Decisional second-preimage resistance
When does SPR imply PRE?

Daniel J. Bernstein, Andreas Hülsing
Motivation

This work

• answers a long standing subtle question about the relation of hash function properties
• provides a tool that enables tight security proofs for hash-based signatures
Cryptographic hash functions

• Efficient function
  
  \[ h : \{0,1\}^n \times \{0,1\}^{\ell(n)} \rightarrow \{0,1\}^n \]

• We write \( h(k, x) = h_k(x) \)

• Key \( k \) in this case is public information. Think of function description.
Collision resistance

Success probability of an adversary $A$ against collision resistance (CR) of $h$ is defined as

$$\text{Succ}_{h}^{CR}(A) = \Pr[k \leftarrow_R \{0,1\}^n, (x_1, x_2) \leftarrow A(k): h_k(x_1) = h_k(x_2) \land (x_1 \neq x_2)]$$
Second-preimage resistance (SPR)

Success probability of an adversary $A$ against *second-preimage resistance (SPR) of $h$* is defined as

$$\text{Succ}_h^{\text{SPR}}(A) = \Pr[k \leftarrow_R \{0,1\}^n, x \leftarrow_R \{0,1\}^{l(n)},
\text{ } x' \leftarrow A(k, x) : h_k(x) = h_k(x') \land (x \neq x')]$$
Security properties:
Preimage resistance / One-wayness

Success probability of an adversary $A$ against preimage resistance (PRE) of $h$ is defined as

$\text{Succ}_h^{\text{PRE}}(A) = \Pr[k \leftarrow_R \{0,1\}^n, x \leftarrow_R \{0,1\}^{l(n)},$

$y \leftarrow h_k(x), x' \leftarrow A(k,y) : h_k(x') = y]$
Relations

Stronger assumption / easier to break

Assumption / Attacks

weaker assumption/ harder to break

Collision-Resistance

2nd-Preimage-Resistance

One-way
CR implies SPR?

Reduction $B^{_{ASPR}}_{_{CR}} (k)$:

1. $x \leftarrow_R \{0,1\}^{l(n)}$
2. $x' \leftarrow A_{_{SPR}}(k, x)$
3. Return $(x, x')$

$\text{Succ}_{_{h}}^{_{CR}} (B^{_{ASPR}}_{_{CR}}) = \text{Succ}_{_{h}}^{_{SPR}} (A_{_{SPR}})$
SPR implies PRE?

Reduction $B_{SPR}^{APRE}(k, x)$:

1. $y = h_k(x)$
2. $x' \leftarrow A_{PRE}(k, y)$
3. Return $x'$

Succ$h^{SPR}(B_{SPR}^{APRE}) \geq$?

$0.5 \cdot$ Succ$h^{PREF}(A_{PRE})$

Where is the problem?
Positive result

Rogaway-Shrimpton show that for $l(n)$ much bigger then $n$ we are fine
Negative result

The identity function demonstrates that SPR cannot generally imply PRE.
The gap

Functions with $l(n) \approx n$
(especially length preserving)

Exactly the ones we use
in hash-based OTS

Are we doomed?
The general case

• SHA-X ≠ identity function
• SHA-X ≈ random function
Fooling the reduction

• Reductions have to work for all $A$!

• $A(k, y)$:
  1. Compute $X = f_k^{-1}(y)$
  2. If $|X| > 1$, abort
  3. Else $X = \{x\}$, return $x$

For $f_k: \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ random,

$$\text{Succ}^{PRE}_h(A) = \frac{1}{e}$$

$$\text{Succ}^{SPR}_h(B^{APRE}_{SPR}) = 0$$
Decisional second-preimage resistance to the rescue!

- \( P_k(x) = \begin{cases} 0, \text{ if } \left| f_k^{-1}(f_k(x)) \right| = 1 \\ 1, \text{ otherwise} \end{cases} \)

- Can salvage reduction \( B_{SPR}^{APRE} \) if
  1. \( \text{SPprob}(f_k) = \Pr[P_k(x) = 1 \mid x \leftarrow_R \{0,1\}^n] \) is non-negligible, and
  2. it is hard to reliably determine \( P_k(x) \)

We show that \( \text{SPprob}(f_k) \approx \left(1 - \frac{1}{e}\right) \) for the overwhelming majority of all functions. E.g., for a random 256bit hash \( f_k \)

\[ \Pr[\text{SPprob}(f_k) < 0.6] \approx 2^{-2^{239}} \]
DSPR = “It is hard to reliably determine $P_k(x)$”

$$\text{Adv}_{f}^{DSPR}(A) = \max\{0, \Pr[k \leftarrow_R \{0,1\}^n, x \leftarrow_R \{0,1\}^n, b \leftarrow A(k, x): P_k(x) = b] - \text{SPprob}(f_k)\}$$
Some intuition about DSPR

• \([\text{SPprob}(f_k) \approx 1] \Rightarrow [\text{Adv}_{f}^{DSPR}(A) \approx 0]\)

• If \(f_k\) is strongly compressing, it is information-theoretically DSPR.

\[
\text{Adv}_{f}^{DSPR}(A) = \max\{0, \Pr[k \leftarrow_R \{0,1\}^n, x \leftarrow_R \{0,1\}^n, b \leftarrow A(k,x): P_k(x) = b] - \text{SPprob}(f_k)\}
\]
Some intuition about DSPR

• Given SPR adversary $A_{SPR}$ define:

• $B_{DSPR}^{A_{SPR}}(k, x)$:
  1. If $A_{SPR}(k, x)$ succeeds, return 1
  2. Return 0.

• Requires $\text{Succ}_{f}^{SPR}(A_{SPR}) > 2 \cdot \text{SPprob}(f_k) - 1$ for non-zero advantage!

$$\text{Adv}_{f_{DSPR}}(A) = \max\{0, \Pr[k \leftarrow_R \{0,1\}^n, x \leftarrow_R \{0,1\}^n, b \leftarrow A(k, x): P_k(x) = b] - \text{SPprob}(f_k)\}$$
DSPR at work

Reduction $B_{SPR}^{APRE}(k,x)$:
1. $y = h_k(x)$
2. $x' \leftarrow A_{PRE}(k,y)$
3. Return $x'$

Reduction $C_{DSPR}^{APRE}(k,x)$:
1. $y = h_k(x)$
2. $x' \leftarrow A_{PRE}(k,y)$
3. Return 1 if $x' \neq x$
4. Return 0

We show

$Succ_f^{PRE}(A_{PRE}) \leq Adv_f^{DSPR}(C_{DSPR}^{APRE}) + 3 \cdot Succ_f^{SPR}(B_{SPR}^{APRE})$
Application to hash-based signatures

• Interactive Game T-OpenPRE:

1. Generate T pairs $(k_i, y_i) = (k_i, f_k(x_i))$, $x_i \leftarrow_R \{0,1\}^n$

2. Give pairs to $A$ and allow $A$ to ask for up to $T - 1$ of the $x_i$

3. Output 1 if $(j, x) \leftarrow A()$ is a preimage $f_{k_j}(x) = y_j$ for “unopened” image $y_j$

Variants of this naturally arise in security proof of WOTS, and L-OTS
Pre $\Rightarrow$ T-OpenPRE is non-tight!

Given $A_{T-\text{OpenPRE}}$
build $B(k, y)$:
1. Play T-OpenPRE game but replace random pair $(k_c, y_c)$ by challenge $(k, y)$
2. If $A$ asks to open position $c$, abort
3. If $A$ returns $(i, x)$, output $x$

Reduction loss of $1/T!$
Multi-target DSPR

Definition 31 (T-DSPR). Let $T$ be a positive integer. Let $\mathcal{A}$ be an algorithm with output in $\{1, \ldots, T\} \times \{0, 1\}$. The advantage of $\mathcal{A}$ against the $T$-target decisional second-preimage resistance of a keyed hash function $H$ is

$$\text{Adv}_{H}^{T-\text{DSPR}}(\mathcal{A}) \overset{\text{def}}{=} \max\{0, q - p\}$$

where

$$q = \Pr\left[(x_1, k_1, \ldots, x_T, k_T) \leftarrow_R (\mathcal{X} \times \mathcal{K})^T; (j, b) \leftarrow A(x_1, k_1, \ldots, x_T, k_T) : P_{k_j}(x_j) = b\right];$$

$$p = \Pr\left[(x_1, k_1, \ldots, x_T, k_T) \leftarrow_R (\mathcal{X} \times \mathcal{K})^T; (j, b) \leftarrow A(x_1, k_1, \ldots, x_T, k_T) : P_{k_j}(x_j) = 1\right];$$

and $P_{k_j} = \text{SPexists}(H_{k_j})$. 
T-DSPR + T-SPR ⇒ T-OpenPRE, tightly!

• Use T-target versions of $B_{SPR}^{APRE}$, and $C_{DSPR}^{APRE}$

• Can replace all pairs by challenges
  • We do know a preimage

• If $A_{T-OpenPRE}$ always returns known image, $C_{T-DSPR}^{AT-OpenPRE}$ will have advantage in breaking T-DSPR

• Else, $B_{T-SPR}^{AT-OpenPRE}$ succeeds with high probability
More in paper

• DSPR is quantum-hard for random functions
• Detailed analysis of SPprob
• Full proofs

• See “The SPHINCS+ Signature Framework“ (CCS‘19) for application to SPHINCS+ and other hash-based signatures.
Questions?

Paper(s) available at
https://sphincs.org/resources.html