CRYPTOGRAPHY LECTURE, 2022

Computer Science and Algorithmics, PWr

Mirosław Kutyłowski

Materials:

- webpage of my lecture,
- `internal` subpage with info only for participants
  - login: `student`
  - password: `R2D2`
  - contents: mp4's of the lectures, pdf's from the lectures
  - link to a nice textbook from CMU

Approach

- ”joy of cryptography”
- concrete but not too formal, too mathematical, but for clever students
- sometimes simple tasks (e.g. in Python) to touch concrete programming…
- … but the most important issue is to think (critically)
- not just learning schemes, techniques…

Social competence goal:

- do not trust anybody
- believe but check

Grading (reference information on the webpage):

- lecture: very short tests (5, max10 minutes) over MS Teams (probably) 0-2 points), a small security puzzle
- homeworks: short but concrete (0-2 points), deadlines strict
- in-class tests: short (but not very short), on paper, grading adjusted
- the sum of points determine the final grade according to the scale given on the webpage, 40% – course passed, over 80% -very good

Absence

- in case of illness, a serious personal problem etc please let us know (please do not present your medical documentation etc.)
- there will be time in June to handle all this cases (extra tests)

Contact

- email (preferred)
- MS Teams
- starosta – phone (let us exchange mobile phone numbers)
- office hours: do not take them too seriously, sometimes we meet online in the evening, etc.
  - first think if you can explain the issue in an email. Written information is less error prone

“osoby, które ze względu na stan zdrowia, niepełnosprawność lub inne obiektywne przesłanki mogą mieć szczególne potrzeby związane ze sposobem realizacji zajęć, zaliczenia bądź przygotowaniem materiałów proszone są o zgłoszenie się na konsultacje lub po zajęciach, napisanie takiej informacji na prywatnym czacie, bądź napisanie e-maila w tej sprawie. Będę starał/starała się, aby na moich zajęciach każdy miał równe prawo do zdobycia wiedzy i rozliczenia się z niej.”

“Ladies and Gentlemen, those who, due to their health condition, disability or other objective reasons, may have special needs related to classes leading, crediting method or materials preparation are asked to report for consultations or after the classes, write such information in a private chat or write an e-mail about the matter. I will try to ensure that during my classes everyone has an equal right to gain knowledge and its’ crediting."

I. Introduction

classical point of view:

- encryption, decryption, cryptanalysis

real scope (some key areas):

- asymmetric encryption
- authentication
- key agreement
- blockchain
- signatures, electronic seals,
- ...

Asymmetric encryption

Alice holds a key pair:

- public key X
- private key x

Bob creates a ciphertext of plaintext M with X:

- $C := \mathrm{Enc}_X(M)$

Alice can decrypt with x:

- $M' := \mathrm{Dec}_x(C)$

Properties: $M' = M$, without x it is impossible to derive ANY information on M

(Asymmetric) Authentication

Alice holds a key pair:

- public key X
- private key x

Bob knows that X is attributed to Alice (how???)

Alice interacts with Bob and

– proves that she holds a private key corresponding to X

– . . . but does not reveal x

Key agreement (e.g. in TLS)

Alice and Bob do not share a key

key agreement protocol output:

- Alice knows K
- Bob knows K
- nobody else can derive K based on the messages exchanged

(Yes, Diffie-Hellman protocol, but not only)

Blockchain

a data structure:

- the only operation supported is `append`
- modification, insertion, removal not at the end is **detectable** (and treated as a fraud)
- (physical modification is frequently easy)

Digital Signatures

Alice holds a key pair:

- public key X
- private key x

Alice signs M

- $s := \mathrm{Sign}_x(M)$

Signature Validation (Verification):

- $\mathrm{Test}(s, M, X, \ldots.)$

Feature:

- a person holding a key corresponding to X must have been creating s

History

simple rewriting methods like Caesar cipher

generally (substitution ciphers): permutation π on the alphabet

plaintext $P_1P_2P_3P_4P_5P_6\ldots.$ (tekst jawny)

ciphertext $\pi(P_1)\pi(P_2)\pi(P_3)\pi(P_4)\pi(P_5)\pi(P_6)\ldots.$

Problem: frequency analysis, bigrams, even worse for some languages

Challenge

- what to do if no electronic equipment should be used, ...
- ... and decryption /encryption operations performed not by a highly educated person?

Perfect Security

for every c and messages m_1, m_2

$$\Pr(c = \mathrm{Enc}_k(m_1) \text{ for } k \text{ random}) = \Pr(c = \mathrm{Enc}_k(m_2) \text{ for } k \text{ random})$$

Semantic Security

left-or-right game:

1. Alice chooses m_1, m_2

2. Bob picks b at random, creates $c := \mathrm{Enc}(m_b)$ and shows c

3. Alice returns a bit b'

Alice wins if $b' = b$.

Semantic security means: advantage of Alice is negligible, all she can do is to guess b

One-time pad

- key is a random bitstring K of length n
- plaintext: a bitsting P of length n
- ciphertext C:

  $C(i) := K(i) \otimes P(i)$ where $\otimes$ stands for XOR

Features:

- a key can be used at most once:

  $$C(i) \otimes C'(i) = (K(i) \otimes P(i)) \otimes (K(i) \otimes P'(i)) = P(i) \otimes P'(i)$$

- perfect security: for each pair (ciphertext-plaintext) there is exactly one matching key

What to do with long plaintexts?

Shannons Theorem

If Enc is a perfect encryption scheme, k is the keylength, plaintexts are m bit strings, then $k \geq m$.

Proof

One-way functions - (funkcja jednokierunkowa?)

perfect security is not possible in real life situation so looking for alternatives

- absolute security versus computational security

- adversary may guess the key ⇒ probability of a successful attack >0

What is admissible attack success probability p?

- if $p(n)$ is a **negligible function** (zaniedbywalna?):

  for each polynomial Q for almost all n : $p(n) < \frac{1}{Q(n)}$

- if $p < \frac{1}{2^{128}}$ (for example)

One-way function F

- it is easy to compute $F(x)$ (theoretical formulation: polynomial time)
- it is hard to compute F^{-1}: for a given y find any x such that $F(x) = y$
- what does it really mean “hard”?
  - for any (polynomial) adversary A
    1. choose x at random, $y := F(x)$, give y to adversary
    2. A outputs x'
    3. if $F(x') = F(x)$, then A wins
  - ”hard” means that probability of A to win is negligible

Example application -commitment

1. Alice chooses x, computes $y := F(x)$ and presents y to Bob

2. Bob presents z

3. Alice reveals x

4. Bob checks that $F(x) = y$

5. Alice and Bob compute $x + z \bmod 2$

(tossing a coin over Internet) , Bob cannot see what has been chosen by Alice

(there are some subtle issues. . . e.g. Alice can find x' such that $F(x') = y$)

Securing communication

Alice receives messages, she must be convinced that message $n+1$ comes from the same person as message n

Solution (e.g. µTesla protocol)

- message n contains metadata y where $y=F(x)$ for x chosen at random
- message $n+1$ contains x

for adversary: impossible to find x in a short time and authenticate own message

Weak one-way function

the same but probability to invert is limited, e.g. $<\frac{1}{2}$ or $\frac{1}{1000000}$

(it might be still useful but might be much cheaper)

One-way does not mean “no information about the input”

example:

- F a good one-way function
- $F'(x, y) = F(x) || y$
- F' is still one-way function

If P=NP then there are no one-way functions

nondeterministic polynomial algorithm for inverting one-way function F

- given y, guess x
- check that $F(x) = y$

If $P = \mathrm{NP}$, then there is a deterministic polynomial algorithm achieving the same, contrary to the definition of a one-way function

Candidates for one-way functions - via factorization

domain of F: pairs of prime numbers of length n

$F(p,q) = p \cdot q$

Complexity of factorization:

- school method: $\approx 2^n$

- number thieve, etc: still more than $2^{c \cdot \log n}$

- but be careful: check what is the secure size (e.g. for $n = 500$ these methods are still practical)

Candidate for a weak one-way function

domain of F: pairs of numbers of length n

$F(p,q) = p \cdot q$

Why?

Thm. about density of prime numbers

probability that an n bit number chosen at random is prime is at least $\frac{1}{2n}$

Hardcore predicate H for one-way function F

given $y = F(x)$, is $H(x)$=true ?

that is: we are looking for properties of $F^{-1}(y)$ instead of x

formally:

for any polynomial adversary A, a game:

1. choose x at random, $y := F(x)$
2. give y to A
3. A outputs b
4. A wins if $b = H(x)$

probability to win should be at most $\frac{1}{2} + \varepsilon$ where ε is negligible

Inner product

let F be one-way function

then inner product is a hardcore predicate

$y = F(x)$, $h_r(x) = <r, x>$

Proof sketch

take r - a bitstring with 1 on position i and 0 elsewhere

$h_r(x) = x_i$

so knowing $h_r(x)$ for any r we could reconstruct x bit by bit

Candidate for a weak one-way function- exponentiation

$F(x) = g^x$ in a finite multiplicative cyclic group with generator g

e.g.

$F(x) = g^x \bmod p$, where p is a large prime

$F(x) = x \cdot P$, where P is a point on elliptic curve (group with addition of EC points)

Inverse function: so called *discrete logarithm*

Discrete Logarithm Problem Assumption for G

in the group G exponentiation is a one-way function

Application - Pedersen Commitment

- cyclic group with hard DLP
- random generators g and h ($\log_g(h)$ must be unknown)
- commitment for x:
  - choose r at random
  - $c := g^x \cdot h^r$
- Claim: cheating at opening would enable calculating $\log_g(h)$, as DLP is not solvable, cheating is impossible, too

indeed, if $g^{x'} \cdot h^{r'} = c = g^x \cdot h^r$, then $h = g^{(x'-x)/(r-r')}$ and DLP problem can be solved for h

LPN Learning Parity with Noise - candidate

input: string s

output: a sequence of pairs $(a_i, s \otimes a_i +_2 e_i)$ where e_i is a noise vector with Bernoulli distribution, with a relatively small error probability (but still not too small)

LPN hardness – looks like a simple linear algebra but noise makes a difference

– to apply linear algebra we would have to guess correctly the position of errors