CS 6260
Applied Cryptography

Alexandra (Sasha) Boldyreva

Block ciphers, pseudorandom functions and permutations

Block ciphers

Building blocks for symmetric encryption.

Examples: DES, 3DES, AES...

$M \to E_K \to C$

- A block cipher is a function family $E:\{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$, where k-key length, n-input and output lengths are the parameters
- Notation: for every $K \in \{0,1\}^k$ $E_K(M)=E(K,M)$
- For every $K \in \{0,1\}^k$, $E_K(\cdot)$ is a permutation (one-to-one and onto function). For every $C \in \{0,1\}^n$ there is a single $M \in \{0,1\}^n$ s.t. $C=E_K(M)$
- Thus each block cipher has an inverse for every key: $E_K^{-1}(\cdot)$ s.t. $E_K(E_K^{-1}(M))=C$ for all $M,C \in \{0,1\}^n$
- For every $K \in \{0,1\}^k$, $E_K(\cdot),E_K^{-1}(\cdot):\{0,1\}^n \to \{0,1\}^n$

DES

- Key length k=56, input and output length n=64
- 1973. NBS (National Bureau of Standards) announced a search for a data protection algorithm to be standardized
- 1974. IBM submits a design based on “Lucifer” algorithm
- 1975. The proposed DES is published
- 1976. DES approved as a federal standard
- DES is highly efficient: $\approx 2.5 \cdot 10^7$ DES computations per second

Security of block ciphers

- Any block cipher E is subject to exhaustive key-search: given (M1,C1=E(K,M1),…,(Mq,Cq=E(K,Mq)) an adversary can recover K (or another key consistent with the given pairs) as follows:

EKS_E((M1,C1),…(Mq,Cq))

For i=1,…,2^k do
  if E(Ti,M1)=C1 then //Ti is i-th k-bit string//
    if E(Ti,Mj)=Cj for all 2≤j≤q then return Ti EndIf
  EndIf
EndFor

Security of block ciphers

- Exhaustive key search takes 2^k block cipher computations in the worst case.
- On the average: $\sum_{i=1}^{2^k} i \cdot \Pr[K = T_i] = \sum_{i=1}^{2^k} \frac{i}{2^k} = \frac{1}{2^k} \cdot \sum_{i=1}^{2^k} i$

$$= \frac{1}{2^k} \cdot \frac{2^k(2^k+1)}{2} = \frac{2^k+1}{2} \approx 2^{k-1}$$

- DES has a property that $\mathrm{DES}_K(x) = \overline{\mathrm{DES}_{\overline{K}}(\overline{x})}$, this speeds up exhaustive search by a factor of 2
- For DES (k=56) exhaustive search takes $2^{55}/2{\cdot}2.5{\cdot}10^7$ that is about 23 years

Security of DES

- There are more sophisticated attacks known:
  - differential cryptoanalysis: finds the key given about 2^{47} chosen plaintexts and the corresponding ciphertexts
  - linear cryptoanalysis: finds the key given about 2^{42} known plaintext and ciphertext pairs
- These attacks require too many data, hence exhaustive key search is the best known attack. And it can be mounted in parallel!
- A machine for DES exhaustive key search was built for $250,000. It finds the key in about 56 hours on average.
- A new block cipher was needed....
- Triple-DES: 3DES(K1||K2,M)=DES(K2, DES^{-1}(K1, DES(K2,M)).
  - 3DES's keys are 112-bit long. Good, but needs 3 DES computations

Advanced Encryption Standard (AES)

- 1998. NIST announced a search for a new block cipher.
- 15 algorithms from different countries were submitted
- 2001. NIST announces the winner: an algorithm Rijndael, designed by Joan Daemen and Vincent Rijmen from Belgium.
- AES: block length n=128, key length k is variable: 128, 192 or 256 bits.
- Exhaustive key search is believed infeasible

Limitations of key-recovery based security

- A classical approach to block cipher security: key recovery should be infeasible.
- I.e. given (M1,E(K,M1),...,Mq,E(K,Mq)), where K is chosen at random and M1,...Mq are chosen at random (or by an adversary), the adversary cannot compute K in time t with probability ϵ.
- Necessary, but is it sufficient?
- Consider E'(K,M1||M2)=E(K,M1)||M2 for some "good" E. Key recovery is hard for E' as well, but it does not look secure.
- Q. What property of a block cipher as a building block would ensure various security properties of different constructions?

Intuition

- We want that (informally)
  - key search is hard
  - a ciphertext does not leak bits of the plaintext
  - a ciphertext does not leak any function of a plaintexts
  - ....
  - there is a "master" property of a block cipher as a building block that enables security analysis of protocols based on block ciphers
- It is good if ciphertexts "look" random

- Pseudorandom functions (PRFs) and permutations (PRPs) are very important tools in cryptography. Let's start with the notion of function families:
- A function family F is a map Keys(F)×Dom(F) → Range(F).
- For any K∈Keys(F) we define F_K=F(K,M), call it an **instance** of F.
- Notation $f \xleftarrow{\$} F$ is the shorthand for $K \xleftarrow{\$}$ Keys(F); $f \leftarrow F_K$
- Block cipher E is a function family with Dom(E)=Range(E)=$\{0,1\}^n$ and Keys(K)=$\{0,1\}^k$

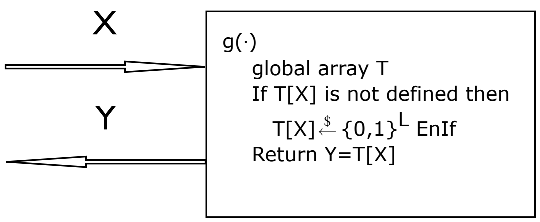
- Let Func(ℓ,L) denote the set of all functions from $\{0,1\}^\ell$ to $\{0,1\}^L$.
- It's a function family where a key specifying an instance is a description of this instance function.
- Q. How large is the key space?
- A.
- We will often consider the case when ℓ=L
- Let's try to understand how a random function (a random instance f of Func(ℓ,L)) behaves

Random functions

- $g \xleftarrow{\$} F(\ell,L)$
- We are interested in the input-output behavior of a random function. Let's imagine that we have access to a subroutine that implements such a function:

```
g(X∈{0,1}^ℓ)
   global array T
   If T[X] is not defined then
      T[X] ←$ {0,1}^L EndIf
   Return T[X]
```

“Black box” access



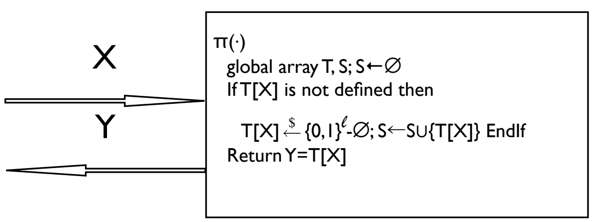
Note that for any X∈$\{0,1\}^\ell$ and Y∈$\{0,1\}^L$ Pr[g(X)=Y]=2^{-L}

Random permutations

- Perm(ℓ) is the set of all permutations on $\{0,1\}^\ell$
- Q. How large is the key space?
- A. $\ell!$
- We are interested in a random instance $\pi \xleftarrow{\$}$ Perm(ℓ)

π(X∈$\{0,1\}^\ell$)
  global array T, S; S←∅
  If T[X] is not defined then
    T[X] $\xleftarrow{\$}$ $\{0,1\}^\ell$-∅; S←S∪{T[X]} EndIf
  Return T[X]

“Black box” access



For any X∈$\{0,1\}^\ell$ and Y∈$\{0,1\}^\ell$ Pr[π(X)=Y]=$2^{-\ell}$

Random functions vs permutations

Fix $X_1, X_2 \in \{0,1\}^\ell$ and $Y_1, Y_2 \in \{0,1\}^L$.

f-random	function	permutation $l = L$
$\Pr[f(X) = Y] =$	2^{-L}	$2^{-\ell}$
$\Pr[f(X_1) = Y_1 \mid f(X_2) = Y_2] =$	2^{-L}	$\begin{cases} \frac{1}{2^\ell - 1} & \text{if } Y_1 \neq Y_2 \\ 0 & \text{if } Y_1 = Y_2 \end{cases}$
$\Pr[f(X_1) = Y \text{ and } f(X_2) = Y] =$	$\begin{cases} 2^{-2L} & \text{if } X_1 \neq X_2 \\ 2^{-L} & \text{if } X_1 = X_2 \end{cases}$	$\begin{cases} 0 & \text{if } X_1 \neq X_2 \\ 2^{-\ell} & \text{if } X_1 = X_2 \end{cases}$
$\Pr[f(X_1) \oplus f(X_2) = Y] =$	$\begin{cases} 2^{-L} & \text{if } X_1 \neq X_2 \\ 0 & \text{if } X_1 = X_2 \text{ and } Y \neq 0^L \\ 1 & \text{if } X_1 = X_2 \text{ and } Y = 0^L \end{cases}$	$\begin{cases} \frac{1}{2^\ell - 1} & \text{if } X_1 \neq X_2 \text{ and } Y \neq 0^\ell \\ 0 & \text{if } X_1 \neq X_2 \text{ and } Y = 0^\ell \\ 0 & \text{if } X_1 = X_2 \text{ and } Y \neq 0^\ell \\ 1 & \text{if } X_1 = X_2 \text{ and } Y = 0^\ell \end{cases}$

Pseudorandom functions (PRFs)

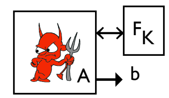
- Informally, a function family F is a PRF if the input-output behavior of its random instance is computationally indistinguishable from that of a random function.

PRFs

- <u>Def.</u> Fix a function family F: Keys(F) × Dom(F) → Range(F)

Experiment $\mathbf{Exp}_F^{\text{prf-1}}(A)$

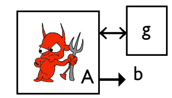
$K \xleftarrow{\$} \text{Keys(F)}$



Return b

Experiment $\mathbf{Exp}_F^{\text{prf-0}}(A)$

$g \xleftarrow{\$} \text{Func(Dom(F),Range(F))}$



Return b

The prf-advantage of an adversary A is

$$\mathbf{Adv}_F^{\text{prf}}(A) \;=\; \Pr\left[\mathbf{Exp}_F^{\text{prf-1}}(A) = 1\right] - \Pr\left[\mathbf{Exp}_F^{\text{prf-0}}(A) = 1\right]$$

F is a secure PRF if for any adversary with "reasonable" resources its prf-advantage is "small".

PRFs

- <u>Def.</u> Fix a function family F: Keys(F) × Dom(F) → Range(F)

Experiment $\mathbf{Exp}_F^{\text{prf-1}}(A)$	Experiment $\mathbf{Exp}_F^{\text{prf-0}}(A)$
$K \xleftarrow{\$} \mathcal{K}$	$g \xleftarrow{\$} \mathsf{Func}(D,R)$
$b \xleftarrow{\$} A^{F_K}$	$b \xleftarrow{\$} A^{g}$
Return b	Return b

The prf-advantage of an adversary A is

$$\mathbf{Adv}_F^{\text{prf}}(A) \;=\; \Pr\left[\mathbf{Exp}_F^{\text{prf-1}}(A) = 1\right] - \Pr\left[\mathbf{Exp}_F^{\text{prf-0}}(A) = 1\right]$$

F is a secure PRF if for any adversary with "reasonable" resources its prf-advantage is "small".

Resources of an adversary

- Time-complexity is measured in some fixed RAM model of computation and includes the maximum of the running-times of A in the experiments, plus the size of the code for A.
- The number of queries A makes.
- The total length of all queries.

Pseudorandom permutations (PRPs)

- Informally, a function family F is a PRF if the input-output behavior of its random instance is computationally indistinguishable from that of a random permutation.

PRPs under chosen-plaintext attacks (CPA)

- Def. Fix a function family F: Keys(F) × Dom(F) → Dom(F)

Experiment $\mathbf{Exp}_F^{\text{prp-cpa-1}}(A)$	Experiment $\mathbf{Exp}_F^{\text{prp-cpa-0}}(A)$
$K \stackrel{\$}{\leftarrow} \mathcal{K}$	$g \stackrel{\$}{\leftarrow} \mathsf{Perm}(D)$
$b \stackrel{\$}{\leftarrow} A^{F_K}$	$b \stackrel{\$}{\leftarrow} A^{g}$
Return b	Return b

The prp-cpa-advantage of an adversary A is

$$\mathbf{Adv}_F^{\text{prp-cpa}}(A) \;=\; \Pr\left[\mathbf{Exp}_F^{\text{prp-cpa-1}}(A)=1\right] - \Pr\left[\mathbf{Exp}_F^{\text{prp-cpa-0}}(A)=1\right]$$

F is a secure PRP under CPA if for any adversary with “reasonable” resources its prf-cpa-advantage is “small”.

PRPs under chosen-ciphertext attacks (CCA)

- Since an inverse function is defined for each instance, we can also consider the case when an adversary gets, in addition, an oracle for g^{-1}
- Def. Fix a permutation family F: Keys(F) × Dom(F) →Dom(F)

Experiment $\mathbf{Exp}_F^{\text{prp-cca-1}}(A)$	Experiment $\mathbf{Exp}_F^{\text{prp-cca-0}}(A)$
$K \stackrel{\$}{\leftarrow} \mathcal{K}$	$g \stackrel{\$}{\leftarrow} \mathsf{Perm}(D)$
$b \stackrel{\$}{\leftarrow} A^{F_K, F_K^{-1}}$	$b \stackrel{\$}{\leftarrow} A^{g, g^{-1}}$
Return b	Return b

The prp-cca-advantage of an adversary A is

$$\mathbf{Adv}_F^{\text{prp-cca}}(A) \;=\; \Pr\left[\mathbf{Exp}_F^{\text{prp-cca-1}}(A)=1\right] - \Pr\left[\mathbf{Exp}_F^{\text{prp-cca-0}}(A)=1\right]$$

- F is a secure PRP under CCA if for any adversary with “reasonable” resources its prf-cca-advantage is “small”.

PRP-CCA ⇒ PRP-CPA

- Theorem. Let F:Keys×D→D be a permutation family. Then for any adversary A that runs in time t and makes q chosen-plaintext queries these totalling μ bits there exists an adversary B that also runs in time t and makes q chosen-plaintext queries these totalling μ bits and no chosen-ciphertext queries such that

$$\mathbf{Adv}_F^{\text{prp-cca}}(B) \;\geq\; \mathbf{Adv}_F^{\text{prp-cpa}}(A)$$

Modeling block ciphers

- Want a "master" property that a block cipher be PRP-CPA or PRP-CCA secure.
- Conjectures:
  - DES and AES are PRP-CCA (thus also PRP-CPA) secure.
  - For any B running time t and making q queries

$$\begin{aligned}\mathbf{Adv}^{\text{prp-cpa}}_{\text{AES}}(B_{t,q}) &\leq c_1 \cdot \frac{t/T_{\text{AES}}}{2^{128}} + c_2 \cdot \frac{q}{2^{128}} \\ \mathbf{Adv}^{\text{prf}}_{\text{AES}}(B_{t,q}) &\leq c_1 \cdot \frac{t/T_{\text{AES}}}{2^{128}} + \frac{q^2}{2^{128}}\end{aligned}$$

The "birthday" attack

- Theorem. For any block cipher E with domain and range $\{0,1\}^\ell$ and any A that makes q queries s.t. $2 \leq q \leq 2^{(\ell+1)/2}$.

$$\mathbf{Adv}^{\text{prf}}_{E}(A) \geq 0.3 \cdot \frac{q(q-1)}{2^\ell}$$

- Lemma. If we throw (at random) q balls into N≥q bins and if $1 \leq q \leq \sqrt{2N}$ then the probability of a collision

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

Proof of the Lemma

// Using that $1 - x \leq e^{-x}$

// Using that if

Thus

Proof of the Theorem

- Adversary A^g

i-th l-bit string

For i=1,..q do $y_i \leftarrow g(\langle x_i \rangle)$ EndFor
If $y_i, \ldots y_q$ are all distinct return 1, else return 0
EndIf

$$\begin{aligned}\mathbf{Adv}^{\text{prf}}_{E}(A) &= \Pr\left[\mathbf{Exp}^{\text{prf-1}}_{E}(A) = 1\right] - \Pr\left[\mathbf{Exp}^{\text{prf-0}}_{E}(A) = 1\right] \\ &= 1 - [1 - C(N,q)] \\ &= C(N,q) \\ &\geq 0.3 \cdot \frac{q(q-1)}{2^l} .\end{aligned}$$