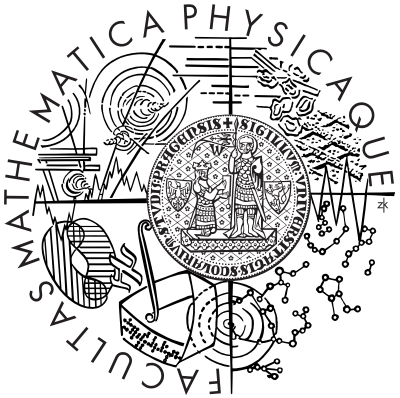
Another Introduction to Modern Cryptography



Miloslav Homer

Podzimní škola katedry algebry

Table of Contents

Symmetric Cryptography

- Encryption
- Decryption
- Key

Asymmetric Cryptography

Definition (One-way function)

Let $f : A \to B$ be a function. We say that f is *one-way* if and only if there exists polynomial time algorithm computing f, but any polynomial randomized algorithm computing f^{-1} succeeds with negligible probability (over the choices of x).

Goals of Cryptography

1. Confidentiality – the state or attribute of being secret; privacy.
2. Integrity – the state or quality of being entire or complete;wholeness.
3. Authentication – validating the genuineness of st. or someone.
4. Non-repudiation – not being able to reject or disclaim as invalid.

P

Definition (DTIME)

Let $T : \mathbb{N} \to \mathbb{N}$ be a function. We denote $DTIME(T(n))$ the set of all Boolean (one bit output) functions that are computable in $c.T(n)$-time for some constant $c > 0$.

Definition (class P)

$$P = \bigcup_{c \in \mathbb{N}} DTIME(n^c).$$

NP

Definition (NTIME)

Let $T : \mathbb{N} \to \mathbb{N}$ be a function. We denote $NTIME(T(n))$ the set of all Boolean (one bit output) functions that are computable in non-deterministic $c.T(n)$-time for some constant $c > 0$.

Definition (class NP)

$$NP = \bigcup_{c \in \mathbb{N}} DTIME(n^c).$$

Algorithmica

- A world where $P = NP$ or $NP \subseteq BPP$.
- Travelling salesmen no longer suffer from headaches.
- No cryptography available.
- Professor Grouse utterly fails to humiliate Gauss.

Heuristica

- *NP* problems intracable in worst-case, but tracable on average.
- No cryptography either - there exists hard enough problems, but we cannot find them efficiently.
- If the professor is lucky, he might be able to get his revenge.

Pessiland

- *NP* hard on average, no one-way functions.
- Still no cryptography - there exists hard enough problems, but we cannot find solved ones for our use.
- The professor can create a hard enough problem, but he is not able to solve it.

Minicrypt

- Hard problems, one-way functions, no secret communication through public only channels.
- Secure encryption, authentication, zero-knowledge protocols.
- Grousse can create a problem, which he can solve, but Gauss cannot.

Cryptomania

- As Minicrypt, but with secret communication through public only channels.
- E-voting, anonymous digital money, etc.
- Grousse can create a problem, which everybody in the class can solve, except for Gauss.

Definition (One-way function)

Let $f : A \to B$ be a function. We say that f is *one-way* if and only if there exists polynomial time algorithm computing f, but any polynomial randomized algorithm computing f^{-1} succeeds with negligible probability (over the choices of x).

Definition (One-way permutation)

We call bijective one-way function $f : A \to A$ a *one-way permutation*.

Indistinguishability

Definition (Computational indistinguishability)

Let $X = \{X_n\}_{n\in\mathbb{N}}$ and $Y = \{Y_n\}_{n\in\mathbb{N}}$ be probability ensembles such that each X_n and Y_n ranges over strings of length $n \in \mathbb{N}$. We say that X, Y are *computationally indistinguishable* if for every feasible algorithm A the difference

$$d_A(n) = |Pr(A(X_n) = 1) - Pr(A(Y_n) = 1)|$$

is a negligible function in n.

Pseudo-Random Number Generator (PRNG)

Definition (PRNG)

Let $l\colon \mathbb{N} \to \mathbb{N}$ be so that $\forall n \in \mathbb{N}\colon l(n) > n$. A *pseudorandom number generator* with *stretch function* l, is an efficient deterministic algorithm which on input n outputs a $l(n)$ bit sequence indistinguishable from an uniformely chosen $l(n)$ bit sequence.

Hard-core predicate

Definition (HC predicate)

We call function b a *hard-core predicate* of one-way function f if b is easy to evaluate but given $f(x)$ it is infeasible to predict $b(x)$.

PRNG construction from a hard-core predicate

Let f be a one-way permutation, let b be a hard-core predicate of f, let l be a stretch function. Then

$$G(s) = b(s).b(f(s)) \dots b(f^{l(|s|)-1}(s))$$

is a pseudorandom generator.

Pseudo-Random function family

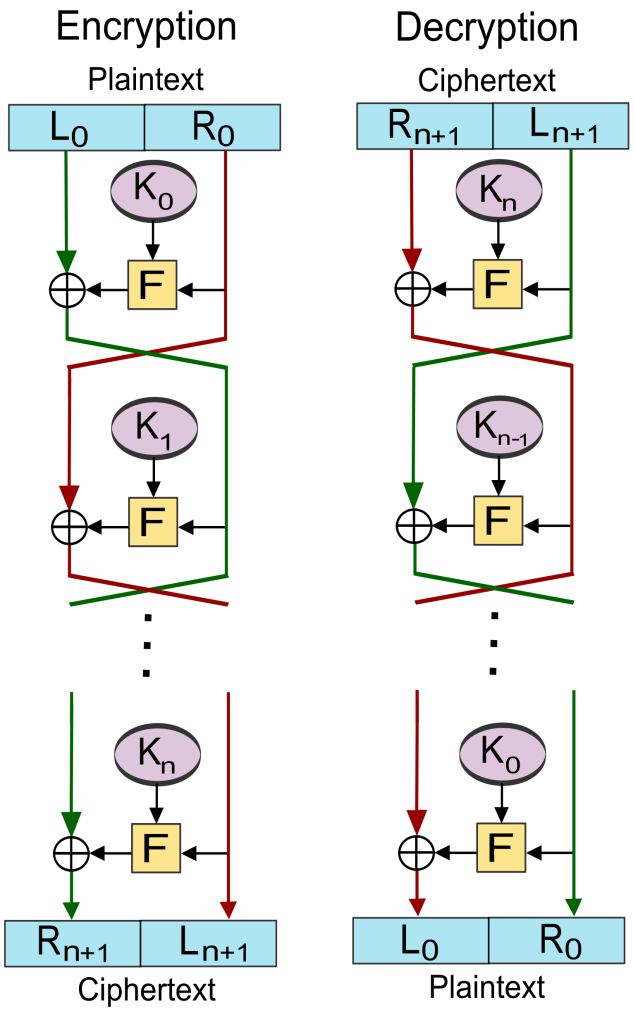
Definition (PRF)

By PRF we mean a collection of efficiently deterministic computable functions S such that for uniformely chosen $s \in |S|$ it is infeasible to distinguish responses of f_s and a truly random function.

Definition (Pseudo-Random permutation family)

By PRP we mean a PRF such that for all $s \in |S|$: f_s is a permutation.

PRF to PRP - Feistel cipher (Luby and Rackoff)



NP Proof Systems

Definition

Let $S \subseteq \{0,1\}^*$ and $\nu\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ be a function so that $x \in S$ if and only if there exists a $w \in \{0,1\}^*$ such that $\nu(x,w) = 1$. If ν is computable in time bounded by a polynomial in the length of its first argument then we say that S is an *NP set* and that ν defines a *NP proof system*.

Interactive Proof System

Definition

An *interactive proof system* for a set S is a two-party game, between a verifier (executing a probabilistic poly-time strategy (denoted V)) and a prover (executing computationally unbounded strategy (denoted P)), satisfying:

- *Completeness* – For every $x \in S$ verifier V always accepts after interacting with the prover P on common input x.
- *Soundness* – For every $x \notin S$ and every potential strategy P^*, the verifier V rejects with prob. at least $1/2$, after interacting with P^* on common input x.

Zero Knowledge Proof

Definition (Zero knowledge proof)

We say that a protocol is *zero-knowledge* if there exists a simulator running in polynomial time (that does not have access to a prover) whose output is computationally indistinguishable from a malicious verifier's output after interaction with a prover.

The Zero Knowledge Theorem

Theorem

Assuming the existence of one-way functions, any NP-proof can be efficiently transformed into a computational zero-knowledge interactive proof.

Non-Interactive Zero Knowledge Proof (NIZK)

Definition

We call the proof *non-interactive* if the interaction between prover and verifier consists of only one message sent by prover to the verifier.

Random Oracle

Definition (Random Oracle)

We call $f : A \to B$ a *random oracle* if and only if f responds to every unique $a \in A$ with a uniformly chosen random $b \in B$. If $a \in A$ is repeated, f responds with the same b.

There exist signature and encryption schemes that are secure in the Random Oracle Model, but for which any implementation of the random oracle results in insecure schemes (eprint 1998/011).

Common Reference String

Definition (Common Reference String)

By *common reference string* we mean a public (ie all parties have access to it) uniformely randomly selected string chosen before any protocol interaction starts.

Any questions?

Thank you for your attention!