Zero-Knowledge Proof.

- allows Prover to convince Verifier that a certain fact is true without giving any information.
- involves a number of challenge-response communication rounds between Prover and Verifier.
  - announcement ; Prover → Verifier
  - challenge ; Verifier → Prover
  - response ; Prover → Verifier
  - verify ; Verifier decides whether to accept or reject.

Proof of Knowledge for the Square Root.

Ex. Let $n = pq$ be the product of two large primes.

Let y be a square mod n with $\gcd(y, n) = 1$ i.e., $x^2 \equiv y \mod n$ for some x.

Prover claims to know a square root x of y. → finding square root mod n is equivalent to factoring n.

RSA hardness problem

$P \Rightarrow V$ i) [announcement]

- chooses a random $r \xleftarrow{\$} \mathbb{Z}_n$
- computes $s \equiv r^2 \mod n$ and sends s to Verifier.

$V \Rightarrow P$ ii) [challenge]

- chooses $\beta \in \{0, 1\}$ and sends β to Prover

↳ if y is not a square, only one (s or ys) is a square modulo n

⇒ 50% Prover will not be able to answer.

⤳ repeat k times.

$P \Rightarrow V$ iii) [response]

- if $\beta = 0$, then $z \equiv r \mod n$

  if $\beta = 1$, then $z \equiv xr \mod n$
- sends z to Verifier

iv) [verification]

- computes $z^2 \mod N$
- if $\beta = 0$, check $z^2 \equiv s \mod n$

  if $\beta = 1$, check $z^2 \equiv ys \mod n$

  $\because z^2 \equiv x^{2\beta} r^2 \equiv y^\beta s \mod n$
- if this is true, then Verifier accepts.

  otherwise, Verifier rejects.