

Cryptography CS 555

Lecture 20-b



Zero-Knowledge Proof

Review

- Identification (entity authentication)
 - password
 - one-time password
 - challenge-response
 - three kinds of challenges
 - symmetric key as well as asymmetric key based

Lecture Outline

- Zero knowledge proof protocols



Fiat-Shamir ID protocol (ZK Proof of knowledge of square root modulo n)

- System parameter: $n=pq,$
- Public identity: $v \quad v = s^2 \text{ mod } n$
- Private authenticator: s
- Protocol (repeat t times)
 1. A: picks random r in Z_n^* , sends $x=r^2 \text{ mod } n$ to B
 2. B checks $x \neq 0$ and sends random c in $\{0,1\}$ to A
 3. A sends y to B, where If $c=0$, $y=r$, else $y=rs \text{ mod } n$.
 4. B accept if $y^2 \equiv xv^c \text{ mod } n$

Properties Zero-Knowledge Proofs

- Properties of ZK Proofs:
 - completeness
 - honest prover who knows the secret convinces the verifier with overwhelming probability
 - soundness (is a proof of knowledge)
 - no one who doesn't know the secret can convince the verifier with nonnegligible probability
 - zero knowledge
 - the proof does not leak any additional information
- How to formalize soundness and ZK?

Formalizing the Soundness Property

- The protocol should be a “proof of knowledge”
- A knowledge extractor exists
 - that given a prover who can successfully convince the verifier, can extract the secret
- Knowledge extractor for the Fiat-Shamir Protocol
 - Given an algorithm A that can convince a verifier,
 - After A has sent x , first challenge it with 0, and receives y_1 such that $y_1^2 = x$
 - then reset A to the state after sending x , challenge it with 1 and receives y_2 such that $y_2^2 = xv$, then compute $s = y_1^{-1}y_2$, we have $s^2 = v$

Formalizing ZK property

- For every possible verifier algorithm, a simulator exists
 - taking what the verifier knows before the proof, can generate a communication transcript that is indistinguishable from one generated during ZK proofs
 - honest verifier ZK considers only the verifier algorithm in the protocol
- Three kinds of indistinguishability
 - perfect (information theoretic)
 - statistical
 - computational

Fiat-Shamir is honest-verifier ZK

- The transcript of a protocol run consists of t tuples (x, c, y) such that x is a random QR in Z_n^* and $y^2 \equiv xv^c \pmod n$
- Proof that Fiat-Shamir is honest verifier ZK
- Construct a simulator as follows
 - Repeat the following: pick random $c \in \{0, 1\}$,
 - if $c=0$, pick random r and outputs $(r^2, 0, r)$
 - if $c=1$, pick random y , and outputs $(y^2v^{-1}, 1, y)$
- The transcript generated by the simulator is from the same prob. distribution

Fiat-Shamir is ZK

- Given any possible verifier V , construct a simulator as follows
 - Repeat the following: pick random $c \in \{0, 1\}$,
 - if $c=0$, pick random r and send r^2 to V , if V challenges 0, output (r^2, c, r) ; otherwise reset V
 - if $c=1$, pick random y , compute $x=y^2v^{-1}$, and send x to V , if V challenges 1, output $(x, 1, y)$; otherwise reset V
 - Observe that r^2 and y^2v^{-1} have the same prob. distribution, the success prob. of one round is at least $\frac{1}{2}$

Running in Parallel

- All rounds in Fiat-Shamir can be run in parallel
 1. A: picks random r_1, r_2, \dots, r_t in Z_n^* , sends $x_1=r_1^2$, $x_2=r_2^2$, ..., $x_t=r_t^2$ to B
 2. B checks the x 's are not 0 and sends t random bit to A
 3. A sends y_1, y_2, \dots, y_k to B,
 4. B accept if $y_j^2 \equiv x_j v^{c_j} \pmod n$
- Is this protocol still a proof of knowledge?
- Is this protocol still honest verifier ZK?
- Is this protocol still ZK?

Schnorr Id protocol (ZK Proof of Discrete Log)

- System parameter: p, q, g
 - $q \mid (p-1)$ and g is an order q element in Z_p^*
- Public identity: v
- Private authenticator: s $v = g^s \text{ mod } p$
- Protocol
 1. A: picks random r in $[1..q]$, sends $x = g^r \text{ mod } p$,
 2. B: sends random challenge e in $[1..2^t]$
 3. A: sends $y=r+sc \text{ mod } q$
 4. B: accepts if $x = (g^y v^{-e} \text{ mod } p)$

Security of Schnorr Id protocol

- probability of forgery: $1/2^t$
- soundness (proof of knowledge):
 - if A can successfully answer two challenges e_1 and e_2 , i.e., A can output y_1 and y_2 such that $x = g^{y_1}v^{-e_1} = g^{y_2}v^{-e_2}$ then $g^{y_1-y_2} = v^{c_1-c_2}$ and thus the secret $s = (y_1 - y_2)(c_1 - c_2)^{-1} \pmod q$
- ZK property
 - honest verifier ZK, why?
 - not ZK if $2^t > \log n$ is used, why?

Converting Interactive ZK to Non-interactive ZK

- The only interactive role played by the verifier is to generate random challenges
 - challenges not predictable by the prover
- The same thing can be done using one-way hash functions

Interactive ZK Implies Signatures

- Given a message M , run all rounds in parallel,
 - generate the commitments all at the same time, let X denote all commitments
 - replace the random challenge of the verifier by the one-way hash $c=h(M||X)$
 - append the response

Schnorr Signature

Key generation (uses $h:\{0,1\}^* \rightarrow \mathbb{Z}_q$)

- Select two primes p and q such that $q \mid p-1$
- Select $1 \leq a \leq q-1$
- Compute $y = g^a \bmod p$

Public key: (p, q, g, y)

Private key: a

Schnorr Signature

Signing message M

- Select random secret k , $1 \leq k \leq q-1$

- Compute

$$r = g^k \text{ mod } p,$$

$$\mathbf{e = h(M || r)}$$

$$\mathbf{s = ae + k \text{ mod } q}$$

Signature is: (r, s)

To verify that (r,s) is the signature of M

- Compute

$$e = h(M || r)$$

- Verify that

$$r = g^{sy-e} \text{ mod } p$$

Summary

- ZK identification = ZK proof of knowledge
 - completeness
 - soundness (is a proof of knowledge)
 - zero-knowledge (weaker property: honest verifier ZK)
- Fiat-Shamir
 - proof of knowing a square root
- Schnorr
 - proof of knowing a discrete log