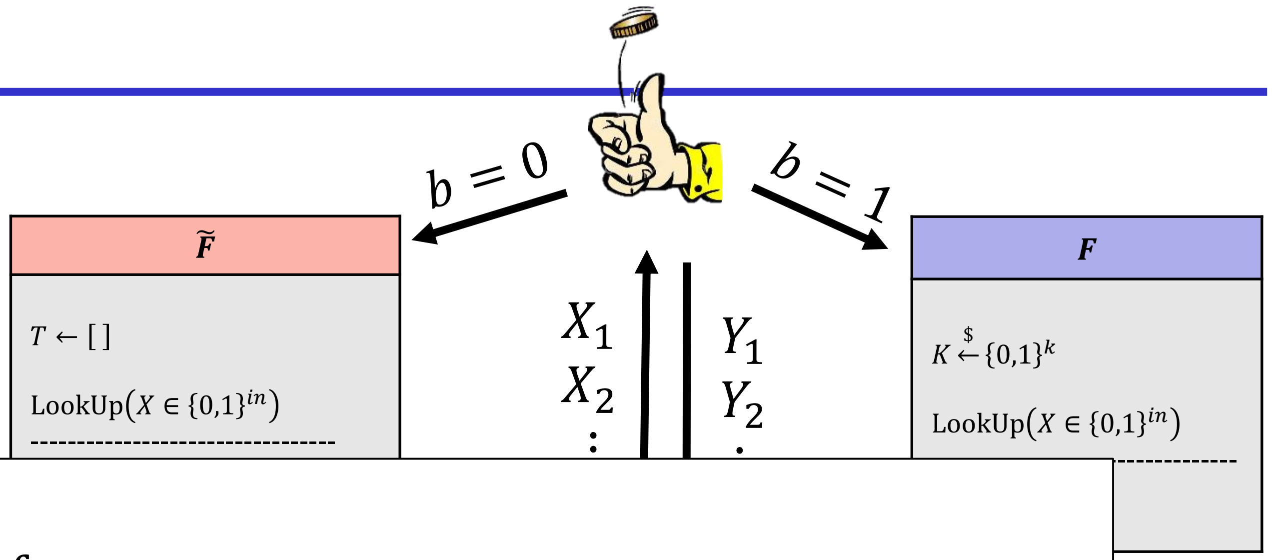
Definition: The **PRF-advantage** of an adversary A is

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) = \left|\Pr\left[A \text{ wins in PRF experiment}\right] - 1/2\right|$$

PRF security definition

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$ $A =$ 

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$



$\mathbf{Adv}_F^{\mathrm{prf}}(A) \approx 0.5$ = adversary is doing well

$\mathbf{Adv}_F^{\mathrm{prf}}(A) \approx 0$ = adversary is doing poorly

Definition: The **PRF-advantage** of an adversary A is

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) = \left|\Pr\left[A \text{ wins in PRF experiment}\right] - 1/2\right|$$

PRF security definition

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$
1. $b \overset{\$}{\leftarrow} \{0,1\}$
2. $F_0 \overset{\$}{\leftarrow} \mathrm{Func}[in, out]$
3. $K \overset{\$}{\leftarrow} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \overset{?}{=} b$

Intuitive idea: F is a **secure** PRF if $\mathrm{Adv}_F^{\mathrm{prf}}(A)$ is "*small*" for all "*practical*" A

Definition: The **PRF-advantage** of an adversary A is

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) = \left|\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] - 1/2\right|$$

Understanding "advantage"

- F is a **secure** PRF if $\mathbf{Adv}_F^{\mathrm{prf}}(A)$ is "*small*" for *all* adversaries A that use a "*practical*" amount of resources

- Advantage depends on the adversary's:
  - strategy
  - available resources: running time, number of oracle calls (calls to $\tilde{F}$ / F), memory…

- What does *small* and *practical* mean?
  - **Example: 80-bit** security:

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) \leq \frac{q}{2^{80}}$$

    for all A that makes at most q oracle calls

  - **Example:** a PRF is insecure if we can come up with an adversary having good advantage and not using too many resources

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$$
\begin{aligned}
\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] &= \Pr[\,b' = 1 \mid b = 1\,] \cdot \Pr[\,b = 1] + \Pr[\,b' = 0 \mid b = 0\,] \cdot \Pr[\,b = 0] \\
&= \Pr[\,b' = 1 \mid b = 1\,] \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2 \\
&= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2 \\
&= 1 \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2
\end{aligned}
$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[\,b' = 1 \mid b = 1\,] \cdot \Pr[\,b = 1] + \Pr[\,b' = 0 \mid b = 0\,] \cdot \Pr[\,b = 0]$$

$$= \Pr[\,b' = 1 \mid b = 1\,] \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2$$

$$= 1 \cdot 1/2 + (1 - \Pr[\,b' = 1 \mid b = 0\,]) \cdot 1/2$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[b' = 1 \mid b = 1] \cdot \Pr[b = 1] + \Pr[b' = 0 \mid b = 0] \cdot \Pr[b = 0]$$

$$= \Pr[b' = 1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= 1 \cdot 1/2 + (1 - \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 0]) \cdot 1/2$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \stackrel{\$}{\leftarrow} \{0,1\}$
2. $F_0 \stackrel{\$}{\leftarrow} \mathrm{Func}[in, out]$
3. $K \stackrel{\$}{\leftarrow} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[b' = 1 \mid b = 1] \cdot \Pr[b = 1] + \Pr[b' = 0 \mid b = 0] \cdot \Pr[b = 0]$$

$$= \Pr[b' = 1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= 1 \cdot 1/2 + (1 - \Pr[Y_0 = X_0 \oplus X_1 \oplus Y_1 \mid b = 0]) \cdot 1/2$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[b' = 1 \mid b = 1] \cdot \Pr[b = 1] + \Pr[b' = 0 \mid b = 0] \cdot \Pr[b = 0]$$

$$= \Pr[b' = 1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= 1 \cdot 1/2 + (1 - \Pr[Y_0 = Z \mid b = 0]) \cdot 1/2$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \overset{\$}{\leftarrow} \{0,1\}$
2. $F_0 \overset{\$}{\leftarrow} \mathrm{Func}[in, out]$
3. $K \overset{\$}{\leftarrow} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \overset{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[\,b' = 1 \mid b = 1\,] \cdot \Pr[\,b = 1] + \Pr[\,b' = 0 \mid b = 0\,] \cdot \Pr[\,b = 0]$$

$$= \Pr[\,b' = 1 \mid b = 1\,] \cdot 1/2 \quad + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[\,b' = 0 \mid b = 0\,] \cdot 1/2$$

$$= 1 \cdot 1/2 \quad + (1 - 2^{-n}) \cdot 1/2$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \overset{\$}{\leftarrow} \{0,1\}$
2. $F_0 \overset{\$}{\leftarrow} \mathrm{Func}[in, out]$
3. $K \overset{\$}{\leftarrow} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \overset{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = \Pr[b' = 1 \mid b = 1] \cdot \Pr[b = 1] + \Pr[b' = 0 \mid b = 0] \cdot \Pr[b = 0]$$

$$= \Pr[b' = 1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= \Pr[Y_0 \oplus Y_1 = X_0 \oplus X_1 \mid b = 1] \cdot 1/2 + \Pr[b' = 0 \mid b = 0] \cdot 1/2$$

$$= 1 - 2^{-n} \cdot 1/2 = 1 - 2^{-n+1}$$

Example

- Let $F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ be defined by $F(K, X) = K \oplus X$

- **Claim:** F is not a secure PRF

A

1. Choose $X_0 \neq X_1 \in \{0,1\}^n$ arbitraritly
2. Query X_0 and X_1 to challenger
3. Receive back $Y_0 = \tilde{F}(X_0)$ and $Y_1 = \tilde{F}(X_1)$
4. Output $b' = 1$ if $Y_0 \oplus Y_1 = X_0 \oplus X_1$, else, output $b' = 0$

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$

1. $b \overset{\$}{\leftarrow} \{0,1\}$
2. $F_0 \overset{\$}{\leftarrow} \mathrm{Func}[in, out]$
3. $K \overset{\$}{\leftarrow} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \overset{?}{=} b$

$$\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] = \Pr[b' = b] = 1 - 2^{-n+1}$$

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) = \left|\Pr\left[\mathbf{Exp}_F^{\mathrm{prf}}(A) \Rightarrow \mathrm{true}\right] - 1/2\right| = |1 - 2^{-n+1} - 1/\ 2| = 1\backslash 2\ - 2^{-n+1} \approx 1/2$$

PRP security definition

$\mathbf{Exp}_F^{\mathrm{prf}}(A)$
1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$} \mathrm{Func}[in, out]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. $\mathbf{return}\ b' \stackrel{?}{=} b$

PRP security definition

$\mathbf{Exp}_F^{\sout{\text{prf}}\ \text{prp}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $F_0 \xleftarrow{\$}$ ~~$\mathrm{Func}[in, out]$~~ $\mathrm{Perm}[n]$
3. $K \xleftarrow{\$} \{0,1\}^k$
4. $F_1 \leftarrow F_K$
5. $b' \leftarrow A^{F_b(\cdot)}$
6. **return** $b' \stackrel{?}{=} b$

$\widetilde{F}$

$T \leftarrow [\,]$

$\mathrm{LookUp}\left(X \in \{0,1\}^{in}\right)$

----

if $T[X]$ undefined **then**:

$T[X] \xleftarrow{\$} \{0,1\}^{out} \setminus T.\mathrm{values}$

return $T[X]$

Definition: The **PRP-advantage** of an adversary A is

$$\mathbf{Adv}_F^{\mathrm{prp}}(A) = \left|\Pr\left[\mathbf{Exp}_F^{\mathrm{prp}}(A) \Rightarrow \mathrm{true}\right] - 1/2\ \right|$$

Block cipher security

- Which security properties should a block cipher satisfy?
  - I.e., what should the **security definition** of a block cipher look like?

E is PRF/PRP secure $\Longrightarrow$ E has properties P1 – P5

- **P1:** Should be hard to obtain K from $Y \leftarrow E_K(X)$ for secret K
- **P2:** Should be hard to obtain K from $Y_1, Y_2, \ldots$ where $Y_i \leftarrow E_K(X_i)$
- **P3:** Should be hard to obtain X from $Y \leftarrow E_K(X)$
- **P4:** Should be hard to obtain *any* X_i from $Y_1, Y_2, \ldots$ where $Y_i \leftarrow E_K(X_i)$
- **P5:** Should be hard to learn *any* bit of X from $Y \leftarrow E_K(X)$
- ~~**P6:** Should be hard to detect *repetitions* among $X_1, X_2, \ldots$ from $Y_1, Y_2, \ldots$~~
- **P7:** …

Not good enough!

Impossible!

E is ***not*** PRF/PRP secure $\Longleftarrow$ E has does ***not*** have properties P1 – P5

Logic 101

$A \Rightarrow B$

is *equivalent to:*

$\bar{A} \Leftarrow \bar{B}$

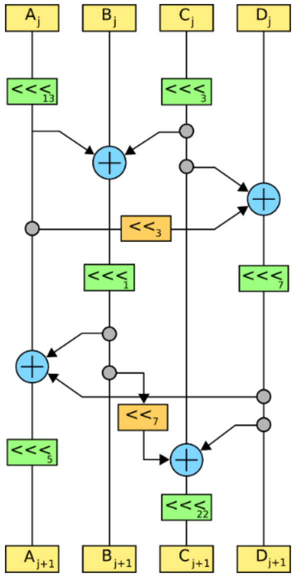
PRP security $\Longrightarrow$ PRF security

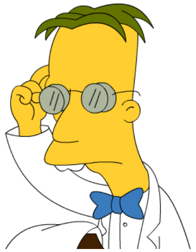
Theorem: (PRP/PRF Switching Lemma)

A secure PRP $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is also a secure PRF.

In particular, for all A making at most q oracle queries:

$$\mathbf{Adv}_E^{\mathrm{prf}}(A) \leq \mathbf{Adv}_E^{\mathrm{prp}}(A) + \frac{2q^2}{2^n}$$





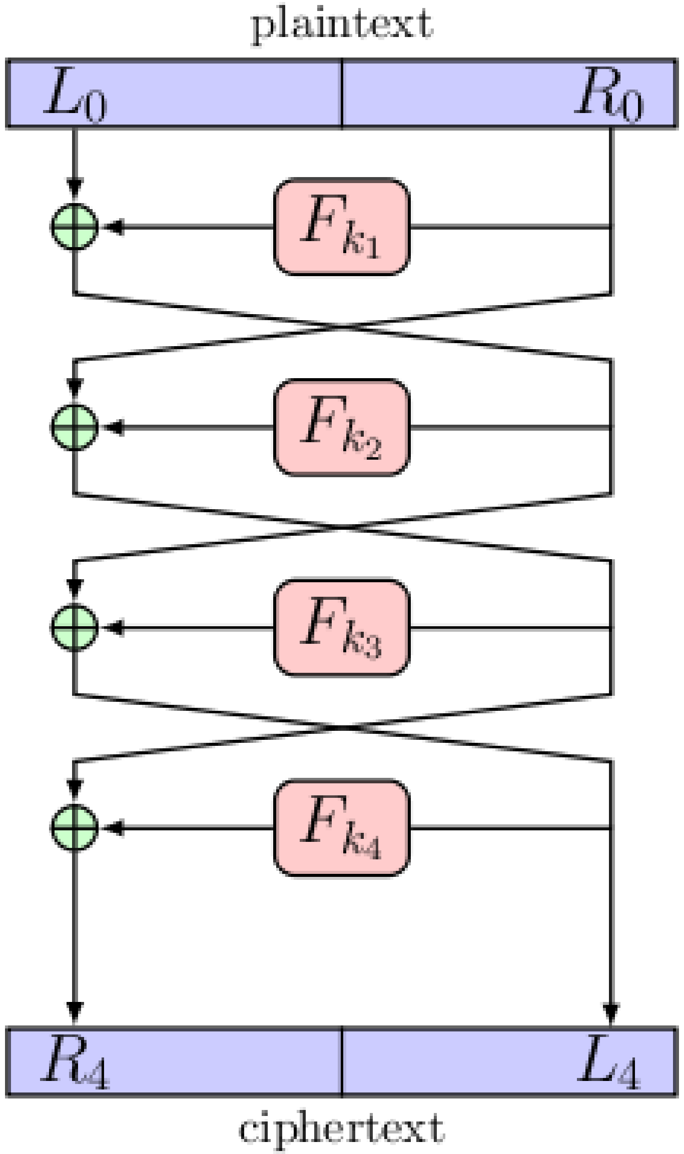
CONSTRUCTING BLOCK CIPHERS

PRPs from PRFs – the Feistel construction

- Let $F : \{0,1\}^k \times \{0,1\}^{n/2} \to \{0,1\}^{n/2}$ be a **PRF**
  - not a *permutation!*
- Function $E(K,X) = \mathrm{Feistel}_F^{(4)}(K,X)$ *is* a **PRP**
  - Called a **Feistel network/construction**
  - $E : \{0,1\}^{4k} \times \{0,1\}^n \to \{0,1\}^n$
- More or less DES:

$$\mathrm{DES} \approx \mathrm{Feistel}_F^{(16)} : \{0,1\}^{56} \times \{0,1\}^{64} \to \{0,1\}^{64}$$

(56-bit key is expanded to 16 48-bit roundkeys)

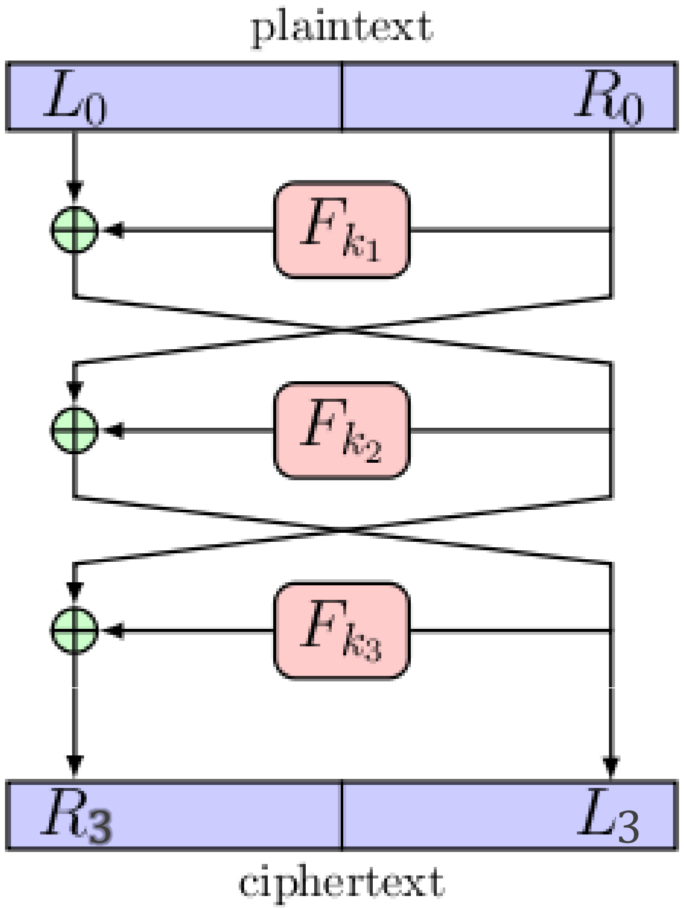


Feistel network security – theory

Theorem: (Luby & Rackoff '86)

If $F : \{0,1\}^k \times \{0,1\}^{n/2} \rightarrow \{0,1\}^{n/2}$ is a **secure** PRF

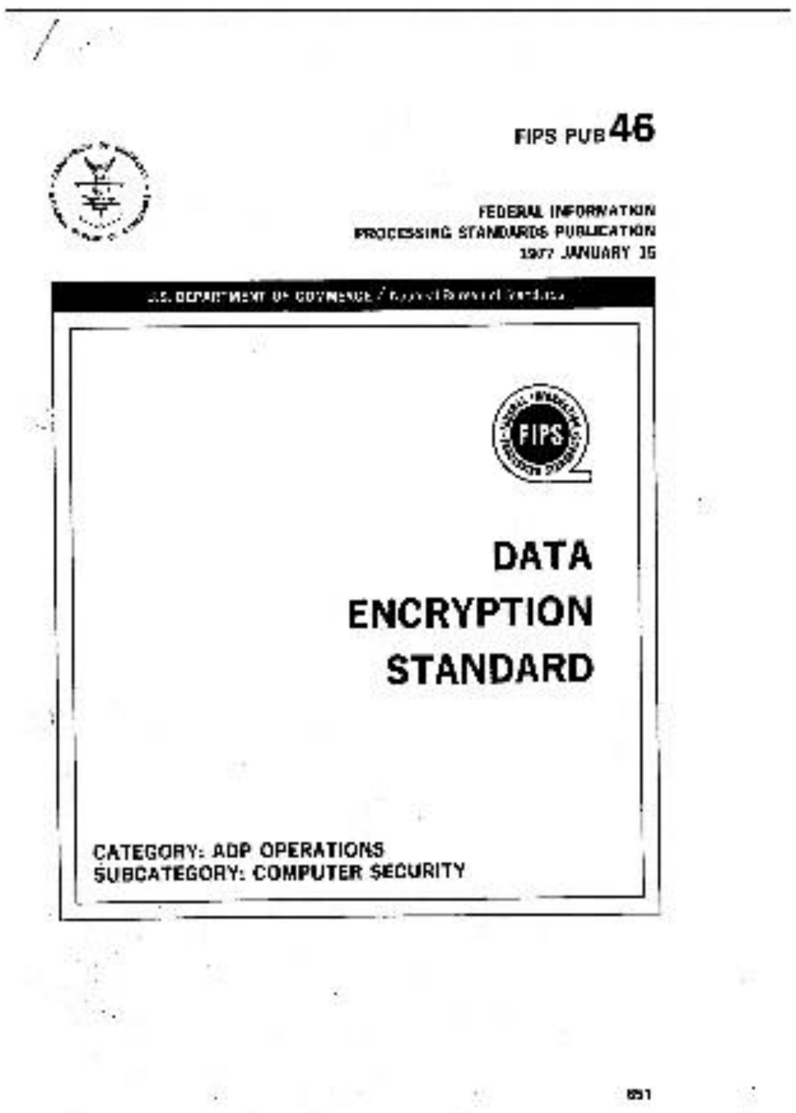
$\Longrightarrow$ 3-round Feistel $E : \{0,1\}^{3k} \times \{0,1\}^n \rightarrow \{0,1\}^n$ is a **secure** PRP



Data Encryption Standard (DES)

- 1972 – NIST calls for a block cipher standard
- 1974 – Horst Feistel at IBM designs *Lucifer*
    - Key-length: 128 bits; block-length: 128 bits
- Lucifer evolves into *DES*
    - Input from the NSA
    - Key-length: 56 bits; block-length: 64 bits
    - #Rounds: 16
- 1976 – Lucifer (now DES) is standardized
- Widely implemented

- 1997 – Broken by exhaustive search
- 2001 – Replaced by AES



FIPS PUB 46

FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION
1977 JANUARY 15

U.S. DEPARTMENT OF COMMERCE /

FIPS

DATA ENCRYPTION STANDARD

CATEGORY: ADP OPERATIONS
SUBCATEGORY: COMPUTER SECURITY

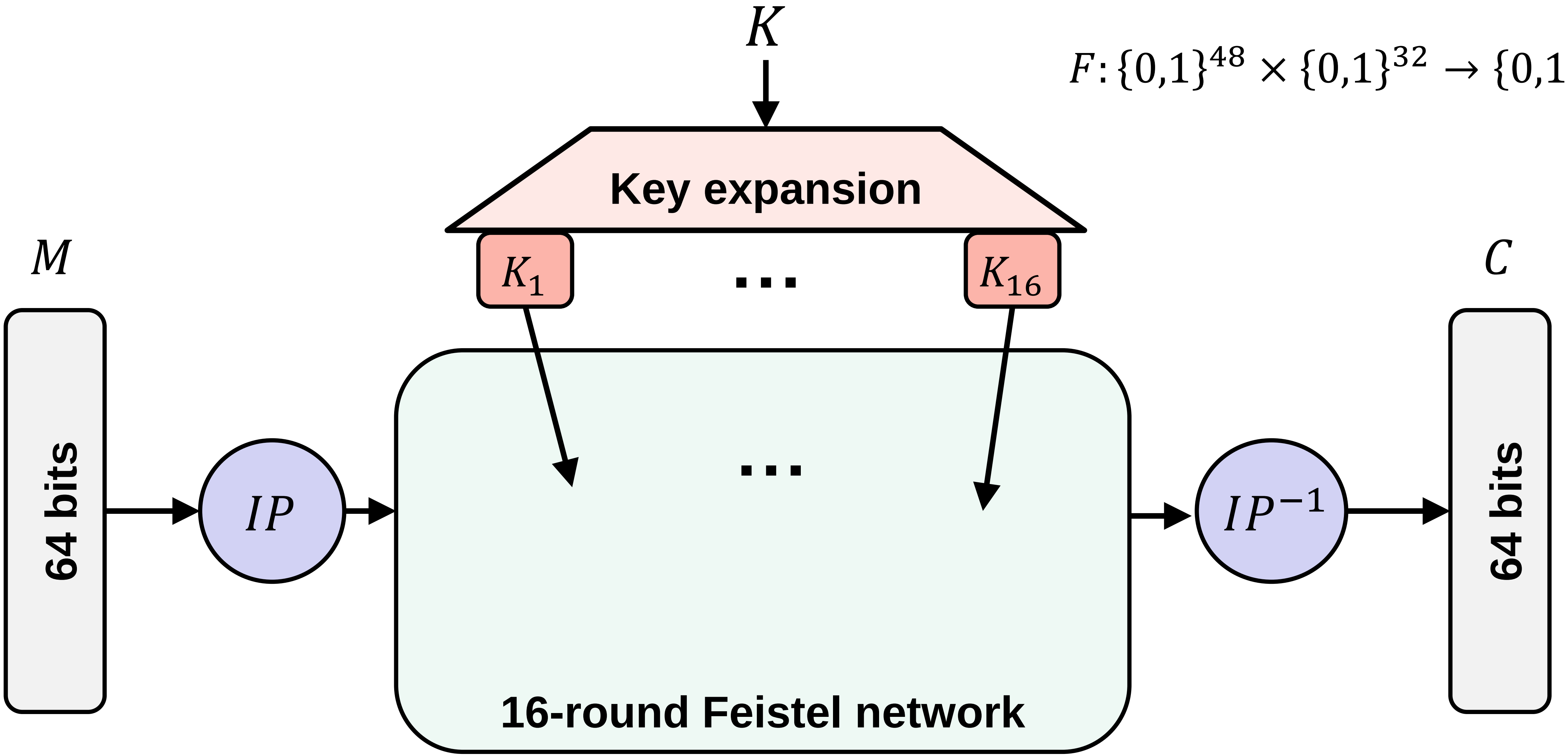
Principles for designing block ciphers

C. Shannon, “Communication Theory of Secrecy Systems”(1949):

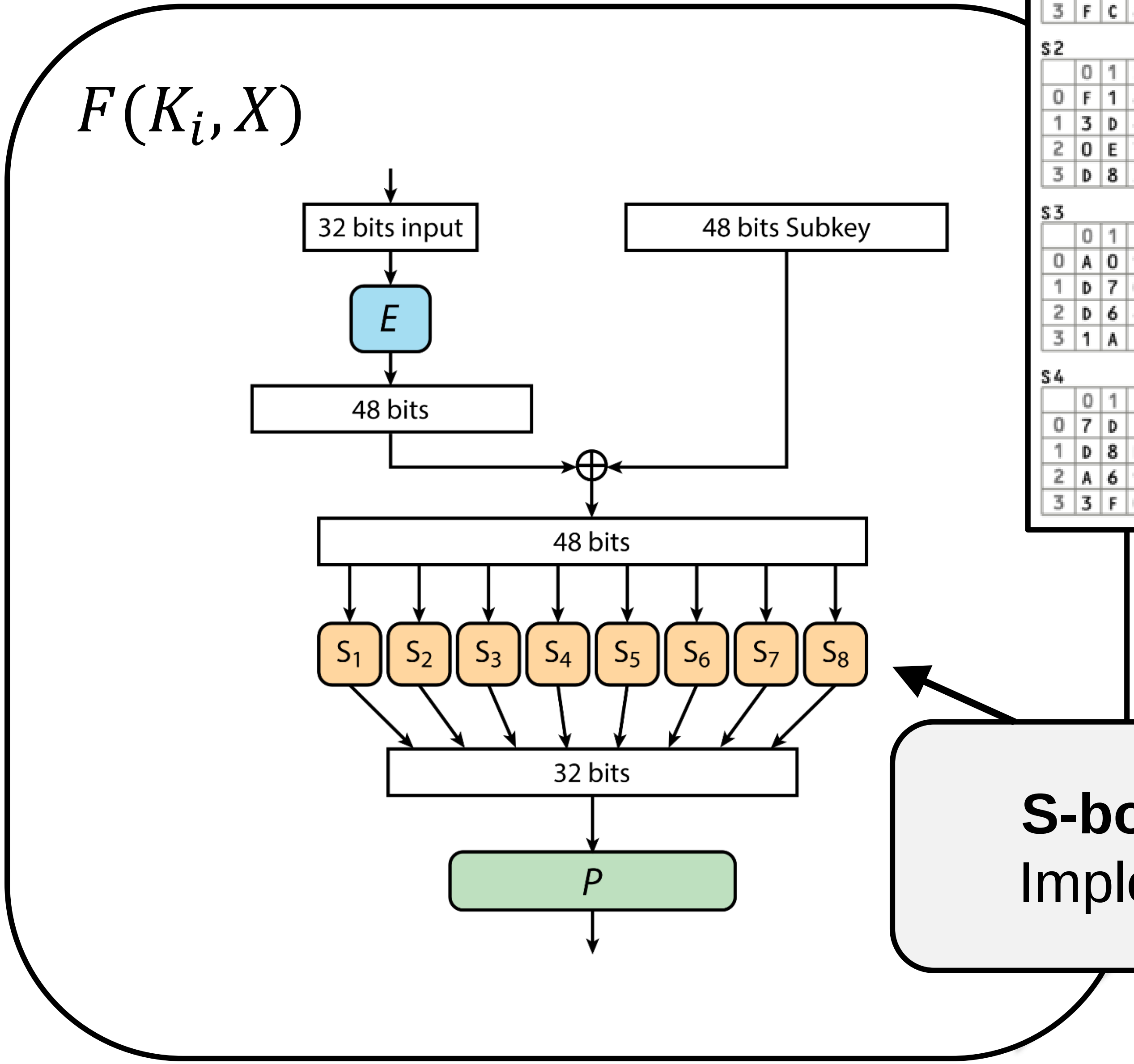
- **Diffusion**: plaintext spread over large parts of the ciphertext
- **Confusion**: a complex relation between plaintext, key and ciphertext

DES

$F: \{0,1\}^{48} \times \{0,1\}^{32} \to \{0,1\}^{32}$



DES round function



S1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7
1	0	F	7	4	E	2	D	1	A	6	C	B	9	5	3	8
2	4	1	E	8	D	6	2	B	F	C	9	7	3	A	5	0
3	F	C	8	2	4	9	1	7	5	B	3	E	A	0	6	D

S2

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	F	1	8	E	6	B	3	4	9	7	2	D	C	0	5	A
1	3	D	4	7	F	2	8	E	C	0	1	A	6	9	B	5
2	0	E	7	B	A	4	D	1	5	8	C	6	9	3	2	F
3	D	8	A	1	3	F	4	2	B	6	7	C	0	5	E	9

S3

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	A	0	9	E	6	3	F	5	1	D	C	7	B	4	2	8
1	D	7	0	9	3	4	6	A	2	8	5	E	C	B	F	1
2	D	6	4	9	8	F	3	0	B	1	2	C	5	A	E	7
3	1	A	D	0	6	9	8	7	4	F	E	3	B	5	2	C

S4

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	7	D	E	3	0	6	9	A	1	2	8	5	B	C	4	F
1	D	8	B	5	6	F	0	3	4	7	2	C	1	A	E	9
2	A	6	9	0	C	B	7	D	F	1	3	E	5	2	8	4
3	3	F	0	6	A	1	D	8	9	4	5	B	C	7	2	E

S5

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	2	C	4	1	7	A	B	6	8	5	3	F	D	0	E	9
1	E	B	2	C	4	7	D	1	5	0	F	A	3	9	8	6
2	4	2	1	B	A	D	7	8	F	9	C	5	6	3	0	E
3	B	8	C	7	1	E	2	D	6	F	0	9	A	4	5	3

S6

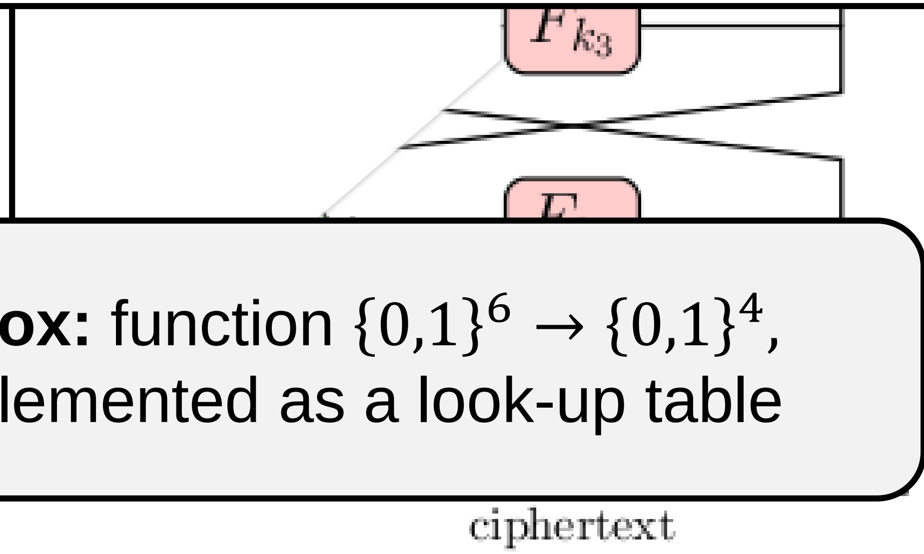
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	C	1	A	F	9	2	6	8	0	D	3	4	E	7	5	B
1	A	F	4	2	7	C	9	5	6	1	D	E	0	B	3	8
2	9	E	F	5	2	8	C	3	7	0	4	A	1	D	B	6
3	4	3	2	C	9	5	F	A	B	E	1	7	6	0	8	D

S7

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	4	B	2	E	F	0	8	D	3	C	9	7	5	A	6	1
1	D	0	B	7	4	9	1	A	E	3	5	C	2	F	8	6
2	1	4	B	D	C	3	7	E	A	F	6	8	0	5	9	2
3	6	B	D	8	1	4	A	7	9	5	0	F	E	2	3	C

S8

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	D	2	8	4	6	F	B	1	A	9	3	E	5	0	C	7
1	1	F	D	8	A	3	7	4	C	5	6	B	0	E	9	2
2	7	B	4	1	9	C	E	2	0	6	A	D	F	3	5	8
3	2	1	E	7	4	A	8	D	F	C	9	0	3	5	6	B



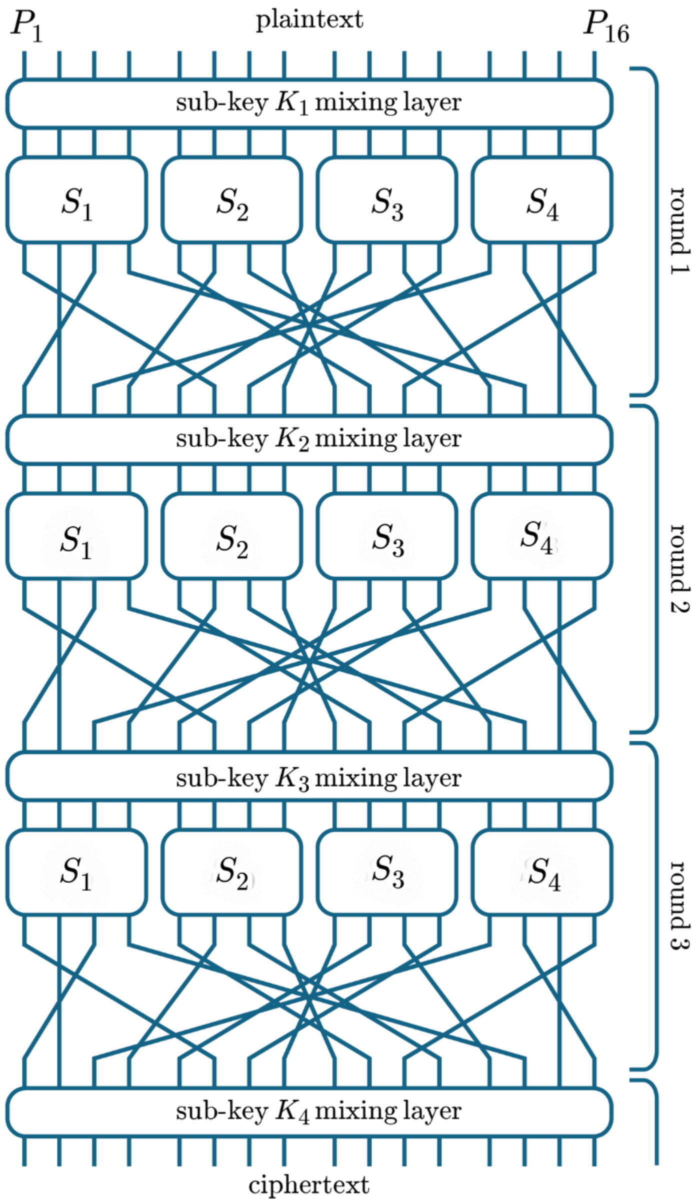
S-box: function $\{0,1\}^6 \rightarrow \{0,1\}^4$,
Implemented as a look-up table

DES properties

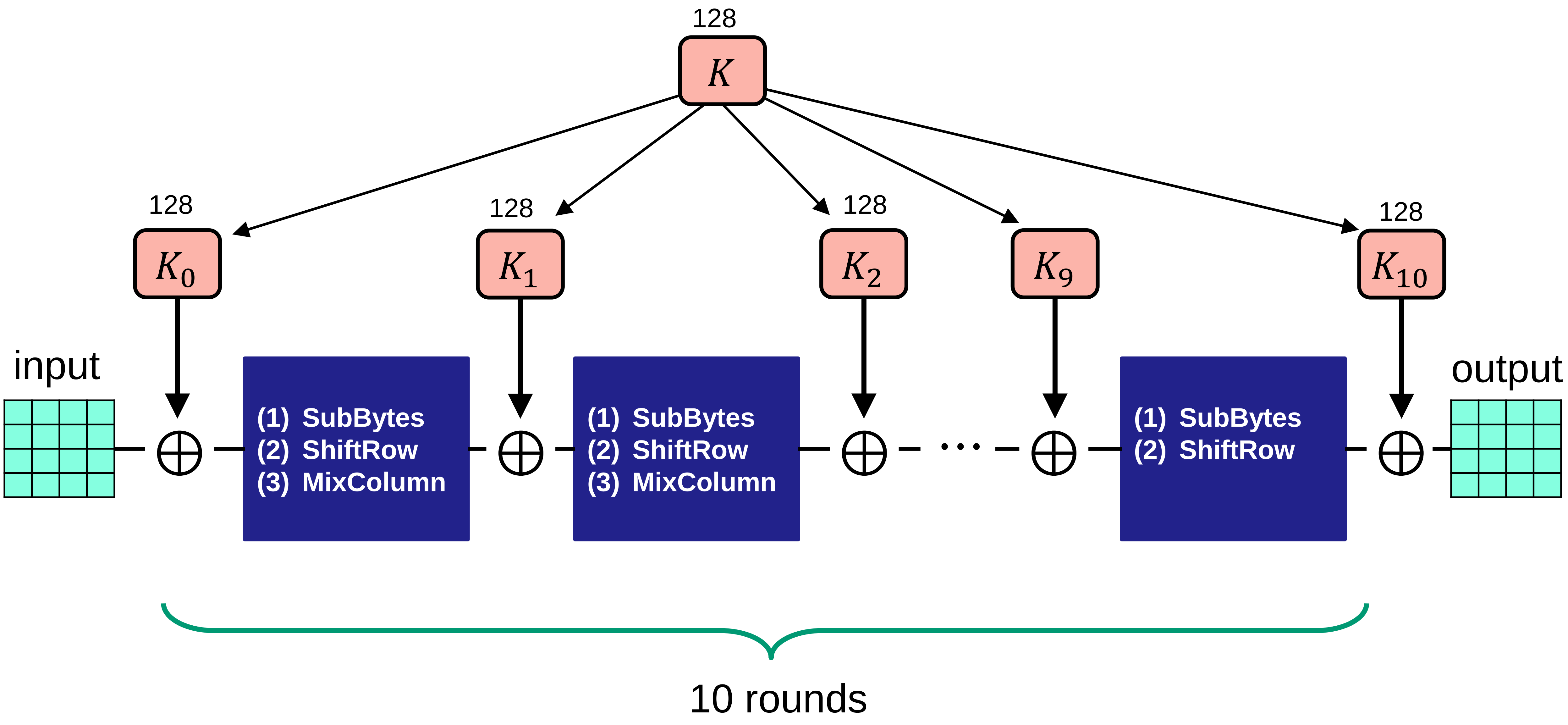
- Easy to implement in hardware
- Not as efficient in software
- Many design decisions still unclear
  - Design criteria classified for many years
  - Controversy around NSA influence
  - Initial S-boxes were changed
  - Switching to 56-bit keys (from 128 bits) probably to allow NSA to decrypt
- **Not secure** since key space and block length too small $\Longrightarrow$ replacement needed

ADVANCED ENCRYPTION STANDARD

Substitution-permutation networks



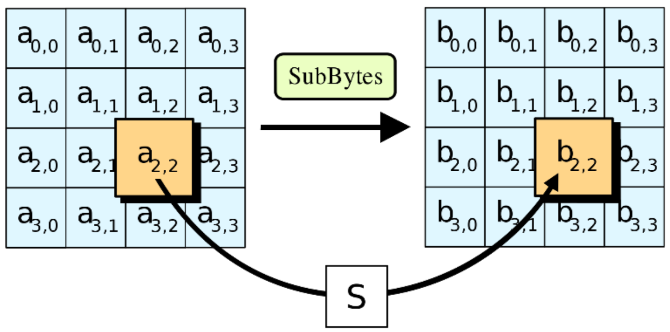
AES-128

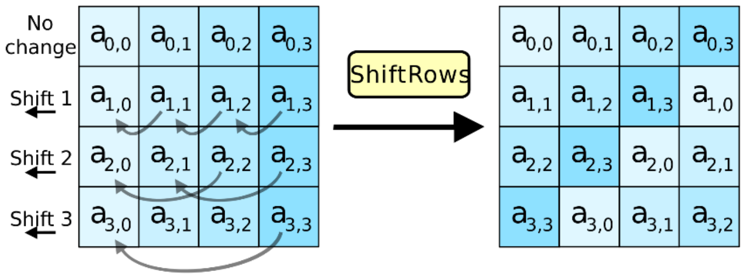


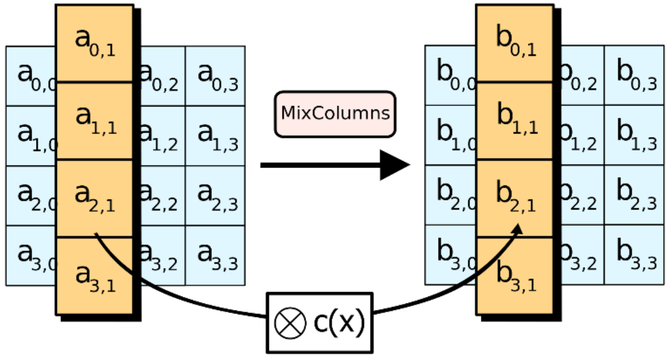
128
K
128
K0
128
K1
128
K2
K9
128
K10
input
(1) SubBytes
(2) ShiftRow
(3) MixColumn
(1) SubBytes
(2) ShiftRow
(3) MixColumn
(1) SubBytes
(2) ShiftRow
output
10 rounds

AES round function

(1) SubBytes
(2) ShiftRow
(3) MixColumn







AES round

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

SubBytes	$b_{i,j} = S[a_{i,j}]$
ShiftRows	$\begin{bmatrix} c_{0,j} \\ c_{1,j} \\ c_{2,j} \\ c_{3,j} \end{bmatrix} = \begin{bmatrix} b_{0,j} \\ b_{1,j-1} \\ b_{2,j-2} \\ b_{3,j-3} \end{bmatrix}$
MixColumns	$\begin{bmatrix} d_{0,j} \\ d_{1,j} \\ d_{2,j} \\ d_{3,j} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} c_{0,j} \\ c_{1,j} \\ c_{2,j} \\ c_{3,j} \end{bmatrix}$
AddRoundKey	$\begin{bmatrix} e_{0,j} \\ e_{1,j} \\ e_{2,j} \\ e_{3,j} \end{bmatrix} = \begin{bmatrix} d_{0,j} \\ d_{1,j} \\ d_{2,j} \\ d_{3,j} \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$

$$\begin{bmatrix} e_{0,j} \\ e_{1,j} \\ e_{2,j} \\ e_{3,j} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S[a_{0,j}] \\ S[a_{1,j-1}] \\ S[a_{2,j-2}] \\ S[a_{3,j-3}] \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

$$= \left(\begin{bmatrix} 02 \\ 01 \\ 01 \\ 03 \end{bmatrix} \cdot S[a_{0,j}] \right) \oplus \left(\begin{bmatrix} 03 \\ 02 \\ 01 \\ 01 \end{bmatrix} \cdot S[a_{1,j-1}] \right)$$

$$\oplus \left(\begin{bmatrix} 01 \\ 03 \\ 02 \\ 01 \end{bmatrix} \cdot S[a_{2,j-2}] \right) \oplus \left(\begin{bmatrix} 01 \\ 01 \\ 03 \\ 02 \end{bmatrix} \cdot S[a_{3,j-3}] \right) \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

$T_0[x] = \left(\begin{bmatrix} 02 \\ 01 \\ 01 \\ 03 \end{bmatrix} \cdot S[x] \right)$	$T_1[x] = \left(\begin{bmatrix} 03 \\ 02 \\ 01 \\ 01 \end{bmatrix} \cdot S[x] \right)$	$T_2[x] = \left(\begin{bmatrix} 01 \\ 03 \\ 02 \\ 01 \end{bmatrix} \cdot S[x] \right)$	$T_3[x] = \left(\begin{bmatrix} 01 \\ 01 \\ 03 \\ 02 \end{bmatrix} \cdot S[x] \right)$

AES round

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

SubBytes	$b_{i,j} = S[a_{i,j}]$
ShiftRows	$\begin{bmatrix} c_{0,j} \\ c_{1,j} \\ c_{2,j} \\ c_{3,j} \end{bmatrix} = \begin{bmatrix} b_{0,j} \\ b_{1,j-1} \\ b_{2,j-2} \\ b_{3,j-3} \end{bmatrix}$
MixColumns	$\begin{bmatrix} d_{0,j} \\ d_{1,j} \\ d_{2,j} \\ d_{3,j} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} c_{0,j} \\ c_{1,j} \\ c_{2,j} \\ c_{3,j} \end{bmatrix}$
AddRoundKey	$\begin{bmatrix} e_{0,j} \\ e_{1,j} \\ e_{2,j} \\ e_{3,j} \end{bmatrix} = \begin{bmatrix} d_{0,j} \\ d_{1,j} \\ d_{2,j} \\ d_{3,j} \end{bmatrix} \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$

$$\begin{bmatrix} e_{0,j} \\ e_{1,j} \\ e_{2,j} \\ e_{3,j} \end{bmatrix} = T_0[a_{0,j}] \oplus T_1[a_{1,j-1}]$$

$$\oplus T_2[a_{2,j-2}] \oplus T_3[a_{3,j-3}] \oplus \begin{bmatrix} k_{0,j} \\ k_{1,j} \\ k_{2,j} \\ k_{3,j} \end{bmatrix}$$

$T_0[x] = \left(\begin{bmatrix} 02 \\ 01 \\ 01 \\ 03 \end{bmatrix} \cdot S[x] \right)$	$T_1[x] = \left(\begin{bmatrix} 03 \\ 02 \\ 01 \\ 01 \end{bmatrix} \cdot S[x] \right)$	$T_2[x] = \left(\begin{bmatrix} 01 \\ 03 \\ 02 \\ 01 \end{bmatrix} \cdot S[x] \right)$	$T_3[x] = \left(\begin{bmatrix} 01 \\ 01 \\ 03 \\ 02 \end{bmatrix} \cdot S[x] \right)$

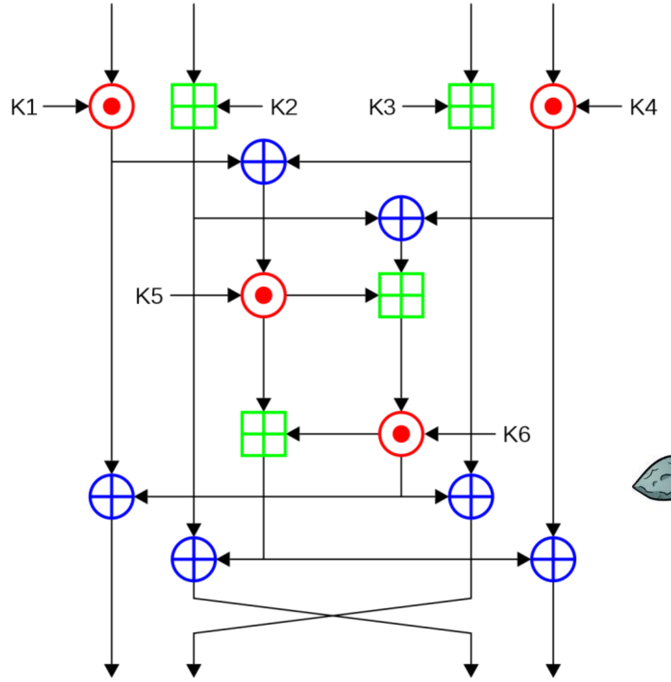
AES performance

- AES is reasonably efficient in software
  - T-table implementation very fast
    (but not secure!)
  - Hard to implement fast and constant-time

	Throughput
AES-128 (in software)	265 MB/s
AES-128 (w/AES-NI)	3.45 GB/s

- Intel introduced dedicated AES instructions into their CPUs (AES-NI):
  - **aesenc**, **aesenclast**: do one round of AES in one cycle
  - **aeskeygenassist:** do AES key expansion
  - **aesdec, aesdeclast:** do one round of AES decryption in one cycle
  - **aesimc:** do AES inverser MixColumns

  - Now standard in all modern CPUs



ATTACKING BLOCK CIPHERS

Attacks on block ciphers

- Brute force attacks: search through every possible key in key space
  - Generic: works for all block ciphers
  - Not practical for large key spaces

- Advanced attacks: try to exploit the concrete details of the block cipher
  - Differential cryptanalysis ('90, but known by the designers of DES + NSA since mid '70)
  - Linear cryptanalysis ('92)
  - AES designed to resist both

- Implementation attacks: vulnerabilities due to implementation characteristics
  - Power draw
  - Timing
  - Cache misses

Summary

- Block ciphers are very important **primitives** (building blocks) – but they are not encryption schemes!

- Correct abstraction: block ciphers = PRPs

- Right security notion for PRFs/PRPs:

  indistinguishability from random function/permutation

- Concrete block cipher designs: DES and AES