

## On the Security of Short Schnorr Signatures

Jeremiah M Blocki

Seunghoon Lee

Department of Computer Science, Purdue University

### Motivation / Contribution

- Schnorr Signatures:  $4k$ -bits long (short) with  $k$ -bit security
  - ✓ One hash value ( $2k$ -bits) + One group element ( $2k$ -bits)
  - ✓ BLS Signatures are shorter ( $2k$ -bits), but less practical
- Folklore:  $3k$ -bit signatures with shorter hash function ( $k$ -bits)
  - ✓ No security proof
- Our Result: Folklore is right!
  - ✓ Concrete security proof in Generic Group Model + Random Oracle Model shows
  - ✓  $3k$ -bit signatures with  $k$ -bits of security

### Generic Group Model

- For a cyclic group  $G = \langle g \rangle$  of order  $q$ , elements of  $G$  are encoded by bit strings of length  $\ell$  in a cryptographic scheme. Let  $\mathbb{G}$  be a set of bit strings of length  $\ell$ , then  $\tau: G \rightarrow \mathbb{G}$  gives the natural representation of  $G$  in  $\mathbb{G}$ .
- The key idea is that an adversary attacks a primitive is only given access to a randomly chosen encoding of a group instead of efficient encodings.
- On input  $(a, b) \in \mathbb{G} \times \mathbb{G}$ , the  $\text{Mult}(\cdot, \cdot)$ ,  $\text{Inv}(\cdot)$  and  $\text{Pow}(\cdot, \cdot)$  oracles return
 
$$\text{Mult}(a, b) = \tau(\tau^{-1}(a) \cdot \tau^{-1}(b))$$

$$\text{Inv}(a) = \tau((\tau^{-1}(a))^{-1})$$

$$\text{Pow}(a, b) = \tau((\tau^{-1}(a))^b)$$
 if  $a, b \in \tau(G)$ .

### $k$ -bits of Security

- We say that a scheme yields “ $k$ -bits of security” if any attacker running in time at most  $t$  should forge a signature with probability at most  $t/2^k$  and this should hold for all  $t \leq 2^k$ .

### Open Questions

- Could one achieve the same concrete security bound for ECDSA/DSA in the generic group and random oracle model?
- Are we able to identify any concrete statements that have been proved about BLS signatures in the generic group and random oracle model?

### References

- Schnorr (1989). Efficient Identification and Signatures for Smart Cards. *CRYPTO '89*.
- Neven, G., Smart, N. & Warinschi, B. (2009). Hash function requirements for Schnorr signatures. *Journal of Mathematical Cryptology*, 3(1), pp. 69-87.
- Seurin Y. (2012). On the Exact Security of Schnorr-Type Signatures in the Random Oracle Model. *EUROCRYPT 2012*.
- Boneh, D., Lynn, B., & Shacham, H. (2004). Short Signatures from the Weil Pairing. *Journal of Cryptology*. 17 (4): 297-319.

### The Schnorr Signature Scheme<sup>[1]</sup>

<b>Kg:</b> $sk \xleftarrow{\$} \mathbb{Z}_q; pk \leftarrow \tau(g^{sk})$ Return $(pk, sk)$	<b>Sign(<math>sk, m</math>):</b> $r \xleftarrow{\$} \mathbb{Z}_q; I \leftarrow \tau(g^r)$ $e \leftarrow H(I  m)$ (first $k$ -bits) $s \leftarrow r + sk \cdot e \pmod q$ Return $\sigma = (s, e)$	<b>Vfy(<math>pk, m, \sigma</math>):</b> $I \leftarrow \text{Mult}(\text{Pow}(\tau(g), s), \text{Pow}(\text{Inv}(pk), e))$ If $H(I  m) = e$ <span style="color: red;">= <math>\tau(g^s \cdot g^{-sk \cdot e})</math></span> Then return 1 Else return 0.
--	---	---

Figure 1. The Schnorr Signature Scheme

- Our Analysis:  $H$  is a random oracle that outputs  $k$  bits (can truncate output if needed)
- Typical: Hashes are  $2k$  bits long ( $4k$ -bit signatures)

### Our Results

- We have the following (informal) form of theorem which guarantees a  $3k$ -bit signature with  $k$ -bits of security:

**Theorem.** Let  $\mathcal{A}$  be an adversary attacking Schnorr signature scheme running in time at most  $t$ . Then the probability that the adversary successfully forge a signature is bounded by

$$\text{Adv}(\mathcal{A}) \leq O\left(\sqrt{\frac{t}{q}} + \frac{t}{2^k} + \frac{t^2}{q}\right)$$

under the Generic Group Model of order  $q$  and Random Oracle Model.

- Set  $q = 2^{2k}$  and select a hash function  $H$  with  $k$  output bits. The resulting signatures have  $k$ -bits of security and length  $k + \log_2 q = k + 2k = 3k$ .

### Security Reduction

- Security reduction starts with the attacker  $\mathcal{A}_{sig}$  that attacks the modified Schnorr signature and builds the discrete-log attacker  $\mathcal{A}_{dlog}$ .

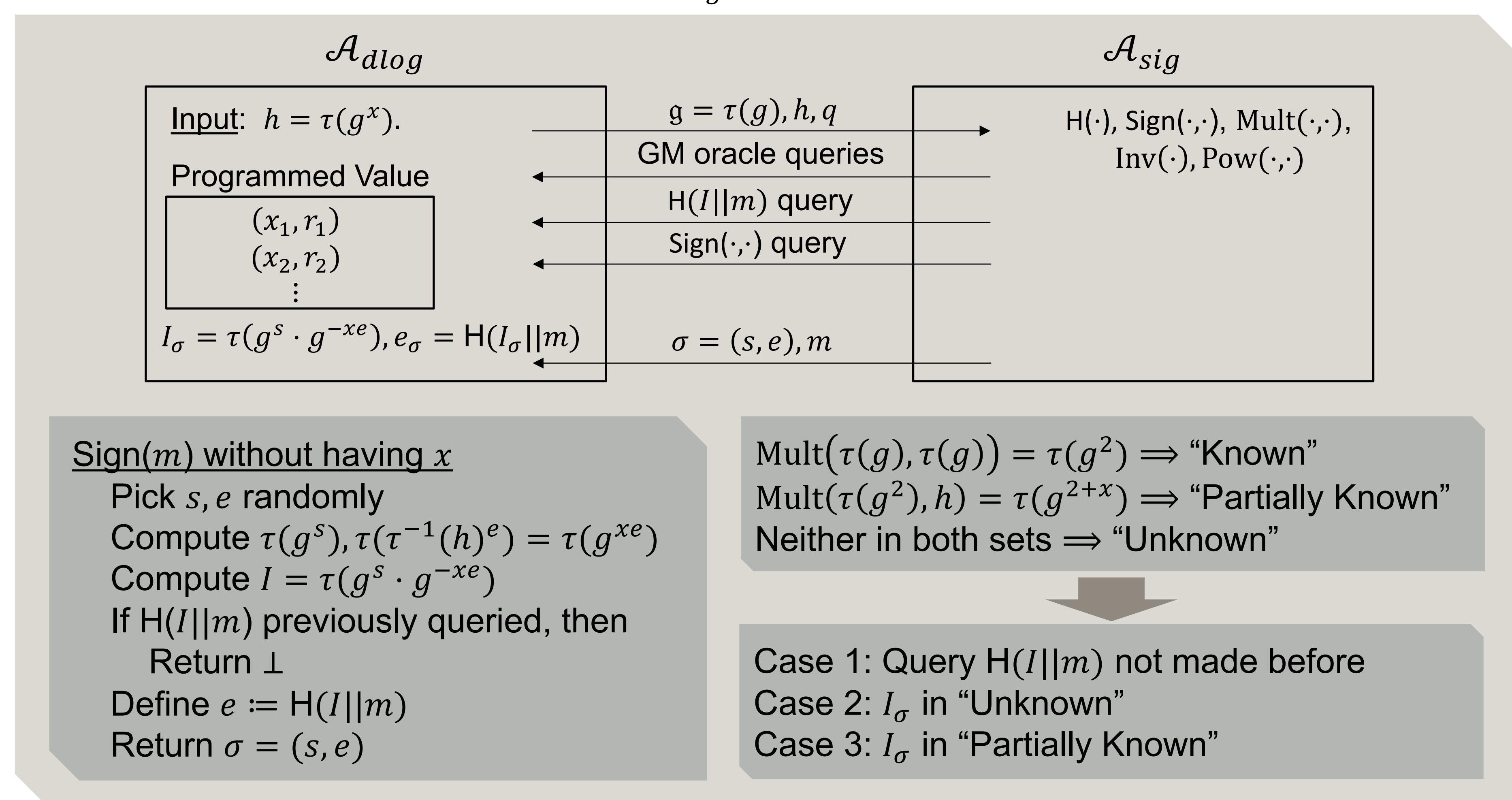


Figure 2. A Security Reduction