ZK Proofs II

Tuesday, September 12, 2017 12:05 PM

Today:

- Finish definition of ZKPok
- ZK for more languages

$$\mathrm{ZKPok}_x \{ (w) : \overbrace{L(x, w)}^{\text{"statement"}} \}$$

(CS 173) statement witness predicate

Languages

a language L is a set of strings

$x \in L$

$L : S \times W \to \{0,1\}$ (statement, witness)

L is an NP-language:

$x \in L$ iff $\exists w$ s.t. $L(x, w) = 1$

Def'n:

A ZKPok scheme for language L is a P.P.T. Prover (P) and verifier (V) satisfying:

- Correctness: $\forall x \in S,\ w \in W,\ \underline{\underline{L(x,w) = 1}}$,

  $\mathrm{Output}_V[P(x,w) \longleftrightarrow V(x)] = 1$

- Honest Verifier Zeroknowledge "Simulatability":

  $\exists S,\ \{\mathrm{View}_V[P(x,w) \longleftrightarrow V(x)]\}$

  comp. indisting. $\approx_c\ \{S(x)\}$

Crypto Joke

Fire Distinguisher

- Extractable:

  $\forall A.\ \Pr[\mathrm{Output}_V[A(x) \longleftrightarrow V(x)] = 1]$ w. non-negl. probability

  then $\exists \mathcal{E}_A$ s.t. $\Pr[w \leftarrow \mathcal{E}_A(x) : L(x,w) = 1] = 1 - \mathrm{negl.}$

  extractor

For the Schnorr protocol:

— Correctness $g^s = g \ldots = (g^{\ldots})(g^{\ldots}) = K X^c$

— Simulation: $S(x) = c \xleftarrow{\$} \mathbb{Z}_{|G|} \setminus \{0\},\ s \xleftarrow{\$} \mathbb{Z}_{|G|},\ K = g^s / X^c$ ✓

— Extractor: $\mathcal{E}_\mathcal{A}(x)$

Suppose $\mathcal{A}$ s.t. $\Pr[\mathcal{A}(1^\lambda) \leftrightarrow V \text{ output } 1] = P(\lambda)$ ← non negligible

$$\mathcal{A} \xrightarrow{K} V,\quad \xleftarrow{c},\quad \xrightarrow{s}$$

Define $\mathcal{E}_\mathcal{A}(x)$ as:

Run $\mathcal{A}(x)$ until it outputs K.

Make a "snapshot" of $\mathcal{A}$ as $\mathcal{A}'$

Sample $c_1 \xleftarrow{\$} \mathbb{Z}_{|G|} \setminus \{0\}$ ← set difference

$c_2 \xleftarrow{\$} \mathbb{Z}_{|G|} \setminus \{0\}$

Let $s_1 \leftarrow \mathcal{A}'(c_1)$

$s_2 \leftarrow \mathcal{A}'(c_2)$

Note that with p^2, $\left.\begin{aligned} X^{c_1} K &= g^{s_1} \\ X^{c_2} K &= g^{s_2} \end{aligned}\right\}$ Check this, repeat as necessary!

We solve for $\boxed{x} = (s_1 - s_2)/(c_1 - c_2)$ ← extended euclidean algorithm

$\mathbb{F}_p$

$X^{(c_1 - c_2)} = g^{(s_1 - s_2)}$

$g^{(s_1 - s_2)/(c_1 - c_2)} = X$

Comp. Sound.: $x \in L$, Prover doesn't necessarily "know" w

Extending ZKPoK to other languages:

$$\mathrm{ZKPoK}\ \{(a,b) : g^a = A,\ g^b = B\}$$

— repeat Schnorr twice?

— Use the same c?

$P(a,b)$ $\qquad\qquad V(A,B)$

$k_1 \xleftarrow{\$} \mathbb{Z}_p$

$k_2 \xleftarrow{\$} \mathbb{Z}_p$

$$\xrightarrow{K_1, K_2} \quad c \xleftarrow{\$} \mathbb{Z}_p \setminus \{0\}$$

$$\xleftarrow{c}$$

$s_1 = k_1 + ca$

$s_2 = k_2 + cb$ $\qquad \xrightarrow{s_1, s_2}$ Check $g^{s_1} \stackrel{?}{=} A^c K_1$

$g^{s_2} \stackrel{?}{=} B^c K_2$

To check:
- Correctness
- Simulatability
- extractability

$View_V = (\underbrace{K_1, K_2}, \underbrace{c, s_1, s_2})$

Commitments:

(com, open)

- hiding $com_r(x) \to c$ reveal nothing about x for $r \xleftarrow{\$} \mathbb{Z}_p$

r ← blinding

- binding Cannot generate collision (r, r', x, x', c) s.t.

$open(r, x, c) = 1$

$open(r', x', c) = 1$

Pedersen Commitment:

Uses $h \xleftarrow{\$} G \setminus \{e\}$ (an alternate generator)

$Com_r(x) = g^x h^r$ ← blinding

$open(r, x, C)$: check $C \stackrel{?}{=} g^x h^r$

Hiding: given x, bijection from r to C.

$f_x(r) = g^x h^r$ $h = g^z$

$g^x g^{zr} = g^{x+zr}$

Binding: Reduction to Discrete Log New Proof Technique!!

Suppose $\mathcal{A}$ s.t. $h \xleftarrow{\$} G$

$Adv_{\mathcal{A}}^{Binding} = Pr\left[(c, r_1, r_2, x_1, x_2) \leftarrow \mathcal{A}(1^\lambda, h) \begin{array}{l} open(r_1, x_1, c) = 1 \\ \text{and } open(r_2, x_2, c) = 1 \end{array}\right] \neq negl$

Then, we construct $\mathcal{A}'$ that wins DLOG

$\mathcal{A}'(X)$ ← group element

goal: output x s.t. $X = g^x$

Let $h = X$

(repeat)

$(c, r_1, r_2, x_1, x_2) \leftarrow A(1, h)$ poly ... if needed

until $c = g^{x_1} X^{r_1} = g^{x_2} X^{r_2}$

Solve $x = (x_1 - x_2)/(r_2 - r_1)$

$$g^{x_1 - x_2} = X^{r_2 - r_1}$$

$$X = g^{(x_1 - x_2)/(r_2 - r_1)}$$

output x. Solution

ZKPoK $\{(x, r) : C = g^x h^r\}$