Seminar: Algorithms of IT Security and Cryptography

Zero-Knowledge Proofs and Protocols

Nikolay Vyahhi

June 8, 2005

Abstract

A proof is whatever convinces me.
Shimon Even, 1978.

Zero-knowledge proof is usual proof, but you must not give more information to verifier, than your statement (which you prove) can give alone. So, in this paper, some facts about zero-knowledge, interactive protocols and proofs will be given. Also, with examples.

Contents

1 Introduction

Main idea of zero-knowledge is to prove some fact (theorem for example) to another person, but don't give him more information, than only this fact.

1.1 Applications

Applications of zero-knowledge are:

- authentication (user proves to system, that he is valid user)
  Weakness: Adversary E can prove to B, that she is A, just by asking A to prove it to her and simulating this protocol with B.
- protecting against chosen message attack by augmenting the ciphertext by a zero-knowledge proof of knowledge of the cleartext.
- non-oblivious commitment schemes

2 Theory

2.1 Interactive Proof Systems and Interactive Protocol

Intuitively, what should we require from an efficient theorem-proving procedure?

- That it should be possible to "prove" a true theorem.
- That it should be impossible to "prove" a false theorem.
- That communicating the "proof" should be efficient. Namely regardless of how much time it takes to come up with the proof, its correctness should be efficiently verified.

More formal. An **interactive Turing machine (ITM)** is a Turing machine equipped with read-only input tape, a work tape, a random tape, one read-only and one write-only communication tapes. The random tape contains an infinite sequence of random bits, and can be scanned only from left to right.

An **interactive protocol** is an ordered pair of ITM's A (prover) and B (verifier) such that A and B share the same input tape, B's write-only communication tape is A's read-only communication tape and vice versa. And machine A is not computationally bounded, while B is bounded by a polynomial in the length of common input. These two machines take turns in being active, with B being active first. During an active stage $A(B)$ perform some internal computation using its tapes; and then it writes a string (for $B(A)$) on its write-only communication tape. Then it deactivates and machine $B(A)$ becomes active. Machine B accepts (or rejects) the input by outputting "accept" (or "reject") and terminating the protocol.

An interactive protocol (A,B) is called an **interactive proof system** for language L over $(0,1)^*$ if we have the following:

- For each k, for sufficiently large $x \in L$ given as input to (A, B), B halts and accepts with probability at least $1 - |x|^{-k}$.
- For each k, for sufficiently large $x \notin L$, for any ITM A', on input x to (A', B), B accepts with probability at most $|x|^{-k}$.

The probabilities here are taken over the readings of random bits of A and B.

Interactive Polynomial time (IP) is the class of languages for which there exists interactive proof system.

2.2 QNR example

Informally, zero-knowledge means that for every polynomial time B', the distribution that B' "sees" on all its tapes, when interacting with A on input $x \in L$, is "indistinguishable" from a distribution that can be computed from x in polynomial time.

Let's consider

$$\text{QNR} = \{\ (x, y) — y \text{ is quadratic nonresidue mod } x\ \}$$

That means, that there is no such z, that $y = z^2\ mod\ x$. So lets try to prove with zero-knowledge for some y, that it is from QNR. With prover A, verifier B, input (x, y) and $|x| = n$.

- B begins by flipping coins to obtain random bits $b_1, b_2, ...b_n$.
- Then B flips additional coins for obtaining random $z_1, z_2...z_n$ ($0 < z_i < x$ and $gcd(z_i, x) = 1$ for each z_i).
- B computes $w_1, w_2, ...w_n$ as follows:

$$w_i = (z_i^2)\ mod\ x,\ if\ b_i = 0$$

$$w_i = (z_i^2 y)\ mod\ x,\ otherwise,\ if\ b_i = 1$$

- B sends $w_1, w_2, ...w_n$ to A.
- A computes (somehow) for each i whether or not w_i is quadratic residue mod x, and sends this information $(c_1, c_2, ...c_n)$ to B.
- B checks if $b_i = c_i$ for every i, and if so is "convinced" that $(x, y) \in QNR$.

It seems to be zero-knowledge proof, but... What if B were to cheat? B could begin by setting $w_1 = 42$ for example, and then behave correctly. So, B can compute whether or not 42 is a quadratic residue x, given x and a quadratic nonresidue y. At this time it is not known how compute this in polynomial time, so this proof system may not be zero-knowledge! How to construct right zero-knowledge proof of QNR we'll see later.

2.3 Indistinguishability of Random Variables

Consider **families of random variables** $U = U(x)$, where $x \in L$, a particular subset of $\{0, 1\}^*$, and all random variables take values in $\{0, 1\}^*$.

Let $U(x)$ and $V(x)$ be two families of random variables. We want to express the fact that, when the length of x increases, $U(x)$ essentially becomes "replaceable" by $V(x)$. So, a random sample is selected form $U(x)$ or from $V(x)$ and it is handed to a "judge". After studying the sample, he proclaims, from which families our sample is.

Two families of random variables $\{U(x)\}$ and $\{V(x)\}$ are:

- **Equal** if the judges verdict will be meaningless even if he is given samples of arbitrary size and he can study them for an arbitrary amount of time.
- **Statically indistinguishable** if the judges verdict became meaningless when he is given an infinite amount of time but only random, polynomial (in $|x|$) size samples to work on.
- **Computationally indistinguishable** if the judges verdict become meaningless when he is only given polynomial ($|x|$)-size samples and polynomial ($|x|$) time.

2.4 Approximability of Random Variables

Let M be a probabilistic Turing machine that on input x always halts. We denote by $M(x)$ the random variable that, for each string, which is equal to α, have the same probability that M on input x outputs α.

U is **perfectly approximable** on L if there exist a probabilistic Turing machine M, running expected polynomial time, such that for all $x \in L$, $M(x)$ is equal to $U(x)$.

U is **statically (computationally) approximable** on L if there exist a probabilistic Turing machine M, running expected polynomial time, such that for families of random variables $\{M(x)\}$ and $\{U(x)\}$ are statically (computationally) indistinguishable on L.

2.5 Zero-Knowledge

So, ITM B' has an extra input tape H, which length is bounded above be a polynomial in the length of x. When B' interacts with A, A sees only x on its tape, whereas B' sees (x, H). H is just a some knowledge about x that the cheating B' already possess. Or H can be considered as the history of previous interactions that B' is trying to use to get knowledge from A.

Let $View_{A,B'}(x, H)$ be the random variables whose value is view of B' (random tape, messages between parties, private tape). For convenience, we consider each view to be a string from $\{0, 1\}*$ of length $|x|^c$ for some fixed $c > 0$.

We say that (A, B) is **perfectly (statically) (computationally) zero-knowledge on** L **for** B' if the family of random variables $View_{A,B}$ is perfectly (statically) (computationally) approximable on

$$L' = \{(x, H)|x \in L \; and \; |H| = |x|^c\}.$$

And, at last, an interactive protocol (A, B) is **perfectly (statically) (computationally) zero-knowledge on L** if it is perfectly (statically) (computationally) zero-knowledge on L for all probabilistic polynomial time ITM B'. Note, that this definition only depends on A and not at all on B.

Usually, only computationally zero-knowledge is consided.

2.6 Known Facts and Open Problems

- Every language in NP has a perfect zero knowledge proof (if one-way permutations exists).
- Every language in IP has a zero knowledge proof.
- Its known that (obvious)

  $$BPP \subseteq PZK \subseteq SZK \subseteq CZK \subseteq IP$$

  BPP means class of bounded probabilistic polynomial time problems, PZK (SZK, CZK) is the class of languages for which there exists perfectly (statically, computationally) zero-knowledge proof, IP - interactive proof.
- Goldreich's belief is that

  $$BPP \subset PZK \subseteq SZK \subset CZK = IP$$

- The relationship of PZK and SZK remains an open problem (with no evidence either way).

3 Exapmles

3.1 Graph Isomorphism

Problem (GI Graph Isomorphism): You have two graphs (G_0, G_1), are they isomorphic?

- A chooses one graph (G_0 or G_1), and transform it to any another isomorphic one G_2 (anyhow).
- A sends this graph G_2 to B.
- B flips a coin, and sends this bit b (0 or 1) to A. A must show isomorphism of G_2 and G_b to B, otherwise B can not accept.

So, if A cheating, she can't show isomorphism of those two graphs with probability $\frac{1}{2}$. But A can cheat with $\frac{1}{2}$ probability also. And if B repeats this protocol n times, A can cheat at most with probability only $\frac{1}{2}^n = 2^{-n}$. At last, B can't get some additional information from this interaction, so it's really zero-knowledge proof.

3.2 Graph NonIsomorphism

Problem (GNI - Graph NonIsomorphism): You have two graphs (G_0, G_1), are they nonisomorphic?

- B chooses one graph (G_0 or G_1), and transform it to any another isomorphic one G_2 (anyway).
- B sends this graph G_2 to A.
- A must say, which graph was chosen by B.

If A cheating, so graphs G_0 and G_1 are isomorphic, and she can not say exactly, to which one G_2 is isomorphic. Probability of being caught is $1 - \frac{1}{2}^n$. But (!) B can get some additional information from this interaction. So it's not zero-knowledge at all. Here, we can see the same situation, like with QNR earlier.

So, we must modify verifier B, so that he'll prove to the prover A, that he (B) knows the answer to his query graph (i.e. he knows an isomorphism to the appropriate input graph), and the prover answers the query only if she is convinced of this claim. Of course, that B's proof must be zero-knowledge.

3.3 Quadratic NonResidue

Like we saw earlier, **Problem (QNR - Quadratic NonResidue)**:

$$\text{QNR} = \{\ (x, y) \text{ — } y \text{ is quadratic nonresidue mod } x\ \}$$

That means, that there is no such z, that $y = z^2\ mod\ x$. Let's consider a zero-knowledge proof for this.

- B picks a random integer r and one *bit*. if $bit = 0$ then B sets $w = r^2\ mod\ x$, otherwise $w = r^2 y\ mod\ x$. B sends w to A.

- For some $1 \leq j \leq m$, B picks random integer r_{j_1}, r_{j_2} and random bit_j. B sets

  $$a_j = r_{j_1}^2 \; mod \; x$$

  $$b_j = yr_{j_2}^2 \; mod \; x$$

  If $bit_j = 1$, B sends A the ordered pair (a_j, b_j), else (b_j, a_j).

- A sends B an m-long random bit vector $i = i_1, i_2, ...i_m$.

- B sends A the sequence $v = v_1, v_2, ...v_m$.

  - if $i_j = 0$ then $v_j = (r_{j_1}, r_{j_2})$
  - if $i_j = 1$ then
    - if $bit = 0$ then $v_j = rr_{j_1} \; mod \; x$
    - else $v_j = yrr_{j_2} \; mod \; x$.

  The intuition behind this step is as follows: if $i_j = 0$, then B is convincing A that pair was chosen correctly; if $i_j = 1$ then B is convincing that if pair was chosen correctly, then w was chosen correctly.

- A verifies that the sequence v was properly constructed. If not, A sends terminate to B and halts. Otherwise. A sets answer $= 0$ if w is a quadratic residue mod x and 1 otherwise, A sends answer to B.

- B checks whether $answer = bit$. If so B continues the protocol, otherwise B rejects and halts.

- After m repetition of this protocol, if B did not reject thus far, B accepts and halts.

Conclusion: So, we force B to prove, that he is not cheating. And now he can not obtain any other information from this protocol (only is y a quadratic nonredisue or not). $\Rightarrow$ It's a (statically) zero-knowledge proof.

4 Non-Interactive ZK Proofs

Non-interactive proofs used when A and B can't interact directly. General Idea is to use one-way function instead of verifier B. So, for example:

- A generates n random numbers, and so generates n different isomorphic (to initial) problems.
- A publish all this new problems.
- A uses one-way functions, to generate "random" bit string b from definitions of that new problems, which was published (it'll be like B's random tape).
- If $b_i = 0$ then A proves isomorphism of initial and i-th new problem, otherwise she opens solution of i-th new problem. Then A publish this information.
- Anyone can verify this proof without interaction.

But, A must chose large n, otherwise it'll be simple to cheat (for A), because A has more time than in online interaction with B.

References

[GMR89] S. Goldwasser, S. Micali, C. Rackoff. The knowledge complexity of interactive proof systems, 1989 (1986 originally).

[FFS88] U. Fiege, A. Fiat, A. Shamir. Zero-Knowledge Proofs of Identity, 1988.

[Schneier96] B. Schneier. Applied Cryptography, 1996.

[Goldreich01] O. Goldreich. Foundation of Cryptography, 2001.