Designing a ZK-SNARK to Prove Knowledge of Roots of a Quadratic Equation.

1. The Two Types of "Roots"

- **Roots of the equation** (secret):
  The values that satisfy the equation (e.g., $x = 5$ and $x = 10$ for $x^2 - 15x + 50 = 0$). These are **private** and must never appear in the circuit.
- **Roots of the vanishing polynomial** (public):
  The gate locations (e.g., $x = 1, 2$) where constraints are enforced. These are **public** and define $Z(x)$.

2. Why $Z(x)$ ≠ Equation Roots

Using secret roots (e.g., $Z(x) = (x - 5)(x - 10)$) would:

- **Reveal the solution** (since $Z(x)$ is public).
- **Break zero-knowledge**.

Instead, we:

1. **Encode the equation** into constraints at **public gate points** (e.g., $x = 1, 2$).
2. Use **public** $Z(x)$ (e.g., $Z(x) = (x - 1)(x - 2)$) to enforce constraints.

---

3. Correct Approach for $x^2 - 15x + 50 = 0$ with Root $x = 5$

We'll prove knowledge of $x = 5$ using gates at $x = 1, 2$ and $Z(x) = (x - 1)(x - 2)$.

Step 1: Circuit Design

Gate	Left (L)	Right (R)	Output (O)	Constraint
1	x	x	x^2	$x \cdot x = x^2$
2	x^2	1	$15x - 50$	$x^2 = 15x - 50$

Witness ($x = 5$):

- Gate 1: $5 \cdot 5 = 25$
- Gate 2: $25 = 15 \cdot 5 - 50$ (since $75 - 50 = 25$)

Step 2: Interpolate Polynomials at Gates $x = 1, 2$

x	$L(x)$	$R(x)$	$O(x)$
1	5	5	25
2	25	1	25

- $L(x)$ (Left inputs: 5, 25):

$$L(x) = 5\frac{(x-2)}{(1-2)} + 25\frac{(x-1)}{(2-1)} = 20x - 15$$

- $R(x)$ (Right inputs: 5, 1):

$$R(x) = 5\frac{(x-2)}{-1} + 1\frac{(x-1)}{1} = -4x + 9$$

- $O(x)$ (Outputs: 25, 25):

$$O(x) = 25 \quad (\text{constant})$$

Step 3: Construct $W(x) = L(x) \cdot R(x) - O(x)$

$$) = (20x - 15)(-4x + 9) - 25 = -80x^2 + 180x + 60x - 135 - 25 = \boxed{-80x^2 + 240x - 160}$$

Step 4: Vanishing Polynomial $Z(x) = (x-1)(x-2)$

$$Z(x) = x^2 - 3x + 2$$

Step 5: Divide $W(x)$ by $Z(x)$

$$\frac{-80x^2 + 240x - 160}{x^2 - 3x + 2} = -80$$

Exactly! So:

$$W(x) = -80 \cdot Z(x) \quad \text{and} \quad H(x) = -80$$

4. Trusted Setup & Proof Generation

1. **SRS**: Publish $[g, g^s, g^{s^2}]$ (degree 2).
2. **Prover** (with $x = 5$):
   - Computes encrypted evaluations:

$$g^{L(s)} = g^{20s-15}, \quad g^{R(s)} = g^{-4s+9}, \quad g^{O(s)} = g^{25}, \quad g^{H(s)} = g^{-80}$$

   - Sends these to the verifier.

5. Verification

Verifier checks:

$$\text{pairing}(g^{L(s)}, g^{R(s)}) \stackrel{?}{=} \text{pairing}(g^{O(s)}, g) \cdot \text{pairing}(g^{Z(s)}, g^{H(s)})$$

Expanding:

$$\underbrace{\text{pairing}(g^{20s-15}, g^{-4s+9})}_{j^{(20s-15)(-4s+9)}} \stackrel{?}{=} \underbrace{\text{pairing}(g^{25}, g)}_{j^{25}} \cdot \underbrace{\text{pairing}(g^{s^2-3s+2}, g^{-80})}_{j^{-80(s^2-3s+2)}}$$

Simplify right side:

$$j^{25} \cdot j^{-80s^2+240s-160} = j^{-80s^2+240s-135}$$

Left side:

$$j^{(20s-15)(-4s+9)} = j^{-80s^2+180s+60s-135} = j^{-80s^2+240s-135}$$

Equal! Proof is valid.

Key Insights

1. $Z(x)$ **uses public gate roots** (1, 2), **not equation roots** (5, 10).
2. **Circuit constraints** encode the equation:
   - Gate 1 computes x^2.
   - Gate 2 enforces $x^2 = 15x - 50$.
3. **Zero-knowledge**: The verifier learns g^{20s-15} (encrypted), not $x = 5$.

This proves knowledge of $x = 5$ **without revealing it**, using public gates and vanishing polynomial.