CSE107: Intro to Modern Cryptography

https://cseweb.ucsd.edu/classes/sp22/cse107-a/

Emmanuel Thomé

April 7, 2022

Lecture 4

Block ciphers and Pseudo-random functions

Plan

Attacks on DES

Beyond DES

AES

Further security metrics

PRF security and the birthday bound

Exhaustive Key Search attack

Let $E\colon \text{Keys} \times \text{D} \to \text{R}$ be a function family with $\text{Keys} = \{\, T_1, \ldots, T_N \}$ and $\text{D} = \{x_1, \ldots, x_d\}$. Let $1 \le q \le d$ be a parameter.

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \mathbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} \; : \; E(T_i, M_j) = C_j)$ then return T_i

Question: What is $\mathbf{Adv}^{\text{kr}}_{E}(A_{\text{eks}})$?

Exhaustive Key Search attack

Let $E\colon \text{Keys} \times \text{D} \to \text{R}$ be a function family with $\text{Keys} = \{T_1, \ldots, T_N\}$ and $\text{D} = \{x_1, \ldots, x_d\}$. Let $1 \leq q \leq d$ be a parameter.

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \mathbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} \;:\; E(T_i, M_j) = C_j)$ then return T_i

Question: What is $\mathbf{Adv}^{\text{kr}}_{E}(A_{\text{eks}})$?

Answer: It equals 1.

Because

- There is some i such that $T_i = K$, and
- K is consistent with $(M_1, C_1), \ldots, (M_q, C_q)$.

Exhaustive Key Search attack

Let $E\colon \text{Keys} \times \text{D} \to \text{R}$ be a function family with $\text{Keys} = \{T_1, \ldots, T_N\}$ and $\text{D} = \{x_1, \ldots, x_d\}$. Let $1 \le q \le d$ be a parameter.

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \mathbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} : E(T_i, M_j) = C_j)$ then return T_i

Question: What is $\mathbf{Adv}_E^{\text{tkr}}(A_{\text{eks}})$?

Exhaustive Key Search attack

Let $E\colon \text{Keys} \times \text{D} \to \text{R}$ be a function family with $\text{Keys} = \{T_1, \ldots, T_N\}$ and $\text{D} = \{x_1, \ldots, x_d\}$. Let $1 \le q \le d$ be a parameter.

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \textbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} \,:\, E(T_i, M_j) = C_j)$ then return T_i

Question: What is $\textbf{Adv}^{\text{tkr}}_E(A_{\text{eks}})$?

Answer: Hard to say! Say $K = T_m$ but there is a $i < m$ such that $E(T_i, M_j) = C_j$ for $1 \le j \le q$. Then T_i, rather than K, is returned.

In practice if $E\colon \{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}^\ell$ is a "real" block cipher and $q > k/\ell$, we expect that $\textbf{Adv}^{\text{tkr}}_E(A_{\text{eks}})$ is close to 1 because K is likely the only key consistent with the input-output examples.

How long does exhaustive key search take?

DES can be computed at well over 10 Gbits/sec in hardware.

DES plaintext = 64 bits

Chip can perform $10^{10}/64 = 1.625 \times 10^8$ DES computations per second

Expect $A_{\rm eks}$ $(q = 1)$ to succeed in 2^{55} DES computations, so it takes time

$$\frac{2^{55}}{1.625 \times 10^8} \approx 2.2 \times 10^8 \text{ seconds} \approx 7 \text{ years!}$$

Small optimization with "Key Complementation" ⇒ 3.5 years

This is (somewhat) prohibitive. Does this mean DES is secure?

Differential and linear cryptanalysis

Exhaustive key search is a generic attack: Did not attempt to "look inside" DES and find/exploit weaknesses.

The following non-generic key-recovery attacks on DES have advantage close to one and running time smaller than 2^{56} DES computations:

Attack	when	q, running time
Differential cryptanalysis	1992	2^{47}
Linear cryptanalysis	1993	2^{44}

Differential and linear cryptanalysis

Exhaustive key search is a generic attack: Did not attempt to "look inside" DES and find/exploit weaknesses.

The following non-generic key-recovery attacks on DES have advantage close to one and running time smaller than 2^{56} DES computations:

Attack	when	q, running time
Differential cryptanalysis	1992	2^{47}
Linear cryptanalysis	1993	2^{44}

But merely storing 2^{44} input-output pairs requires 281 Terabytes.

In practice these attacks were prohibitively expensive.

Note: DES withstood differential cryptanalysis attacks quite well. This was the explanation for the S-boxes tables in its design.

EKS revisited

adversary A_{eks}

For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \textbf{Fn}(M_j)$

For $i = 1, \ldots, N$ do

if $(\forall j \in \{1, \ldots, q\} \; : \; E(T_i, M_j) = C_j)$ then return T_i

EKS revisited

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \mathbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} : E(T_i, M_j) = C_j)$ then return T_i

Observation: The E computations can be performed in parallel!

EKS revisited

adversary A_{eks}
For $j = 1, \ldots, q$ do $M_j \leftarrow x_j$; $C_j \leftarrow \mathbf{Fn}(M_j)$
For $i = 1, \ldots, N$ do
if $(\forall j \in \{1, \ldots, q\} \; : \; E(T_i, M_j) = C_j)$ then return T_i

Observation: The E computations can be performed in parallel!

In 1993, Wiener designed a dedicated DES-cracking machine:

- \$1 million
- 57 chips, each with many, many DES processors
- Finds key in 3.5 hours

RSA DES challenges

$K \xleftarrow{\$} \{0,1\}^{56}$; $Y \leftarrow \mathrm{DES}(K, X)$; Publish Y on website.

Reward for recovering X

Challenge	Post Date	Reward	Result
I	1997	\$10,000	Distributed.Net: 4 months
II	1998	Depends how fast you find key	Distributed.Net: 41 days. EFF: 56 hours
III	1998	As above	$<$ 28 hours

DES security summary

DES is considered broken because its short key size permits rapid key search.

But DES is a very strong design as evidenced by the fact that there are no practical attacks that exploit its structure.

Plan

Attacks on DES

Beyond DES

AES

Further security metrics

PRF security and the birthday bound

2DES

Block cipher $2DES : \{0,1\}^{112} \times \{0,1\}^{64} \to \{0,1\}^{64}$ is defined by

$$2DES_{K_1 K_2}(M) = DES_{K_2}(DES_{K_1}(M))$$

- Exhaustive key search takes 2^{112} DES computations, which is too much even for machines
- Resistant to differential and linear cryptanalysis.

Meet-in-the-middle attack on 2DES

Suppose K_1K_2 is a target 2DES key and adversary has M, C such that

$$C = 2DES_{K_1K_2}(M) = DES_{K_2}(DES_{K_1}(M))$$

Then

$$DES^{-1}_{K_2}(C) = DES_{K_1}(M)$$

Meet-in-the-middle attack on 2DES

Suppose $DES^{-1}_{K_2}(C) = DES_{K_1}(M)$ and $T_1, \ldots, T_N$ are all possible DES keys, where $N = 2^{56}$.

T_1	$DES(T_1, M)$
T_i	$DES(T_i, M)$
T_N	$DES(T_N, M)$

Table L

$DES^{-1}(T_1, C)$	T_1
$DES^{-1}(T_j, C)$	T_j
$DES^{-1}(T_N, C)$	T_N

Table R

Attack idea:

- Build L,R tables

Meet-in-the-middle attack on 2DES

Suppose $DES^{-1}_{K_2}(C) = DES_{K_1}(M)$ and $T_1, \ldots, T_N$ are all possible DES keys, where $N = 2^{56}$.

$K_1 \rightarrow$ (row T_i of Table L)

T_1	$DES(T_1, M)$
T_i	$DES(T_i, M)$
T_N	$DES(T_N, M)$

Table L

$\overset{\text{equal}}{\longleftrightarrow}$

$DES^{-1}(T_1, C)$	T_1
$DES^{-1}(T_j, C)$	T_j
$DES^{-1}(T_N, C)$	T_N

$\leftarrow K_2$

Table R

Attack idea:

- Build L,R tables
- Find i, j s.t. $L[i] = R[j]$
- Guess that $K_1 K_2 = T_i T_j$

Meet-in-the-middle attack on 2DES

Let $T_1, \ldots, T_{2^{56}}$ denote an enumeration of DES keys.

adversary A_{MinM}
$M_1 \leftarrow 0^{64}$; $C_1 \leftarrow \mathbf{Fn}(M_1)$
for $i = 1, \ldots, 2^{56}$ do $L[i] \leftarrow \mathsf{DES}(T_i, M_1)$
for $j = 1, \ldots, 2^{56}$ do $R[j] \leftarrow \mathsf{DES}^{-1}(T_j, C_1)$
$S \leftarrow \{ (i, j) \,:\, L[i] = R[j] \}$
Pick some $(l, r) \in S$ and return $T_l \,\|\, T_r$

This uses $q = 1$ plaintext-ciphertext pair and is unlikely to return the target key. For that one should extend the attack to a larger value of q.

Running time of Meet-in-the-middle attack

adversary A_{MinM}
$M_1 \leftarrow 0^{64}$; $C_1 \leftarrow \textbf{Fn}(M_1)$
for $i = 1, \ldots, 2^{56}$ do $L[i] \leftarrow \mathsf{DES}(T_i, M_1)$
for $j = 1, \ldots, 2^{56}$ do $R[j] \leftarrow \mathsf{DES}^{-1}(T_j, C_1)$
$S \leftarrow \{ \, (i, j) \; : \; L[i] = R[j] \, \}$
Pick some $(l, r) \in S$ and return $T_l \parallel T_r$

Let T_{DES} be the time to compute DES or DES^{-1}.

Let $k = 56$ be the key length. Let $\ell = 64$ be the block length.

Each "for" loop takes $\mathcal{O}(2^k \cdot T_{\mathsf{DES}})$ time.

To create S, we can sort the tables and then compare entries. Recall that sorting a size N list takes $\mathcal{O}(N \log(N))$ comparisons. So the time for this step is $\mathcal{O}(k\ell \cdot 2^k)$. Why? $N = 2^k$, and comparison is $\mathcal{O}(\ell)$.

Running time of Meet-in-the-middle attack

adversary A_{MinM}
$M_1 \leftarrow 0^{64}$; $C_1 \leftarrow \mathbf{Fn}(M_1)$
for $i = 1, \ldots, 2^{56}$ do $L[i] \leftarrow \mathsf{DES}(T_i, M_1)$
for $j = 1, \ldots, 2^{56}$ do $R[j] \leftarrow \mathsf{DES}^{-1}(T_j, C_1)$
$S \leftarrow \{ (i, j) : L[i] = R[j] \}$
Pick some $(l, r) \in S$ and return $T_l \| T_r$

Let T_{DES} be the time to compute DES or DES^{-1}.

Let $k = 56$ be the key length. Let $\ell = 64$ be the block length.

Overall attack takes time $\mathcal{O}(2^k \cdot (T_{\mathsf{DES}} + k\ell))$.

In practice this should be around 2^{57} DES/DES^{-1} operations, which is about the same as the cost of exhaustive key search on DES itself.

BUT: this also costs 2^{56} memory, which is a significant problem.

3DES

Block ciphers

$$3DES3 : \{0,1\}^{168} \times \{0,1\}^{64} \rightarrow \{0,1\}^{64}$$

$$3DES2 : \{0,1\}^{112} \times \{0,1\}^{64} \rightarrow \{0,1\}^{64}$$

are defined by

$$3DES3_{K_1 \,\|\, K_2 \,\|\, K_3}(M) \;=\; DES_{K_3}(DES^{-1}_{K_2}(DES_{K_1}(M)))$$

$$3DES2_{K_1 \,\|\, K_2}(M) \;=\; DES_{K_2}(DES^{-1}_{K_1}(DES_{K_2}(M)))$$

Meet-in-the-middle attack on 3DES3 reduces its "effective" key length to 112.

Block size limitation

Later we will see "birthday" attacks that "break" a block cipher $E : \{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}^\ell$ in time $2^{\ell/2}$

For DES this is $2^{64/2} = 2^{32}$ which is small, and this is unchanged for 2DES and 3DES.

Would like a larger block size.

Plan

Attacks on DES

Beyond DES

AES

Further security metrics

PRF security and the birthday bound

AES

1998: NIST announces competition for a new block cipher

- key length 128 (+ requirement to have several other possible lengths)
- block length 128
- faster than DES in software

Submissions from all over the world: MARS, Rijndael, Two-Fish, RC6, Serpent, Loki97, Cast-256, Frog, DFC, Magenta, E2, Crypton, HPC, Safer+, Deal

AES

1998: NIST announces competition for a new block cipher

- key length 128 (+ requirement to have several other possible lengths)
- block length 128
- faster than DES in software

Submissions from all over the world: MARS, Rijndael, Two-Fish, RC6, Serpent, Loki97, Cast-256, Frog, DFC, Magenta, E2, Crypton, HPC, Safer+, Deal

2001: NIST selects Rijndael to be AES.

AES: versions

AES has several different versions.

- 128-bit key.
- 192-bit key.
- 256-bit key.

The *block length* is 128 bits in all cases. Only the key schedule and the number of rounds vary (10, 12, 14);.

AES

function $\mathsf{AES}_K(M)$

  $(K_0, \ldots, K_{10}) \leftarrow \mathit{expand}(K)$

  $s \leftarrow M \oplus K_0$

  for $r = 1$ to 10 do

    $s \leftarrow S(s)$

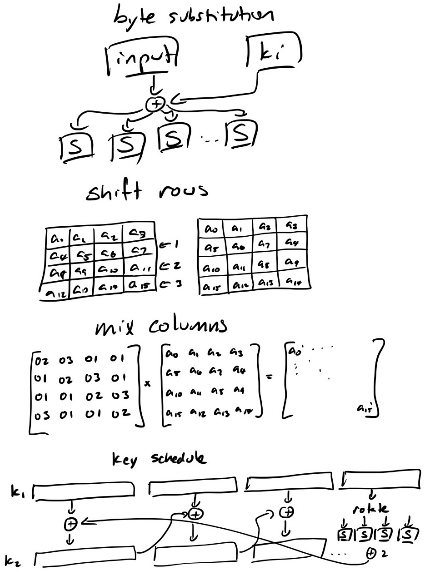
    $s \leftarrow \mathit{shift\text{-}rows}(s)$

    if $r \le 9$ then $s \leftarrow \mathit{mix\text{-}cols}(s)$ fi

    $s \leftarrow s \oplus K_r$

  end for

  return s



- Fewer tables than DES
- Finite field operations

The AES movie

http://www.youtube.com/watch?v=H2LlHOw_ANg

Implementing AES

	Code size	Performance
Pre-compute and store round function tables	largest	fastest
Pre-compute and store S-boxes only	smaller	slower
No pre-computation	smallest	slowest

AES-NI: Hardware for AES, now present on most processors.
Your laptop has it! Can run AES at around 1 cycle/byte. VERY fast!

Security of AES

Best known key-recovery attack [BoKhRe11] takes $2^{126.1}$ time, which is only marginally better than the 2^{128} time of EKS.

There are attacks on reduced-round versions of AES as well as on its sibling algorithms AES192, AES256. Many of these are "related-key" attacks.

After 20 years, AES has withstood a great number of attack attempts, which have barely made a dent. This shows the strength of the design.

Exercise: given 1 year $\approx 2^{25}$ seconds:
2^{128} operations with 2^{20} cores at 8 GHz (2^{33} Hz) $=$

Security of AES

Best known key-recovery attack [BoKhRe11] takes $2^{126.1}$ time, which is only marginally better than the 2^{128} time of EKS.

There are attacks on reduced-round versions of AES as well as on its sibling algorithms AES192, AES256. Many of these are "related-key" attacks.

After 20 years, AES has withstood a great number of attack attempts, which have barely made a dent. This shows the strength of the design.

Exercise: given 1 year $\approx 2^{25}$ seconds:
2^{128} operations with 2^{20} cores at 8 GHz (2^{33} Hz) $= 2^{50}$ years ($\approx$ a quadrillion years).

Plan

Attacks on DES

Beyond DES

AES

Further security metrics

PRF security and the birthday bound

Limitations of security against key recovery

So far, a block cipher has been viewed as secure if it resists key recovery, meaning there is no efficient adversary A having $\mathbf{Adv}_E^{\mathrm{kr}}(A) \approx 1$.

Is security against key recovery enough?

Not really. For example define $E\colon \{0,1\}^{128} \times \{0,1\}^{256} \to \{0,1\}^{256}$ by

$$E_K(M[1]M[2]) = M[1] \| \mathsf{AES}_K(M[2])$$

This is as secure against key-recovery as AES, but not a "good" blockcipher because half the message is in the clear in the ciphertext.

So what?

Possible reaction: But DES, AES are not designed like E above, so why does this matter?

Answer: It tells us that security against key recovery is not, as a block-cipher property, sufficient for security of uses of the block cipher.

As designers and users we want to know what properties of a block cipher give us security when the block cipher is used.

So what is a “good” block cipher?

Possible Properties	Necessary?	Sufficient?
security against key recovery	YES	NO!
hard to find M given $C = E_K(M)$	YES	NO!
$\vdots$		

We can’t define or understand security well via some such (indeterminable) list.

We want a single “master” property of a block cipher that is sufficient to ensure security of common usage of the block cipher.

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!
-
-

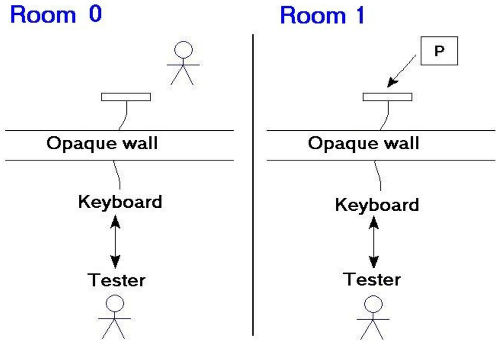
Clearly, no such list is a satisfactory answer to the question.

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.

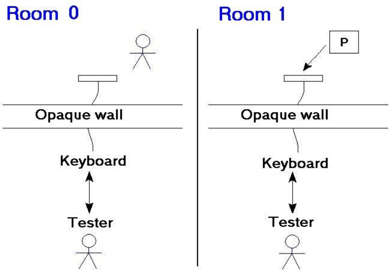
Turing Intelligence Test



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 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.

Plan

Further security metrics

- Real versus Ideal
- The prf advantage
- An Example

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 (R is a set)

procedure Initialize
$T[] \leftarrow (\perp \text{ for all } x)$

procedure Finalize(x)
Return x

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

Adversary A will play this game.

- A makes queries to **Fn**
- Eventually A halts with some true/false output.
- The game's outcome is exactly A's 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 Initialize
$T[] \leftarrow (\perp \text{ for all } x)$

procedure Finalize(x)
Return x

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

adversary A
$y \leftarrow \mathbf{Fn}(01)$ // just an arbitrary query
return $(y = 000)$

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

Random functions

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

procedure Initialize
$T[] \leftarrow (\perp$ for all $x)$

procedure Finalize(x)
Return x

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

adversary A
$y \leftarrow \mathbf{Fn}(01)$ // just an arbitrary query
return $(y = 000)$

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

Random functions

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

procedure Initialize
$T[] \leftarrow (\perp$ for all $x)$

procedure Finalize(x)
Return x

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

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

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

Random functions

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

procedure Initialize
$T[] \leftarrow (\perp \text{ for all } x)$

procedure Finalize(x)
Return x

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 $(y_1 = 010 \wedge y_2 = 011)$

$$\Pr\left[\mathrm{Rand}^{A}_{\{0,1\}^3} \Rightarrow \mathsf{true}\right] = 2^{-6}$$

Random functions

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

procedure Initialize
$T[] \leftarrow (\perp \text{ for all } x)$

procedure Finalize(x)
Return x

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 $(y_1 \oplus y_2 = 101)$

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

Random functions

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

procedure Initialize
$T[] \leftarrow (\perp$ for all $x)$

procedure Finalize(x)
Return x

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 $(y_1 \oplus y_2 = 101)$

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

Recall: Function families

Definition: family of functions

A family of functions (also called a function family) is a two-input function

$$F : \mathrm{Keys} \times \mathrm{D} \to \mathrm{R}.$$

Notation: For $K \in \mathrm{Keys}$ we let

$$F_K : \begin{cases} \mathrm{D} & \to & \mathrm{R} \\ x & \mapsto & F(K, x) \end{cases}$$

In other words, $F_K(x) = F(K, x)$ for any $x \in \mathrm{D}$.

Examples:

- DES: $\mathrm{Keys} = \{0,1\}^{56}$, $\mathrm{D} = \mathrm{R} = \{0,1\}^{64}$
- Any block cipher: $\mathrm{D} = \mathrm{R}$ 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.

F is not a PRF if a tester can tell the input-output behavior of F_k apart from the input-output behavior of a random function.
Note that the tester does not have to get the key K to succeed!

Plan

Further security metrics

- Real versus Ideal
- The prf advantage
- An Example

Games defining prf advantage of an adversary against F

Let F: Keys $\times$ D $\rightarrow$ R be a family of functions.

Game Real_F

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

procedure Finalize(x)
Return x

procedure Fn(x)
return $F_K(x)$

Game Rand_R

procedure Initialize
$T[] \leftarrow (\perp$ for all $x)$

procedure Finalize(x)
Return x

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

The adversary A will play both games, and must tell which is which.
A's output is a guess.

PRF advantage

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

Associated to F, A are the probabilities

$$\Pr\left[\mathrm{Real}_F^A \Rightarrow 1\right] \quad \text{and} \quad \Pr\left[\mathrm{Rand}_{\mathsf{R}}^A \Rightarrow 1\right]$$

that A outputs 1 in each world.

Definition of $\mathbf{Adv}^{\mathrm{prf}}$

The advantage of A is

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

$\mathbf{Adv}_F^{\mathrm{prf}}(A) \approx 1$ 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.

Plan

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

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

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

Game Real_F

procedure Initialize
$K \xleftarrow{\$} \{0,1\}^\ell$

procedure Fn(x)
Return $K \oplus x$

We have

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

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 $K \oplus x$

We have

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

because

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

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

We have

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

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

We have

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

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

We have

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

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

Example: Conclusion

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

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

and A is efficient (2 queries and very simple computation).

Conclusion: F is not a secure PRF.

Plan

Attacks on DES

Beyond DES

AES

Further security metrics

PRF security and the birthday bound

Plan

PRF security and the birthday bound

- Happy birthday
- Block ciphers as PRFs

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

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

- 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

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

- 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

- $C(365, q) \approx q^2/2/365$
- 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 $y_1, \ldots, y_q \xleftarrow{\$} \{1, \ldots, N\}$ and let

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

Birthday setting: $N = 365$

Birthday Problem

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

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

Birthday setting: $N = 365$

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

Birthday collisions formula

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

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

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 $1 \leq q \leq \sqrt{2N}$.

Plan

PRF security and the birthday bound

- Happy birthday
- Block ciphers as PRFs

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

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

procedure Fn(x)
return $E_K(x)$

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

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

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

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

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

Analysis

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

Game Real_E

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

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 $y_i \leftarrow \mathbf{Fn}(x_i)$
if $y_1, \ldots, y_q$ are all distinct
then return 1 else return 0

Then

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

Analysis

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

Game Real_E

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

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 $y_i \leftarrow \mathbf{Fn}(x_i)$
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.

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

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

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

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 $y_i \leftarrow \textbf{Fn}(x_i)$
if $y_1, \ldots, y_q$ are all distinct
then return 1 else return 0

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

so

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

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.

How are they related?

An example of a reduction

Proposition (which we won't prove)

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^\ell$ be a family of functions.
Given any kr-adversary B making q (distinct!) oracle queries, we can construct a PRF adversary A making q oracle queries such that

$$\mathbf{Adv}_E^{\mathrm{kr}}(B) \leq \mathbf{Adv}_E^{\mathrm{prf}}(A) + 2^{k-q\ell} .$$

The running time of A is that of B plus $O(q\ell)$.

Interpretation:

$$\begin{aligned} E \text{ is PRF secure } &\Rightarrow \mathbf{Adv}_E^{\mathrm{prf}}(A) \text{ is small for any } A \text{ (definition)} \\ &\Rightarrow \mathbf{Adv}_E^{\mathrm{kr}}(B) \text{ is small (by the proposition above)} \\ &\Rightarrow E \text{ is KR-secure.} \end{aligned}$$

Example: If $E = \mathrm{AES}$ and $q = 2$ then $2^{k-q\ell} = 2^{-128}$.

This is called a proof by reduction.

Consequence of the reduction

The proposition proves that PRF security of E implies

- KR (and hence TKR) security of E.
- PRF security also implies many other security attributes of E

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

Block Cipher Security 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. (And for DES, these inherent limitations imply that DES is obsolete.)

You can assume this in designs and analyses.

But beware that the future may prove these assumptions wrong!