PSEUDO-RANDOM FUNCTIONS
Recall

We studied security of function families (in particular, block ciphers) against key recovery.

But we saw that security against key recovery is not sufficient to ensure that natural usages of a block cipher are secure.

We want to answer the question:

What is a good block cipher?

where “good” means that natural uses of the block cipher are secure.

We could try to define “good” by a list of necessary conditions:

• Key recovery is hard
• Recovery of $M$ from $C = E_K(M)$ is hard
• . . .

But this is neither necessarily correct nor appealing.
Q: What does it mean for a program to be “intelligent” in the sense of a human?

Possible answers:

- It can be happy
- It recognizes pictures
- It can multiply
- But only small numbers!

Clearly, no such list is a satisfactory answer to the question.
Q: What does it mean for a program to be “intelligent” in the sense of a
human?

Turing’s answer: A program is intelligent if its input/output behavior is
indistinguishable from that of a human.
Turing Intelligence Test

Behind the wall:
- **Room 1**: The program $P$
- **Room 0**: A human
Turing Intelligence Test

Game:
- Put tester in room 0 and let it interact with object behind wall
- Put tester in room 1 and let it interact with object behind wall
- Now ask tester: which room was which?

The measure of “intelligence” of $P$ is the extent to which the tester fails.
<table>
<thead>
<tr>
<th>Notion</th>
<th>Real object</th>
<th>Ideal object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligence</td>
<td>Program</td>
<td>Human</td>
</tr>
<tr>
<td>PRF</td>
<td>Block cipher</td>
<td>?</td>
</tr>
</tbody>
</table>
## Real versus Ideal

<table>
<thead>
<tr>
<th>Notion</th>
<th>Real object</th>
<th>Ideal object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligence</td>
<td>Program</td>
<td>Human</td>
</tr>
<tr>
<td>PRF</td>
<td>Block cipher</td>
<td>Random function</td>
</tr>
</tbody>
</table>

Nadia Heninger

UCSD
Random functions

Game \texttt{Rand}_R \quad // \text{ here } R \text{ is a set}

\begin{algorithm}
\caption{Fn(x)}
\begin{algorithmic}
\Procedure{Fn}{x}
\If{$T[x] = \bot$}
\State{$T[x] \leftarrow \$ R$}
\EndIf
\State{return $T[x]$}
\EndProcedure
\end{algorithmic}
\end{algorithm}

Adversary $A$

- Make queries to $\texttt{Fn}$
- Eventually halts with some output

We denote by \[
\Pr \left[ \text{Rand}^A_R \Rightarrow d \right]
\]
the probability that $A$ outputs $d$
Random functions

Game $\text{Rand}_{\{0,1\}^3}$

**procedure** $\text{Fn}(x)$

- if $T[x] = \bot$ then $T[x] \leftarrow \{0, 1\}^3$
- return $T[x]$

adversary $A$

$y \leftarrow \text{Fn}(01)$

return $(y = 000)$

$$\Pr \left[ \text{Rand}^A_{\{0,1\}^3} \Rightarrow \text{true} \right] =$$
Game $\text{Rand}_{\{0,1\}^3}$

**procedure** $\text{Fn}(x)$

- if $T[x] = \bot$ then $T[x] \leftarrow \{0, 1\}^3$
- return $T[x]$

adversary $A$

- $y \leftarrow \text{Fn}(01)$
- return $(y = 000)$

$$\Pr\left[\text{Rand}^A_{\{0,1\}^3} \Rightarrow \text{true}\right] = 2^{-3}$$
Game \( \text{Rand}_{\{0,1\}^3} \)

**procedure** \( \text{Fn}(x) \)

if \( T[x] = \bot \) then \( T[x] \leftarrow \{0,1\}^3 \)

return \( T[x] \)

adversary \( A \)
\[
\begin{align*}
  y_1 &\leftarrow \text{Fn}(00) \\
  y_2 &\leftarrow \text{Fn}(11) \\
  \text{return } (y_1 = 010 \land y_2 = 011)
\end{align*}
\]

\[
\Pr \left[ \text{Rand}_{\{0,1\}^3}^A \Rightarrow \text{true} \right] =
\]
Random function

Game \( \text{Rand}_{\{0,1\}^3} \)

procedure \( \text{Fn}(x) \)
if \( T[x] = \bot \) then \( T[x] \leftarrow \{0, 1\}^3 \)
return \( T[x] \)

adversary \( A \)
\( y_1 \leftarrow \text{Fn}(00) \)
\( y_2 \leftarrow \text{Fn}(11) \)
return \((y_1 = 010 \land y_2 = 011)\)

\[
\Pr \left[ \text{Rand}^A_{\{0,1\}^3} \Rightarrow \text{true} \right] = 2^{-6}
\]
Random function

Game $\text{Rand}_{\{0,1\}^3}$

procedure $\text{Fn}(x)$

if $T[x] = \bot$ then $T[x] \leftarrow \{0, 1\}^3$

return $T[x]$

adversary $A$

$y_1 \leftarrow \text{Fn}(00)$

$y_2 \leftarrow \text{Fn}(11)$

return $(y_1 \oplus y_2 = 101)$

$$\Pr \left[ \text{Rand}^A_{\{0,1\}^3} \Rightarrow \text{true} \right] =$$
Random function

Game $\text{Rand}_{\{0,1\}^3}$

procedure $\text{Fn}(x)$

if $T[x] = \bot$ then $T[x] \leftarrow \{0, 1\}^3$

return $T[x]$

adversary $A$

$y_1 \leftarrow \text{Fn}(00)$

$y_2 \leftarrow \text{Fn}(11)$

return $(y_1 \oplus y_2 = 101)$

$$\Pr \left[ \text{Rand}_{\{0,1\}^3}^A \Rightarrow \text{true} \right] = 2^{-3}$$
Recall: Function families

A family of functions (also called a function family) is a two-input function $F : \text{Keys} \times D \rightarrow R$. For $K \in \text{Keys}$ we let $F_K : D \rightarrow R$ be defined by $F_K(x) = F(K, x)$ for all $x \in D$.

Examples:

- **DES:** $\text{Keys} = \{0, 1\}^{56}$, $D = R = \{0, 1\}^{64}$
- **Any block cipher:** $D = R$ and each $F_K$ is a permutation
### Real versus Ideal

<table>
<thead>
<tr>
<th>Notion</th>
<th>Real object</th>
<th>Ideal object</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRF</td>
<td>Family of functions (eg. a block cipher)</td>
<td>Random function</td>
</tr>
</tbody>
</table>

$F$ is a PRF if the input-output behavior of $F_K$ looks to a tester like the input-output behavior of a random function.

Tester does **not** get the key $K$!
Games defining prf advantage of an adversary against $F$

Let $F: \text{Keys} \times D \to R$ be a family of functions.

Associated to $F$, $A$ are the probabilities

\[
\Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] \quad \text{and} \quad \Pr \left[ \text{Rand}_R^A \Rightarrow 1 \right]
\]

that $A$ outputs 1 in each world. The advantage of $A$ is

\[
\text{Adv}^\text{prf}_F (A) = \Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] - \Pr \left[ \text{Rand}_R^A \Rightarrow 1 \right]
\]
PRF advantage

<table>
<thead>
<tr>
<th>A’s output $d$</th>
<th>Intended meaning: I think I am in game</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Real</td>
</tr>
<tr>
<td>0</td>
<td>Random</td>
</tr>
</tbody>
</table>

$\text{Adv}_{F}^{\text{prf}}(A) \approx 1$ means $A$ is doing well and $F$ is not prf-secure.
$\text{Adv}_{F}^{\text{prf}}(A) \approx 0$ (or $\leq 0$) means $A$ is doing poorly and $F$ resists the attack $A$ is mounting.
PRF security

Adversary advantage depends on its

- strategy
- resources: Running time $t$ and number $q$ of oracle queries

**Security:** $F$ is a (secure) PRF if $\text{Adv}^{\text{prf}}_F(A)$ is “small” for ALL $A$ that use “practical” amounts of resources.

**Example:** 80-bit security could mean that for all $n = 1, \ldots, 80$ we have

$$\text{Adv}^{\text{prf}}_F(A) \leq 2^{-n}$$

for any $A$ with time and number of oracle queries at most $2^{80-n}$.

**Insecurity:** $F$ is insecure (not a PRF) if we can specify an $A$ using “few” resources that achieves “high” advantage.
Example

Define $F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ by $F_K(x) = K \oplus x$ for all $K, x \in \{0, 1\}^\ell$. Is $F$ a secure PRF?

<table>
<thead>
<tr>
<th>Game $\text{Real}_F$</th>
<th>Game $\text{Rand}_{{0,1}^\ell}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>procedure Initialize</strong></td>
<td><strong>procedure $\text{Fn}(x)$</strong></td>
</tr>
<tr>
<td>$K \leftarrow {0, 1}^\ell$</td>
<td>if $T[x] = \bot$ then $T[x] \leftarrow {0, 1}^\ell$</td>
</tr>
<tr>
<td><strong>procedure $\text{Fn}(x)$</strong></td>
<td>Return $T[x]$</td>
</tr>
<tr>
<td>Return $K \oplus x$</td>
<td></td>
</tr>
</tbody>
</table>

So we are asking: Can we design a low-resource $A$ so that

$$\text{Adv}^{\text{prf}}_F (A) = \Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] - \Pr \left[ \text{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1 \right]$$

is close to 1?
Example

Define $F$: $\{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ by $F_K(x) = K \oplus x$ for all $K, x \in \{0, 1\}^\ell$. Is $F$ a secure PRF?

So we are asking: Can we design a low-resource $A$ so that

$$\text{Adv}_{prf}^F(A) = \Pr \left[ \text{Real}_A^F \Rightarrow 1 \right] - \Pr \left[ \text{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1 \right]$$

is close to 1?

Exploitable weakness of $F$: For all $K$ we have

$$F_K(0^\ell) \oplus F_K(1^\ell) = (K \oplus 0^\ell) \oplus (K \oplus 1^\ell) = 1^\ell$$
Example: The adversary

\[ F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell \text{ is defined by } F_K(x) = K \oplus x. \]

adversary \( A \)

if \( F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell \) then return 1 else return 0
Example: Real game analysis

\( F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell \) is defined by \( F_K(x) = K \oplus x \).

**adversary** \( A \)

if \( F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell \) then return 1 else return 0

---

**Game Real_\( F \)**

**procedure Initialize**

\( K \leftarrow \{0, 1\}^\ell \)

**procedure F_n(x)**

Return \( K \oplus x \)

\[
\Pr \left[ \text{Real}_{\mathcal{F}}^A \rightarrow 1 \right] =
\]
Example: Real game analysis

$F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ is defined by $F_K(x) = K \oplus x$.

adversary $A$

if $F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell$ then return 1 else return 0

Game Real$_F$

procedure Initialize

$K \leftarrow \{0, 1\}^\ell$

procedure $F_n(x)$

Return $K \oplus x$

$$Pr \left[ \text{Real}_F^A \rightarrow 1 \right] = 1$$

because

$$F_n(0^\ell) \oplus F_n(1^\ell) = F_K(0^\ell) \oplus F_K(1^\ell) = (K \oplus 0^\ell) \oplus (K \oplus 1^\ell) = 1^\ell$$
Example: Rand game analysis

$F: \{0, 1\}^\ell \times \{0, 1\}^\ell \to \{0, 1\}^\ell$ is defined by $F_K(x) = K \oplus x$.

adversary $A$

if $F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell$ then return 1 else return 0

Game $\text{Rand}_{\{0,1\}^\ell}$

procedure $F_n(x)$

if $T[x] = \bot$ then $T[x] \leftarrow \{0, 1\}^\ell$

Return $T[x]$

$\Pr \left[ \text{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1 \right] = \ldots$
Example: Rand game analysis

\[ F: \{0, 1\}^\ell \times \{0, 1\}^\ell \to \{0, 1\}^\ell \] is defined by \( F_K(x) = K \oplus x \).

**adversary** \( A \)

if \( F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell \) then return 1 else return 0

---

**Game** \( \text{Rand}_{\{0,1\}^\ell} \)

**procedure** \( F_n(x) \)

if \( T[x] = \bot \) then \( T[x] \leftarrow \$ \{0, 1\}^\ell \)

Return \( T[x] \)

\[
\Pr \left[ \text{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1 \right] = \Pr \left[ F_n(1^\ell) \oplus F_n(0^\ell) = 1^\ell \right] =
\]
Example: Rand game analysis

\[ F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell \text{ is defined by } F_K(x) = K \oplus x. \]

**adversary** \( A \)

if \( Fn(0^\ell) \oplus Fn(1^\ell) = 1^\ell \) then return 1 else return 0

\[
\text{Game } \text{Rand}_{\{0,1\}^\ell} \\
\text{procedure } Fn(x) \\
\text{if } T[x] = \perp \text{ then } T[x] \leftarrow \{0, 1\}^\ell \\
\text{Return } T[x]
\]

\[
\Pr \left[ \text{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1 \right] = \Pr \left[ Fn(1^\ell) \oplus Fn(0^\ell) = 1^\ell \right] = 2^{-\ell}
\]
because \( Fn(0^\ell), Fn(1^\ell) \) are random \( \ell \)-bit strings.
Example: Conclusion

\( F: \{0,1\}^\ell \times \{0,1\}^\ell \rightarrow \{0,1\}^\ell \) is defined by \( F_K(x) = K \oplus x \).

**adversary** \( A \)

if \( F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell \) then return 1 else return 0

Then

\[
\text{Adv}^\text{prf}_F(A) = \Pr[\text{Real}^A_F \Rightarrow 1] - \Pr[\text{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1]
\]

\[
= 1 - 2^{-\ell}
\]

and \( A \) is efficient.

**Conclusion:** \( F \) is not a secure PRF.
Birthday Problem

We have \( q \) people \( 1, \ldots, q \) with birthdays \( y_1, \ldots, y_q \in \{1, \ldots, 365\} \). Assume each person’s birthday is a random day of the year. Let

\[
C(365, q) = \Pr[2 \text{ or more persons have same birthday}]
= \Pr[y_1, \ldots, y_q \text{ are not all different}]
\]

- What is the value of \( C(365, q) \)?
- How large does \( q \) have to be before \( C(365, q) \) is at least \( 1/2 \)?
Birthday Problem

We have \( q \) people \( 1, \ldots, q \) with birthdays \( y_1, \ldots, y_q \in \{1, \ldots, 365\} \). Assume each person’s birthday is a random day of the year. Let

\[
C(365, q) = \Pr \left[ \text{2 or more persons have same birthday} \right]
\]

\[
= \Pr \left[ y_1, \ldots, y_q \text{ are not all different} \right]
\]

• What is the value of \( C(365, q) \)?
• How large does \( q \) have to be before \( C(365, q) \) is at least \( 1/2 \)?

Naive intuition:
• \( C(365, q) \approx q/365 \)
• \( q \) has to be around 365
Birthday Problem

We have \( q \) people \( 1, \ldots, q \) with birthdays \( y_1, \ldots, y_q \in \{1, \ldots, 365\} \). Assume each person’s birthday is a random day of the year. Let

\[
C(365, q) = \Pr \{\text{2 or more persons have same birthday}\} \\
= \Pr \{y_1, \ldots, y_q \text{ are not all different}\}
\]

• What is the value of \( C(365, q) \)?
• How large does \( q \) have to be before \( C(365, q) \) is at least 1/2?

Naive intuition:
• \( C(365, q) \approx q/365 \)
• \( q \) has to be around 365

The reality
• \( C(365, q) \approx q^2/365 \)
• \( q \) has to be only around 23
Birthday collision bounds

$C(365, q)$ is the probability that some two people have the same birthday in a room of $q$ people with random birthdays.

<table>
<thead>
<tr>
<th>$q$</th>
<th>$C(365, q)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.253</td>
</tr>
<tr>
<td>18</td>
<td>0.347</td>
</tr>
<tr>
<td>20</td>
<td>0.411</td>
</tr>
<tr>
<td>21</td>
<td>0.444</td>
</tr>
<tr>
<td>23</td>
<td>0.507</td>
</tr>
<tr>
<td>25</td>
<td>0.569</td>
</tr>
<tr>
<td>27</td>
<td>0.627</td>
</tr>
<tr>
<td>30</td>
<td>0.706</td>
</tr>
<tr>
<td>35</td>
<td>0.814</td>
</tr>
<tr>
<td>40</td>
<td>0.891</td>
</tr>
<tr>
<td>50</td>
<td>0.970</td>
</tr>
</tbody>
</table>
Pick $y_1, \ldots, y_q \leftarrow \{1, \ldots, N\}$ and let

$$C(N, q) = \Pr [y_1, \ldots, y_q \text{ not all distinct}]$$

Birthday setting: $N = 365$
Birthday Problem

Pick $y_1, \ldots, y_q \leftarrow \{1, \ldots, N\}$ and let

$$C(N, q) = \Pr [y_1, \ldots, y_q \text{ not all distinct}]$$

Birthday setting: $N = 365$

Fact: $C(N, q) \approx \frac{q^2}{2N}$
Birthday collisions formula

Let $y_1, \ldots, y_q \leftarrow \{1, \ldots, N\}$. Then

$$1 - C(N, q) = \Pr[y_1, \ldots, y_q \text{ all distinct}]$$

$$= 1 \cdot \frac{N - 1}{N} \cdot \frac{N - 2}{N} \cdots \frac{N - (q - 1)}{N}$$

$$= \prod_{i=1}^{q-1} \left(1 - \frac{i}{N}\right)$$

SO

$$C(N, q) = 1 - \prod_{i=1}^{q-1} \left(1 - \frac{i}{N}\right)$$
Birthday bounds

Let

\[ C(N, q) = \Pr [y_1, \ldots, y_q \text{ not all distinct}] \]

Fact: Then

\[ 0.3 \cdot \frac{q(q - 1)}{N} \leq C(N, q) \leq 0.5 \cdot \frac{q(q - 1)}{N} \]

where the lower bound holds for \( 1 \leq q \leq \sqrt{2N} \).
Let $E : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ be a block cipher.

Can we design $A$ so that

$$\text{Adv}_{E}^{\text{prf}}(A) = \Pr[\text{Real}_E \Rightarrow 1] - \Pr[\text{Rand}_{\{0,1\}^\ell} \Rightarrow 1]$$

is close to 1?
Defining property of a block cipher: $E_K$ is a permutation for every $K$.

So if $x_1, \ldots, x_q$ are distinct then

- $F_n = E_K \Rightarrow F_n(x_1), \ldots, F_n(x_q)$ distinct
- $F_n$ random $\Rightarrow F_n(x_1), \ldots, F_n(x_q)$ not necessarily distinct

This leads to the following attack:

**adversary $A$**

Let $x_1, \ldots, x_q \in \{0, 1\}^\ell$ be distinct
for $i = 1, \ldots, q$ do $y_i \leftarrow F_n(x_i)$
if $y_1, \ldots, y_q$ are all distinct then return 1
else return 0
Real world analysis

Let $E : \{0,1\}^k \times \{0,1\}^\ell \rightarrow \{0,1\}^\ell$ be a block cipher

**Game Real$_E$**

**procedure Initialize**

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

**procedure Fn($x$)**

Return $E_K(x)$

**adversary $A$**

Let $x_1, \ldots, x_q \in \{0,1\}^\ell$ be distinct

for $i = 1, \ldots, q$ do $y_i \leftarrow \text{Fn}(x_i)$

if $y_1, \ldots, y_q$ are all distinct

then return 1 else return 0

Then

$$\Pr \left[ \text{Real}_E^A \Rightarrow 1 \right] =$$
Let $E : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ be a block cipher

**Game Real$_E$**

**procedure Initialize**

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

**procedure Fn($x$)**

Return $E_K(x)$

**adversary $A$**

Let $x_1, \ldots, x_q \in \{0, 1\}^\ell$ be distinct

for $i = 1, \ldots, q$ do $y_i \leftarrow Fn(x_i)$

if $y_1, \ldots, y_q$ are all distinct

then return 1 else return 0

Then

$$\Pr \left[ \text{Real}_E \Rightarrow 1 \right] = 1$$

because $y_1, \ldots, y_q$ will be distinct because $E_K$ is a permutation.
Let \( E : \{0, 1\}^K \times \{0, 1\}^\ell \to \{0, 1\}^\ell \) be a block cipher

**Game \( \text{Rand}_{\{0,1\}^\ell} \)**

**procedure \( \text{Fn}(x) \)**

if \( T[x] = \perp \) then \( T[x] \leftarrow \$ \{0, 1\}^\ell \)
Return \( T[x] \)

**adversary \( A \)**

Let \( x_1, \ldots, x_q \in \{0, 1\}^\ell \) be distinct for \( i = 1, \ldots, q \) do \( y_i \leftarrow \text{Fn}(x_i) \)
if \( y_1, \ldots, y_q \) are all distinct then return 1 else return 0

Then
\[
\Pr \left[ \text{Rand}_{\{0,1\}^\ell} \Rightarrow 1 \right] = \Pr [y_1, \ldots, y_q \text{ all distinct}] = 1 - C(2^\ell, q)
\]
because \( y_1, \ldots, y_q \) are randomly chosen from \( \{0, 1\}^\ell \).
Birthday attack on a block cipher

Let $x_1, \ldots, x_q \in \{0, 1\}^\ell$ be distinct
for $i = 1, \ldots, q$ do $y_i \leftarrow \text{Fn}(x_i)$
if $y_1, \ldots, y_q$ are all distinct then return 1 else return 0

$$\text{Adv}^{\text{prf}}_E(A) = \Pr[\text{Real}_E^A \Rightarrow 1] - \Pr[\text{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1]$$

$$= C(2^\ell, q) \geq 0.3 \cdot \frac{q(q-1)}{2^\ell}$$

SO

$$q \approx 2^{\ell/2} \Rightarrow \text{Adv}^{\text{prf}}_E(A) \approx 1.$$
Birthday attack on a block cipher

Conclusion: If $E : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ is a block cipher, there is an attack on it as a PRF that succeeds in about $2^{\ell/2}$ queries.

 Depends on block length, not key length!

<table>
<thead>
<tr>
<th></th>
<th>$\ell$</th>
<th>$2^{\ell/2}$</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>DES, 2DES, 3DES3</td>
<td>64</td>
<td>$2^{32}$</td>
<td>Insecure</td>
</tr>
<tr>
<td>AES</td>
<td>128</td>
<td>$2^{64}$</td>
<td>Secure</td>
</tr>
</tbody>
</table>
We have seen two possible metrics of security for a block cipher $E$

- **(T)KR-security**: It should be hard to find the target key, or a key consistent with input-output examples of a hidden target key.
- **PRF-security**: It should be hard to distinguish the input-output behavior of $E_K$ from that of a random function.

**Fact**: PRF-security of $E$ implies

- KR (and hence TKR) security of $E$
- Many other security attributes of $E$

This is a validation of the choice of PRF security as our main metric.
Our Assumptions

DES, AES are good block ciphers in the sense that they are PRF-secure up to the inherent limitations of the birthday attack and known key-recovery attacks.

You can assume this in designs and analyses.

But beware that the future may prove these assumptions wrong!