Witness Hiding POKs

**Definition:** A POK (P,V) is witness-hiding (WH) if there exists no efficient algorithm which, after interacting arbitrarily with P (possibly in many protocol instantiations), can make V accept with non-negligible probability.

For predicate Q(·,·) and value z, let \( W_z = \{ x : Q(z,x) = \text{true} \} \) be the set of witnesses for z. Consider a setting where \( |W_z| \geq 1 \).

**Definition:** A POK (P,V) is witness-independent (WI) if for any verifier V* the transcript is independent of which witness the prover is using in the proof.

**Theorem:** If one can generate a pair \( (z,x) \) with \( x \) uniform in \( W_z \) and it is computationally infeasible to find a triple \( (z,x,x') \) with \( x \neq x' \) and \( x,x' \in W_z \), then every witness-independent POK for \( Q(·,·) \) is witness-hiding.

2-Extractability

**Definition:** A three-move protocol (round) with challenge space \( C \) is 2-extractable if from any two triples \( (t,c,r) \) and \( (t',c',r') \) with \( c \neq c' \) accepted by V one can efficiently compute an \( x \) with \( Q(z,x) = \text{true} \).

**Theorem:** An interactive protocol consisting of \( s \) 2-extractable rounds with challenge space \( C \) is a proof of knowledge \( Q(·,·) \) if \( 1/(|C|^s) \) is negligible.

**Proof:** Knowledge extractor K:
1. Choose randomness for \( P' \) and execute the protocol between \( P' \) and V.
2. Execute the protocol again (same randomness for \( P' \)).
3a. If \( P' \) accepts in both executions, identify first round with different challenges \( c \) and \( c' \) (but same \( t \)). Use 2-extractability to compute an \( x \), and output it (and stop).
3b. Otherwise, go back to Step 1.

Distinguishing Advantage

**Setting:** Random variables \( X \) and \( Y \), distributions \( P_X \) and \( P_Y \).

**Distinguisher**
- Algorithm \( A \) to distinguish \( X \) from \( Y \).
- Goal: on input \( x \leftrightarrow X \), output \( X' \); on input \( y \leftrightarrow Y \), output \( Y' \).

**Advantage:** \( \Delta_A(X,Y) := |Pr_{X}[A(x) = X'] - Pr_{Y}[A(y) = X']| \)

**Asymptotics**
- Families of random variables \( \{X_n\}_{n \in \mathbb{N}} \) and \( \{Y_n\}_{n \in \mathbb{N}} \)
- \( \Delta_A(X_n,Y_n) := |Pr_{X_n}[A(x) = X'] - Pr_{Y_n}[A(y) = X']| \)

**Indistinguishability Levels**
- Perfect: \( P_X = P_Y \), i.e. \( \forall A: \Delta_A(X_n,Y_n) = 0 \)
- Statistical: \( \forall A: \Delta_A(X_n,Y_n) = \text{negligible in } n \)
- Computational: \( \forall \text{ polytime } A : \Delta_A(X_n,Y_n) = \text{negligible in } n \)

Proofs of Knowledge

Let \( Q(·,·) \) be a binary predicate and let a string \( z \) be given. Consider the problem of proving knowledge of a secret \( x \) such that \( Q(z,x) = \text{true} \).

**Definition:** A protocol \( (P,V) \) is a proof of knowledge for \( Q(·,·) \) if there exists an efficient program (knowledge extractor) \( K \), which can interact with any program \( P' \) for which \( V \) accepts with noticeable (also called non-negligible) probability, and outputs a valid secret \( x \).

\[ P \quad \leftrightarrow \quad V \]

\[ P' \quad \leftrightarrow \quad K \quad \downarrow \quad x \quad \text{extraction} \]

Note: K can rewind \( P' \) (restart with same randomness).