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.

Turing Intelligence Test

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!
-
-

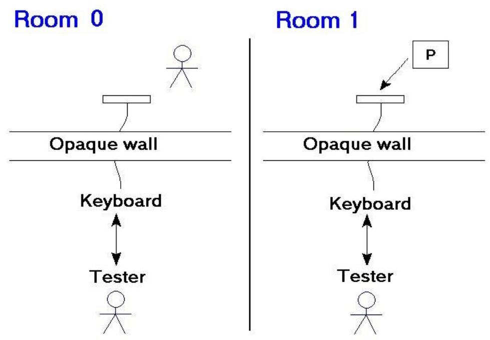
Clearly, no such list is a satisfactory answer to the question.

Turing Intelligence Test

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.

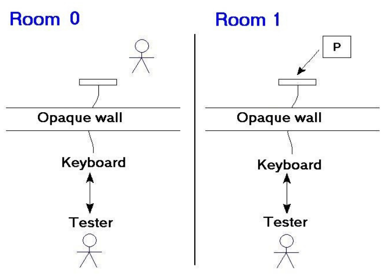
Turing Intelligence Test



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 rooom 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.

Real versus Ideal

Notion	Real object	Ideal object
Intelligence	Program	Human
PRF	Block cipher	?

Real versus Ideal

Notion	Real object	Ideal object
Intelligence	Program	Human
PRF	Block cipher	Random function

Random functions

Game Rand_R // here R is a set

procedure Fn(x)
if $\mathsf{T}[x] = \bot$ then $\mathsf{T}[x] \xleftarrow{\$} R$
return $\mathsf{T}[x]$

Adversary A

- Make queries to **Fn**
- Eventually halts with some output

We denote by

$$\Pr\left[\mathrm{Rand}_R^A \Rightarrow d\right]$$

the probability that A outputs d

Random functions

Game $\mathrm{Rand}_{\{0,1\}^3}$
procedure Fn(x)
if $\mathsf{T}[x] = \bot$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y \leftarrow$ **Fn**(01)
return $(y = 000)$

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^3}^A \Rightarrow \mathsf{true}\right] =$$

Random functions

Game $\mathrm{Rand}_{\{0,1\}^3}$
procedure Fn(x)
if $\mathsf{T}[x] = \bot$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y \leftarrow$ **Fn**(01)
return $(y = 000)$

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^3}^A \Rightarrow \mathsf{true}\right] = 2^{-3}$$

Random function

Game $\mathrm{Rand}_{\{0,1\}^3}$
procedure Fn(x)
if $\mathsf{T}[x] = \bot$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y_1 \leftarrow$ **Fn**(00)
$y_2 \leftarrow$ **Fn**(11)
return $(y_1 = 010 \wedge y_2 = 011)$

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^3}^A \Rightarrow \mathsf{true}\right] =$$

Random function

Game $\text{Rand}_{\{0,1\}^3}$
procedure Fn(x)
if $\mathsf{T}[x] = \perp$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y_1 \leftarrow \mathbf{Fn}(00)$
$y_2 \leftarrow \mathbf{Fn}(11)$
return $(y_1 = 010 \wedge y_2 = 011)$

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

Random function

Game $\text{Rand}_{\{0,1\}^3}$
procedure Fn(x)
if $\mathsf{T}[x] = \perp$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y_1 \leftarrow \mathbf{Fn}(00)$
$y_2 \leftarrow \mathbf{Fn}(11)$
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 Fn(x)
if $\mathsf{T}[x] = \perp$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^3$
return $\mathsf{T}[x]$

adversary A
$y_1 \leftarrow \mathbf{Fn}(00)$
$y_2 \leftarrow \mathbf{Fn}(11)$
return $(y_1 \oplus y_2 = 101)$

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

Recall: Function families

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

Examples:

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

Real versus Ideal

Notion	Real object	Ideal object
PRF	Family of functions (eg. a block cipher)	Random function

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: Keys $\times$ D $\to$ R be a family of functions.

Game Real_F

procedure Initialize
$K \xleftarrow{\$}$ Keys

procedure Fn(x)
Return $F_K(x)$

Game Rand_R

procedure Fn(x)
if $\mathrm{T}[x] = \perp$ then $\mathrm{T}[x] \xleftarrow{\$} \mathrm{R}$
Return $\mathrm{T}[x]$

Associated to F, A are the probabilities

$$\Pr\left[\mathrm{Real}_F^A \Rightarrow 1\right] \quad\Big|\quad \Pr\left[\mathrm{Rand}_R^A \Rightarrow 1\right]$$

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

$$\mathbf{Adv}_F^{\mathrm{prf}}(A) = \Pr\left[\mathrm{Real}_F^A \Rightarrow 1\right] - \Pr\left[\mathrm{Rand}_R^A \Rightarrow 1\right]$$

PRF advantage

A's output d	Intended meaning: I think I am in game
1	Real
0	Random

$\mathbf{Adv}_F^{\mathrm{prf}}(A) \approx 1$ means A is doing well and F is not prf-secure.
$\mathbf{Adv}_F^{\mathrm{prf}}(A) \approx 0$ (or ≤ 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 $\mathbf{Adv}_F^{\mathrm{prf}}(A)$ 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

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

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

Game Real_F
procedure Initialize
$K \xleftarrow{\$} \{0,1\}^\ell$
procedure Fn(x)
Return $K \oplus x$

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

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

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

is close to 1?

Example

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

Game Real_F
procedure Initialize
$K \xleftarrow{\$} \{0,1\}^\ell$
procedure Fn(x)
Return $K \oplus x$

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

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

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

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

adversary A
if $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

Example: Real 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 $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

Game Real_F
procedure Initialize
$K \xleftarrow{\$} \{0,1\}^\ell$
procedure Fn(x)
Return $K \oplus x$

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

Example: Real 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 $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

Game Real_F
procedure Initialize
$K \xleftarrow{\$} \{0,1\}^\ell$
procedure Fn(x)
Return $K \oplus x$

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

because

$$\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = F_K(0^\ell) \oplus F_K(1^\ell) = (K \oplus 0^\ell) \oplus (K \oplus 1^\ell) = 1^\ell$$

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 $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

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

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1\right] =$$

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 $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

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

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1\right] = \Pr\left[\mathbf{Fn}(1^\ell) \oplus \mathbf{Fn}(0^\ell) = 1^\ell\right] =$$

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 $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

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

$$\Pr\left[\mathrm{Rand}_{\{0,1\}^\ell}^A \Rightarrow 1\right] = \Pr\left[\mathbf{Fn}(1^\ell) \oplus \mathbf{Fn}(0^\ell) = 1^\ell\right] = 2^{-\ell}$$

because $\mathbf{Fn}(0^\ell), \mathbf{Fn}(1^\ell)$ are random ℓ-bit strings.

Example: Conclusion

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

adversary A
if $\mathbf{Fn}(0^\ell) \oplus \mathbf{Fn}(1^\ell) = 1^\ell$ then return 1 else return 0

Then

$$\begin{aligned}\mathbf{Adv}^{\mathrm{prf}}_F(A) &= \overbrace{\Pr\left[\mathrm{Real}^A_F \Rightarrow 1\right]}^{1} - \overbrace{\Pr\left[\mathrm{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1\right]}^{2^{-\ell}} \\ &= 1 - 2^{-\ell}\end{aligned}$$

and A is efficient .

Conclusion: F is not a secure PRF.

Exercise

Define the family of functions $F\colon \{0,1\}^{128} \times \{0,1\}^{128} \to \{0,1\}^{128}$ by $F(K, M) = \mathrm{AES}(M, K)$. Show that F is not a secure PRF by presenting in pseudocode an adversary A such that

- $\mathbf{Adv}^{\mathrm{prf}}_F(A) = 1 - 2^{-128}$
- A makes at most 2 queries to its **Fn** oracle
- A is very efficient.

You must *prove* that your A has the above properties.

Exercise

Let $G\colon \{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a family of functions (it is arbitrary but given, meaning known to the adversary) and let $r \geq 1$ be an integer. The *r-round Feistel cipher associated to G* is the family of functions $G^{(r)}\colon \{0,1\}^k \times \{0,1\}^{2l} \to \{0,1\}^{2l}$, defined as follows for any key $K \in \{0,1\}^k$ and input $x \in \{0,1\}^{2l}$:

<u>Function $G^{(r)}(K, x)$</u>
    $L_0 \| R_0 \leftarrow x$
    For $i = 1, \ldots, r$ do
        $L_i \leftarrow R_{i-1}$; $R_i \leftarrow G(K, R_{i-1}) \oplus L_{i-1}$
Return $L_r \| R_r$

By $a\|b$ we are denoting the concatenation of strings a, b. (For example $01\|10 = 0110$.) In the first line, we are parsing x as $x = L_0\|R_0$ with $|L_0| = |R_0| = l$, meaning L_0 is the first l bits of x and R_0 is the rest.

Exercise

1. Show that $G^{(1)}$ is not a secure PRF by presenting in pseudocode a practical adversary A such that $\mathbf{Adv}^{\mathrm{prf}}_{G^{(1)}}(A) = 1 - 2^{-l}$ and A makes one **Fn** query.
2. Show that $G^{(2)}$ is not a secure PRF by presenting in pseudocode a practical adversary A such that $\mathbf{Adv}^{\mathrm{prf}}_{G^{(2)}}(A) = 1 - 2^{-l}$ and A makes two **Fn** queries.

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

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

- 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

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

- 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

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

- 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

q	$C(365, q)$
15	0.253
18	0.347
20	0.411
21	0.444
23	0.507
25	0.569
27	0.627
30	0.706
35	0.814
40	0.891
50	0.970

Birthday Problem

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

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

Birthday setting: $N = 365$

Birthday Problem

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

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

Birthday setting: $N = 365$

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

Birthday collisions formula

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

$$\begin{aligned} 1 - C(N,q) &= \Pr[y_1, \ldots, y_q \text{ all distinct}] \\ &= 1 \cdot \frac{N-1}{N} \cdot \frac{N-2}{N} \cdot \cdots \cdot \frac{N-(q-1)}{N} \\ &= \prod_{i=1}^{q-1} \left(1 - \frac{i}{N}\right) \end{aligned}$$

so

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

Birthday bounds

Let

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

Fact: Then

$$0.3 \cdot \frac{q(q-1)}{N} \le C(N,q) \le 0.5 \cdot \frac{q(q-1)}{N}$$

where the lower bound holds for $1 \le q \le \sqrt{2N}$.

Block ciphers as PRFs

Let $E\colon \{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}^\ell$ be a block cipher.

Game Real_E	Game $\mathrm{Rand}_{\{0,1\}^\ell}$
procedure Initialize $K \xleftarrow{\$} \{0,1\}^k$ **procedure Fn**(x) Return $E_K(x)$	**procedure Fn**(x) if $\mathsf{T}[x] = \perp$ then $\mathsf{T}[x] \xleftarrow{\$} \{0,1\}^\ell$ Return $\mathsf{T}[x]$

Can we design A so that

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

is close to 1?

Block ciphers as PRFs

Defining property of a block cipher: E_K is a permutation for every K

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

- **Fn** $= E_K \Rightarrow$ **Fn**$(x_1), \ldots,$ **Fn**(x_q) distinct
- **Fn** random $\Rightarrow$ **Fn**$(x_1), \ldots,$ **Fn**(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 $y_i \leftarrow$ **Fn**(x_i)
if $y_1, \ldots, y_q$ are all distinct then return 1
else return 0

Real world analysis

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

Game Real_E	**adversary** A
procedure Initialize $K \xleftarrow{\$} \{0,1\}^k$ **procedure Fn**(x) Return $E_K(x)$	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

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

Real world analysis

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

Game Real_E	**adversary** A
procedure Initialize $K \xleftarrow{\$} \{0,1\}^k$ **procedure Fn**(x) Return $E_K(x)$	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

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

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

Rand world analysis

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

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

procedure Fn(x)

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

Return $T[x]$

adversary A

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

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

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

then return 1 else return 0

Then

$$\Pr\left[\mathrm{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1\right] = \Pr[y_1, \ldots, y_q \text{ all distinct}] = 1 - C(2^\ell, q)$$

because $y_1, \ldots, y_q$ are randomly chosen from $\{0,1\}^\ell$.

Birthday attack on a block cipher

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

adversary A

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

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

if $y_1, \ldots, y_q$ are all distinct then return 1 else return 0

$$\begin{aligned}\mathbf{Adv}^{\mathrm{prf}}_E(A) &= \overbrace{\Pr\left[\mathrm{Real}^A_E \Rightarrow 1\right]}^{1} - \overbrace{\Pr\left[\mathrm{Rand}^A_{\{0,1\}^\ell} \Rightarrow 1\right]}^{1-C(2^\ell,q)} \\ &= C(2^\ell, q) \geq 0.3 \cdot \frac{q(q-1)}{2^\ell}\end{aligned}$$

so

$$q \approx 2^{\ell/2} \Rightarrow \mathbf{Adv}^{\mathrm{prf}}_E(A) \approx 1 .$$

Birthday attack on a block cipher

Conclusion: If $E: \{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}^\ell$ 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!

	ℓ	$2^{\ell/2}$	Status
DES, 2DES, 3DES3	64	2^{32}	Insecure
AES	128	2^{64}	Secure

KR-security versus PRF-security

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!

Exercise

We are given a PRF $F\colon \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^k$ and want to build a PRF $G\colon \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^{2k}$. Which of the following work?

1. $\underline{\text{Function } G(K,x)}$
   $y_1 \leftarrow F(K,x)$; $y_2 \leftarrow F(K,\overline{x})$; Return $y_1 \| y_2$
2. $\underline{\text{Function } G(K,x)}$
   $y_1 \leftarrow F(K,x)$; $y_2 \leftarrow F(K,y_1)$; Return $y_1 \| y_2$
3. $\underline{\text{Function } G(K,x)}$
   $L \leftarrow F(K,x)$; $y_1 \leftarrow F(L,0^k)$; $y_2 \leftarrow F(L,1^k)$; Return $y_1 \| y_2$
4. $\underline{\text{Function } G(K,x)}$
   [Your favorite code here]