PSEUDO-RANDOM FUNCTIONS
We studied security of a block cipher 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?
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!
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.
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?
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.
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?

Clarification: Room numbers are in our head, not written on door!
## 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>?</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>
A random function with $L$-bit outputs is implemented by the following box $\text{Fn}$, where $T$ is initially $\bot$ everywhere:

\[
\text{Fn}
\]

If $T[x] = \bot$ then
\[
T[x] \leftarrow \{0, 1\}^L
\]
Return $T[x]$
Random function

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

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

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

return $T[x]$

Adversary $A$

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

We denote by

$$\Pr \left[ \text{Rand}^A_{\{0,1\}} \Rightarrow d \right]$$

the probability that $A$ outputs $d$
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 \( T[x] \)

\begin{align*}
\text{adversary } A \\
y & \leftarrow \text{Fn}(01) \\
\text{return } (y = 000)
\end{align*}

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

procedure $\text{Fn}(x)$
if $T[x] = \perp$ 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 $(y_1 = 010 \land y_2 = 011)$

$$\Pr \left[ \text{Rand}_A^{0,1}_3 \Rightarrow \text{true} \right] =$$
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 $(y_1 = 010 \land y_2 = 011)$

$$\Pr \left[ \text{Rand}_{\{0,1\}^3} \Rightarrow \text{true} \right] = 2^{-6}$$
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 $(y_1 \oplus y_2 = 101)$

$$\Pr \left[ \text{Rand}_{\{0,1\}^3}^A \Rightarrow \text{true} \right] = \frac{13}{65}$$
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 $(y_1 \oplus y_2 = 101)$}

$$\Pr \left[ \text{Rand}^A_{\{0,1\}^3} \Rightarrow \text{true} \right] = 2^{-3}$$
A family of functions $F : \text{Keys}(F) \times \text{Dom}(F) \to \text{Range}(F)$ is a two-argument map. For $K \in \text{Keys}(F)$ we let $F_K : \text{Dom}(F) \to \text{Range}(F)$ be defined by

$$\forall x \in \text{Dom}(F) : F_K(x) = F(K, x)$$

Examples:

- DES: $\text{Keys}(F) = \{0, 1\}^{56}$, $\text{Dom}(F) = \text{Range}(F) = \{0, 1\}^{64}$
- Any block cipher: $\text{Dom}(F) = \text{Range}(F)$ 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 $K$!
Let $F: \text{Keys}(F) \times \text{Dom}(F) \rightarrow \text{Range}(F)$ be a family of functions.

A prf-adversary (our tester) has an oracle $F_n$ for a function from $\text{Dom}(F)$ to $\text{Range}(F)$. It can

- Make an oracle query $x$ of its choice and get back $F_n(x)$
- Do this many times
- Eventually halt and output a bit $d$
Repeat queries

We said earlier that a random function must be consistent, meaning once it has returned $y$ in response to $x$, it must return $y$ again if queried again with the same $x$. This is why we have the “if” in the following: written as

<table>
<thead>
<tr>
<th>Game</th>
<th>procedure $F_n(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand $_{Range(F)}$</td>
<td>if $T[x] \neq \perp$ then $T[x] \leftarrow$ Range($F$)</td>
</tr>
<tr>
<td></td>
<td>Return $T[x]$</td>
</tr>
</tbody>
</table>

Henceforth we make a rule:

- A prf-adversary is not allowed to repeat an oracle query.

Then our game is:

<table>
<thead>
<tr>
<th>Game</th>
<th>procedure $F_n(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand $_{Range(F)}$</td>
<td>$T[x] \leftarrow$ Range($F$)</td>
</tr>
<tr>
<td></td>
<td>Return $T[x]$</td>
</tr>
</tbody>
</table>
Let $F: \text{Keys}(F) \times \text{Dom}(F) \rightarrow \text{Range}(F)$ be a family of functions.

### Intended meaning:

<table>
<thead>
<tr>
<th>$A$'s output $d$</th>
<th>I think I am in the</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Real world</td>
</tr>
<tr>
<td>0</td>
<td>Ideal (Random) world</td>
</tr>
</tbody>
</table>

The harder it is for $A$ to guess world it is in, the “better” $F$ is as a PRF.
The games

Let $F: \text{Keys}(F) \times \text{Dom}(F) \to \text{Range}(F)$ be a family of functions.

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

$$\Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] \quad \bigg| \quad \Pr \left[ \text{Rand}_{\text{Range}(F)}^A \Rightarrow 1 \right]$$

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

$$\text{Adv}^\text{prf}_F (A) = \Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] - \Pr \left[ \text{Rand}_{\text{Range}(F)}^A \Rightarrow 1 \right]$$
Let $F: \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ be defined by $F_K(x) = x$. Let prf-adversary $A$ be defined by

**adversary $A$**

if $F_n(0^{128}) = 0^{128}$ then Ret 1 else Ret 0

---

**Game $\text{Real}_F$**

**procedure Initialize**

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

**procedure $F_n(x)$**

Return $F_K(x)$

---

**Real world**

$A$

$x$

$y$

$y \leftarrow F_K(x)$
Example

Let $F: \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ be defined by $F_K(x) = x$. Let prf-adversary $A$ be defined by

**adversary $A$**

if $F_n(0^{128}) = 0^{128}$ then Ret 1 else Ret 0

---

**Game $\text{Real}_F$**

```plaintext
procedure Initialize
$K \leftarrow \{0, 1\}^k$

procedure $F_n(x)$
Return $F_K(x)$
```

Then

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

Let $F: \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ be defined by $F_K(x) = x$. Let prf-adversary $A$ be defined by

adversary $A$

if $F_n(0^{128}) = 0^{128}$ then Ret 1 else Ret 0

Game $\text{Real}_F$

procedure Initialize

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

procedure $F_n(x)$

Return $F_K(x)$

Then

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

because the value returned by $F_n$ will be $F_n(0^{128}) = F_K(0^{128}) = 0^{128}$ so $A$ will always return 1.
Let \( F : \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128} \) be defined by \( F_K(x) = x \). Let prf-adversary \( A \) be defined by

\[
\text{adversary } A
\]

if \( F_n(0^{128}) = 0^{128} \) then Ret 1 else Ret 0

Then

\[
\Pr \left[ \text{Rand}^A_{\text{Range}(F)} \Rightarrow 1 \right] =
\]
Let $F: \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ be defined by $F_K(x) = x$. Let prf-adversary $A$ be defined by

**adversary $A$**

if $F_0(0^{128}) = 0^{128}$ then Ret 1 else Ret 0

Then

$$\Pr \left[ \text{Rand}_{\text{Range}(F)}^A \Rightarrow 1 \right] = \Pr \left[ F_0(0^{128}) = 0^{128} \right] = 2^{-128}$$

because $F_0(0^{128})$ is a random 128-bit string.
Let $F: \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ be defined by $F_K(x) = x$. Let prf-adversary $A$ be defined by

**adversary $A$**

if $F(0^{128}) = 0^{128}$ then Ret 1 else Ret 0

Then

$$Adv_F^{prf}(A) = \Pr\left[\text{Real}_F^A \Rightarrow 1\right] - \Pr\left[\text{Rand}^A_{\text{Range}(F)} \Rightarrow 1\right]$$

$$= 1 - 2^{-128}$$
The measure of success

Let \( F : \text{Keys}(F) \times \text{Domain}(F) \to \text{Range}(F) \) be a family of functions and \( A \) a prf adversary. Then

\[
\text{Adv}^\text{prf}_F(A) = \Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] - \Pr \left[ \text{Rand}_{\text{Range}(F)}^A \Rightarrow 1 \right]
\]

is a number between \(-1\) and 1.

A “large” (close to 1) advantage means

- \( A \) is doing well
- \( F \) is not secure

A “small” (close to 0 or \( \leq 0 \)) advantage means

- \( A \) is doing poorly
- \( F \) 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 there exists $A$ using “few” resources that achieves “high” advantage.
Define $F : \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ by $F_K(x) = x$ for all $k, x$. Is $F$ a secure PRF?

Can we design $A$ so that

$$\text{Adv}^{\text{prf}}_F (A) = \text{Pr}[\text{Real}_F^{A} \Rightarrow 1] - \text{Pr}[\text{Rand}^{A}_{\text{Range}(F)} \Rightarrow 1]$$

is close to 1?
Define $F : \{0, 1\}^k \times \{0, 1\}^{128} \rightarrow \{0, 1\}^{128}$ by $F_K(x) = x$ for all $k, x$.

Is $F$ a secure PRF?

Can we design $A$ so that

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

is close to 1?

Exploitable weakness of $F$: $F_K(0^{128}) = 0^{128}$ for all $K$. We can determine which world we are in by testing whether $F_n(0^{128}) = 0^{128}$. 
Example 1

Now $F$ is defined by $F_K(x) = x$.

adversary $A$

if $F_n(0^{128}) = 0^{128}$ then return 1 else return 0
Example 1: Analysis

$F$ is defined by $F_K(x) = x$.

adversary $A$

if $F_n(0^{128}) = 0^{128}$ then return 1 else return 0

We already analysed this and saw that

$$\Pr[\text{Real}_F \Rightarrow 1] = 1$$

$$\Pr[\text{Rand}_{\text{Range}(F)} \Rightarrow 1] = 2^{-128}$$
Example 1: Conclusion

$F$ is defined by $F_K(x) = x$.

**adversary $A$**

if $F_n(0^{128}) = 0^{128}$ then return 1 else return 0

Then

$$\text{Adv}_{F}^{\text{prf}}(A) = \Pr[\text{Real}_F^A \Rightarrow 1] - \Pr[\text{Rand}^A_{\text{Range}(F)} \Rightarrow 1]$$

$$= 1 - 2^{-128}$$

and $A$ is efficient.

**Conclusion:** $F$ is not a secure PRF.
Define $F: \{0, 1\}^\ell \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ by $F_K(x) = K \oplus x$ for all $K, x$. Is $F$ a secure PRF?

Can we design $A$ so that

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

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

Can we design $A$ so that

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

is close to 1?

Exploitable weakness of $F$:

$$F_K(0^\ell) \oplus F_K(1^\ell) = (K \oplus 0^\ell) \oplus (K \oplus 1^\ell) = 1^\ell$$

for all $K$. We can determine which world we are in by testing whether

$$\text{Fn}(0^\ell) \oplus \text{Fn}(1^\ell) = 1^\ell.$$
Example 2: 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 2: Real world 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 \( \text{Real}_F \)

procedure Initialize
\( K \leftarrow \{0, 1\}^k \)

procedure \( F_n(x) \)
Return \( F_K(x) \)

Real world

\[ A \]
\[ x \]
\[ \stackrel{\leftarrow}{y} \]
\[ F_n \]
\[ y \leftarrow F_K(x) \]
Example 2: Real world analysis

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

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

Game $\text{Real}_F$

\begin{align*}
\text{procedure Initialize} \\
K \leftarrow \{0, 1\}^k \\
\text{procedure } F_n(x) \\
\text{Return } F_K(x)
\end{align*}

Then

$$\Pr\left[\text{Real}_F^A \Rightarrow 1\right] =$$
Example 2: Real world 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

---

**Game Real\(F\)**

**procedure Initialize**

\[
K \leftarrow \{0, 1\}^k
\]

**procedure \(Fn(x)\)**

Return \(F_K(x)\)

---

Then

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

because

\[
Fn(0^\ell) \oplus Fn(1^\ell) = F_K(0^\ell) \oplus F_K(1^\ell) = (K \oplus 0^\ell) \oplus (K \oplus 1^\ell) = 1^\ell
\]
Example 2: Ideal world 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 \( \text{Rand}_{\text{Range}}(F) \)

procedure \( F_n(x) \)
\( T[x] \leftarrow \{0, 1\}^\ell \) return \( T[x] \)

Ideal (random) world
Example 2: Ideal world 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

Then

$$\Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] =$$
Example 2: Ideal world 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 \( F_n(0^\ell) \oplus F_n(1^\ell) = 1^\ell \) then return 1 else return 0

---

**Game** \( \text{Rand}_{\text{Range}}(F) \)

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

\[ T[x] \leftarrow \{0, 1\}^\ell \text{ return } T[x] \]

---

Then

\[ \Pr \left[ \text{Real}_F^{A} \Rightarrow 1 \right] = \Pr \left[ F_n(1^\ell) \oplus F_n(0^\ell) = 1^\ell \right] = \]
Example 2: Ideal world 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

Then

\[
\Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] = \Pr \left[ F_n(1^\ell) \oplus F_n(0^\ell) = 1^\ell \right] = 2^{-\ell}
\]

because \( F_n(0^\ell), F_n(1^\ell) \) are random \( \ell \)-bit strings.
Example 2: 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

\[
\text{Adv}^\text{prf}_F(A) = \Pr[\text{Real}_F^A \Rightarrow 1] - \Pr[\text{Rand}_{\text{Range}(F)}^A \Rightarrow 1] = 1 - 2^{-\ell}
\]

and \(A\) is efficient.

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

$q$ 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**

$q$ 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 $1/2$?

**Naive intuition:**

- $C(365, q) \approx q/365$
- $q$ has to be around 365
Birthday Problem

$q$ 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?

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>
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: $N = 365$
Birthday Problem

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

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

Birthday setting: $N = 365$

Fact: $C(N, q) \approx \frac{q^2}{2N}$
Let \( y_1, \ldots, y_q \leftrightarrow \{1, \ldots, N\} \). Then

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

\[
= 1 \cdot \frac{N - 1}{N} \cdot \frac{N - 2}{N} \cdot \ldots \cdot \frac{N - (q - 1)}{N}
\]

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

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 \( 1 \leq q \leq \sqrt{2N} \).
Union bound

\[ \Pr [C_1 \vee C_2] = \Pr [C_1] + \Pr [C_2] - \Pr [C_1 \wedge C_2] \]

\[ \leq \Pr [C_1] + \Pr [C_2] \]

More generally

\[ \Pr [C_1 \vee C_2 \vee \cdots \vee C_q] \leq \Pr [C_1] + \Pr [C_2] + \cdots + \Pr [C_q] \]
Arithmetic sums

\[ 0 + 1 + 2 + \cdots + (q - 1) = \]
Arithmetic sums

\[ 0 + 1 + 2 + \cdots + (q - 1) = \frac{q(q - 1)}{2} \]
Birthday bounds

Let

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

Then

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

Proof of this upper bound: Let \( C_i \) be the event that \( y_i \in \{y_1, \ldots, y_{i-1}\} \). Then

\[
C(N, q) = \Pr [C_1 \lor C_2, \ldots, \lor C_q] \\
\leq \Pr [C_1] + \Pr [C_2] + \ldots + \Pr [C_q] \\
\leq \frac{0}{N} + \frac{1}{N} + \ldots + \frac{q - 1}{N} \\
= \frac{q(q - 1)}{2N}.
\]
Let $E : \{0, 1\}^k \times \{0, 1\}^\ell \to \{0, 1\}^\ell$ be a block cipher.

Can we design $A$ so that

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

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

Let us turn this into an attack.
Birthday attack on a block cipher

\[ E : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell \text{ a block cipher} \]

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

**Game $\text{Real}_E$**

**procedure Initialize**

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

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

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.

<table>
<thead>
<tr>
<th>Game $\text{Real}_E$</th>
<th>adversary $A$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>procedure Initialize</strong></td>
<td>Let $x_1, \ldots, x_q \in {0, 1}^\ell$ be distinct</td>
</tr>
<tr>
<td>$K \leftarrow {0, 1}^k$</td>
<td>for $i = 1, \ldots, q$ do $y_i \leftarrow \text{Fn}(x_i)$</td>
</tr>
<tr>
<td><strong>procedure $\text{Fn}(x)$</strong></td>
<td>if $y_1, \ldots, y_q$ are all distinct</td>
</tr>
<tr>
<td>Return $E_K(x)$</td>
<td>then return 1 else return 0</td>
</tr>
</tbody>
</table>

Then

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

because $y_1, \ldots, y_q$ will be distinct because $E_K$ is a permutation.
Ideal world analysis

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

<table>
<thead>
<tr>
<th>Game Rand_{0,1}^\ell</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>procedure</strong> Fn(x)</td>
</tr>
<tr>
<td>$T[x] \leftarrow {0, 1}^\ell$</td>
</tr>
<tr>
<td>Return $T[x]$</td>
</tr>
</tbody>
</table>

**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

$$\Pr \left[ \text{Rand}_{0,1}^\ell A \Rightarrow 1 \right] = \Pr [y_1, \ldots, y_q \text{ all distinct}]$$

$$= 1 - C(2^\ell, q)$$

because $y_1, \ldots, y_q$ are randomly chosen from $\{0, 1\}^\ell$. 
Birthday attack on a block cipher

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

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

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

= $C(2^\ell, q)$

\[
\geq 0.3 \cdot \frac{q(q-1)}{2^\ell}
\]

so

$q \approx 2^{\ell/2} \Rightarrow \text{Adv}_{E}^{\text{prf}}(A) \approx 1$. 
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, 3DES</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>
KR-security versus PRF-security

We have seen two possible metrics of security for a block cipher $E$

- **KR-security**: It should be hard to get $K$ from input-output examples of $E_K$
- **PRF-security**: It should be hard to distinguish the input-output behavior of $E_K$ from that of a random function.

**Question**: Is it possible for $E$ to be

- PRF-secure, but
- **NOT** KR-secure?
Question: Is it possible for a block cipher $E$ to be PRF-secure but not KR-secure?

Why do we care? Because we
- agreed that KR-security is necessary
- claim that PRF-security is sufficient

for secure use of $E$, so a YES answer would render our claim false.

Luckily the answer to the above question is NO.
Fact: PRF-security implies

- KR-security
- Many other security attributes
Key recovery security, formally

Let $F : \text{Keys}(F) \times \text{Domain}(F) \rightarrow \text{Range}(F)$ a family of functions

Let $B$ be an adversary

<table>
<thead>
<tr>
<th>Game $\text{KR}_F$</th>
<th>procedure $\text{Fn}(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{procedure Initialize}$</td>
<td>return $F_K(x)$</td>
</tr>
<tr>
<td>$K \leftarrow $ \text{Keys}(F)$</td>
<td>$\text{procedure Finalize}(K')$</td>
</tr>
<tr>
<td></td>
<td>return $(K = K')$</td>
</tr>
</tbody>
</table>

The $kr$-advantage of $B$ is defined as

$$\text{Adv}_{kr}^F(B) = \Pr \left[ \text{KR}_F^B \Rightarrow \text{true} \right]$$

The oracle allows a chosen message attack.

$F$ is secure against key recovery if $\text{Adv}_{kr}^F(B)$ is “small” for all $B$ of “practical” resources.
Let \( k = L\ell \) and define \( F = \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L \) by

\[
F_K(X) = \begin{bmatrix}
K[1, 1] & K[1, 2] & \cdots & K[1, \ell] \\
K[2, 1] & K[2, 2] & \cdots & K[2, \ell] \\
\vdots & \vdots & \ddots & \vdots \\
K[L, 1] & K[L, 2] & \cdots & K[L, \ell]
\end{bmatrix}
\begin{bmatrix}
X[1] \\
X[2] \\
\vdots \\
X[\ell]
\end{bmatrix} = \begin{bmatrix}
Y[1] \\
Y[2] \\
\vdots \\
Y[L]
\end{bmatrix}
\]

Here the bits in the matrix are the bits in the key, and arithmetic is modulo two.

**Question:** Is \( F \) secure against key-recovery?
Let \( k = L\ell \) and define \( F = \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L \) by

\[
F_K(X) = \begin{bmatrix}
K[1, 1] & K[1, 2] & \cdots & K[1, \ell] \\
K[2, 1] & K[2, 2] & \cdots & K[2, \ell] \\
\vdots & \vdots & \ddots & \vdots \\
K[L, 1] & K[L, 2] & \cdots & K[L, \ell]
\end{bmatrix}
\begin{bmatrix}
X[1] \\
X[2] \\
\vdots \\
X[\ell]
\end{bmatrix} =
\begin{bmatrix}
Y[1] \\
Y[2] \\
\vdots \\
Y[L]
\end{bmatrix}
\]

Here the bits in the matrix are the bits in the key, and arithmetic is modulo two.

**Question:** Is \( F \) secure against key-recovery?

**Answer:** NO
Example

For $1 \leq i \leq \ell$ let:

$$e_j = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix}$$

be the $j$-th unit vector.

$$F_K(e_j) = \begin{bmatrix} K[1, 1] & K[1, 2] & \cdots & K[1, \ell] \\ K[2, 1] & K[2, 2] & \cdots & K[2, \ell] \\ \vdots \\ K[L, 1] & K[L, 2] & \cdots & K[L, \ell] \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} K[1, j] \\ K[2, j] \\ \vdots \\ K[L, j] \end{bmatrix}$$
Adversary $B$

\[
\begin{align*}
K' &\leftarrow \varepsilon \quad // \varepsilon \text{ is the empty string} \\
&\text{for } j = 1, \ldots, \ell \text{ do } y_j \leftarrow F_n(e_j); \ K' \leftarrow K' \parallel y_j \\
&\text{return } K'
\end{align*}
\]

Then

\[
\text{Adv}^{kr}_F(B) = 1.
\]

The time-complexity of $B$ is $t = O(\ell^2 L)$ since it makes $q = \ell$ calls to its oracle and each computation of $F_n = F_K$ takes $O(\ell L)$ time.

So $F$ is insecure against key-recovery.
Why does PRF-security imply KR-security?

Claim: KR-insecurity ⇒ PRF-insecurity

Real world

A

\[ x \]

\[ y \]

Ideal (Random) world

A

\[ x \]

\[ y \]

\[ y \leftarrow F_K(x) \]

\[ y \leftarrow \text{Range}(F) \]

If you give me a method \( B \) to defeat KR-security I can design a method \( A \) to defeat PRF-security.

What \( A \) does:

- Use \( B \) to find key \( K' \)
- Test whether \( F_n(x) = F_{K'}(x) \) for some new point \( x \)
- If this is true, decide it is in the Real world
Issues: To run $B$, adversary $A$ must give it input-output examples under $F_K$.

We have $A$ give $B$ input-output examples under $F_n$. This is correct in the real world but not in the random world. Nonetheless we can show it works.
If $F$ is a PRF then it is KR-secure

Our first example of a proof by reduction!

Given: $F : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$
Given: efficient KR-adversary $B$
Construct: efficient PRF-adversary $A$ such that:

$$\text{Adv}_{F}^{kr}(B) \leq \text{Adv}_{F}^{prf}(A) + \square$$

How to infer that PRF-secure $\Rightarrow$ KR-secure:

$F$ is PRF secure $\Rightarrow$ $\text{Adv}_{F}^{prf}(A)$ is small
$\Rightarrow$ $\text{Adv}_{F}^{kr}(B)$ is small
$\Rightarrow$ $F$ is KR-secure
Our first example of a proof by reduction!

Given: $F : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$

Given: efficient KR-adversary $B$

Construct: efficient PRF-adversary $A$ such that:

$$\text{Adv}_{F}^{\text{kr}}(B) \leq \text{Adv}_{F}^{\text{prf}}(A) + \square$$

Contrapositive:

$F$ not KR-secure $\Rightarrow \text{Adv}_{F}^{\text{kr}}(B)$ is big
$\Rightarrow \text{Adv}_{F}^{\text{prf}}(A)$ is big
$\Rightarrow F$ is not PRF-secure
How reductions work

A will run B as a subroutine

B’s world:

How A runs B

A itself answers B’s oracle queries, giving B the impression that B is in its own correct world.
If $F$ is a PRF then it is KR-secure

Given: $F : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$

Given: efficient KR-adversary $B$

Construct: efficient PRF-adversary $A$ such that:

$$\text{Adv}_{F}^{\text{kr}}(B) \leq \text{Adv}_{F}^{\text{prf}}(A) + \square$$

Idea:
- $A$ uses $B$ to find key $K'$
- Tests whether $K'$ is the right key

Issues:
- $B$ needs an $F_K$ oracle, which $A$ only has in the real world
- How to test $K'$?

How they are addressed:
- $A$ gives $B$ its $F_n$ oracle
- Test by seeing whether $F_{K'}$ agrees with $F_n$ on a new point.
If $F$ is a PRF then it is KR-secure

Given: $F : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$
Given: efficient KR-adversary $B$
Construct: efficient PRF-adversary $A$ such that:

$$\text{Adv}_{F}^{\text{kr}}(B) \leq \text{Adv}_{F}^{\text{prf}}(A) + \square$$

**adversary $A$**

1. $i \leftarrow 0$
2. $K' \leftarrow B^{\text{FnKRSim}}$
3. $x \leftarrow \{0, 1\}^\ell - \{x_1, \ldots, x_i\}$
4. if $F_{K'}(x) = \text{Fn}(x)$ then return 1
5. else return 0

**subroutine $\text{FnKRSim}(x)$**

1. $i \leftarrow i + 1$
2. $x_i \leftarrow x$
3. $y_i \leftarrow \text{Fn}(x)$
4. return $y_i$
adversary $A$

\[
i \leftarrow 0
\]
\[
K' \leftarrow B^\text{FnKRSim}
\]
\[
x \leftarrow \{0, 1\}^\ell - \{x_1, \ldots, x_i\}
\]
if $F_{K'}(x) = \text{Fn}(x)$ then return 1
else return 0

\begin{align*}
\text{subroutine } & \text{FnKRSim}(x) \\
i & \leftarrow i + 1 \\
x_i & \leftarrow x \\
y_i & \leftarrow \text{Fn}(x) \\
\text{return } y_i
\end{align*}

- If $\text{Fn} = F_K$ then $K' = K$ with probability the KR-advantage of $B$, so

\[
\Pr \left[ \text{Real}_F^A \Rightarrow 1 \right] \geq \text{Adv}^k_{F} (B)
\]

- If $\text{Fn}$ is a random function, then due to the fact that $x \notin \{x_1, \ldots, x_i\}$,

\[
\Pr \left[ \text{Rand}_{\text{Range}(F)}^A \Rightarrow 1 \right] = 2^{-L}
\]

So $\text{Adv}^\text{prf}_F (A) \geq \text{Adv}^k_F (B) - 2^{-L}$
If $F$ is PRF-secure then it is KR-secure

**Proposition:** Let $F : \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$ be a family of functions, and $B$ a kr-adversary making $q$ oracle queries. Then there is a PRF adversary $A$ making $q + 1$ oracle queries such that:

$$\text{Adv}^{\text{kr}}_F (B) \leq \text{Adv}^{\text{prf}}_F (A) + 2^{-L}$$

The running time of $A$ is that of $B$ plus $O(q(\ell + L))$ plus the time for one computation of $F$.

**Implication:**

$F$ PRF-secure $\Rightarrow$ $F$ is KR-secure.
Our Assumptions

DES, AES are good block ciphers in the sense of being PRF-secure to the maximum extent possible.