Towards A Standard for Practical Hash-based Signatures

D. Butin, S.-L. Gazdag, A. Hülsing

Technische Universiteit
Eindhoven
University of Technology

Where innovation starts

Hash-based Signature Schemes [Mer89]

- Post quantum
- Only secure hash function
- Security well understood
- Fast

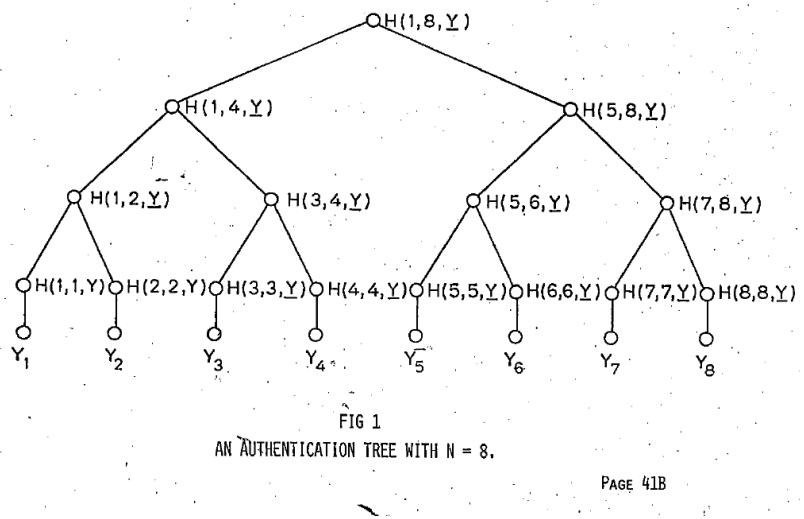
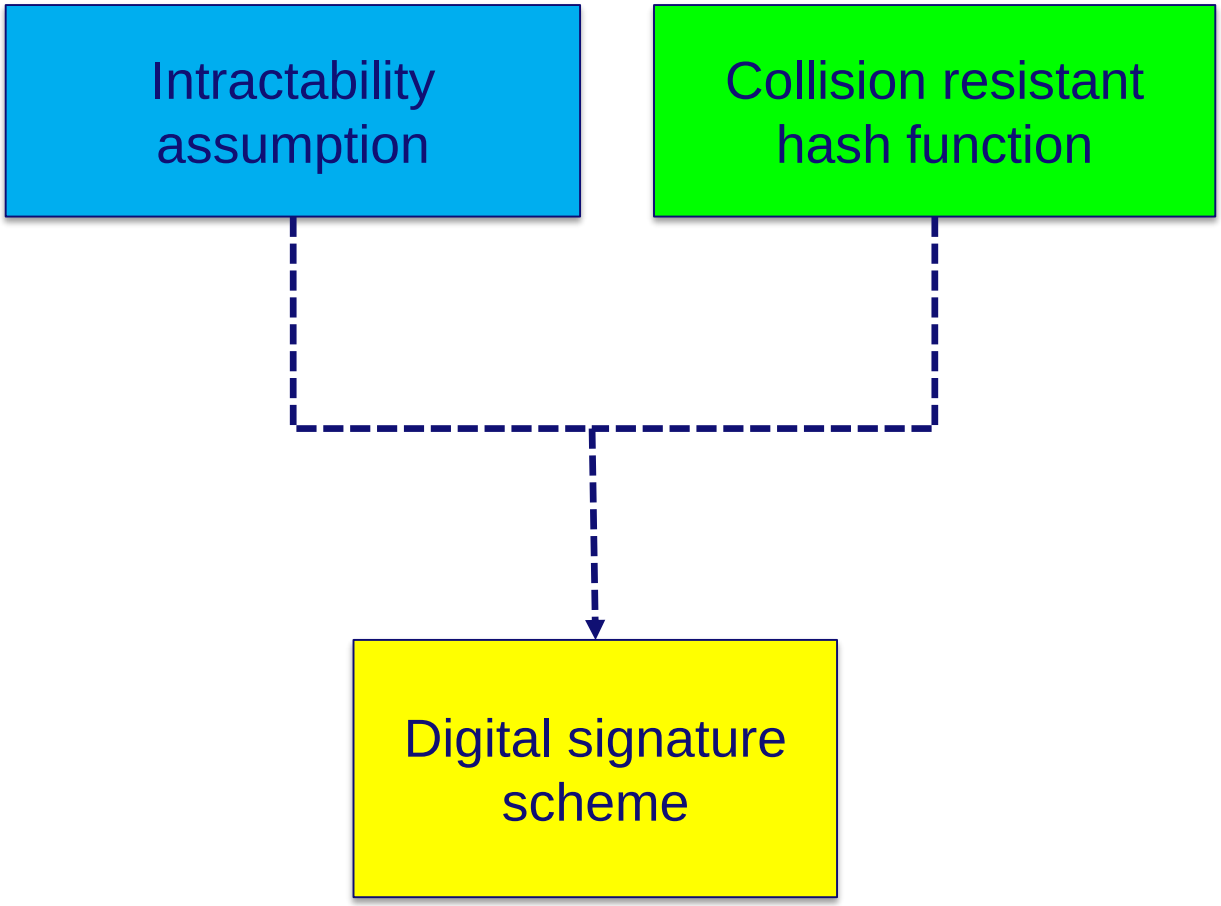


FIG 1
AN AUTHENTICATION TREE WITH N = 8.

PAGE 41B

Security



Post-Quantum Security

n-bit hash function

Grover‘96:

Preimage finding $O(2^n) \rightarrow O(2^{\frac{n}{2}})$

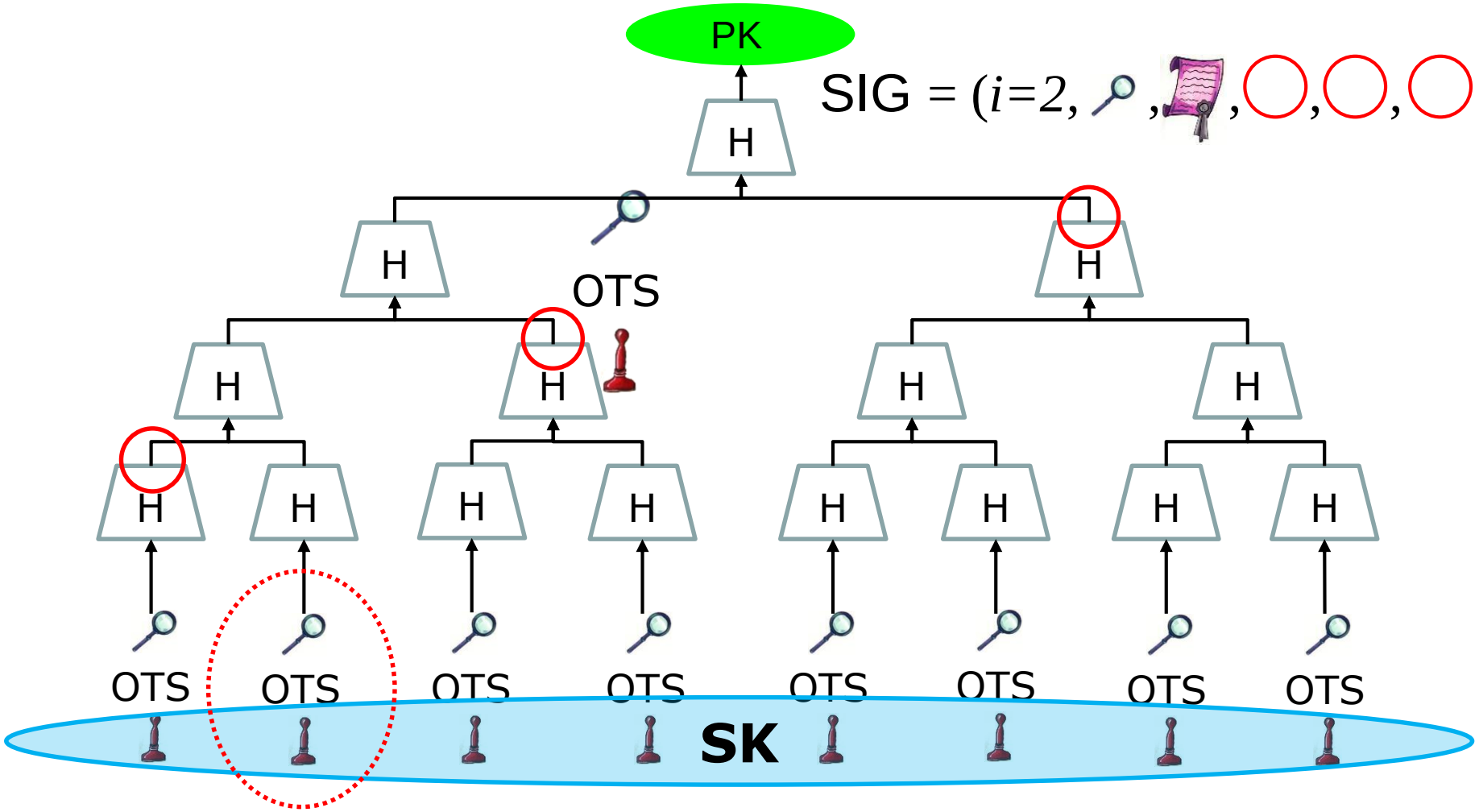
Brassard et al. 1998:

Collision finding $O(2^{\frac{n}{2}}) \rightarrow O(2^{\frac{n}{3}})$

Aaronson & Shi’04:

Quantum collision finding $2^{\frac{n}{3}}$ is lower bound

Merkle's Hash-based Signatures



Practical Challenge: Handle State

- **Can be avoided in theory, paid with efficiency**
- **Different API**
  - **Handle Integration**
- **Prevent copies**
  - **No key back-up**
- **Multi-threading safety**
- **Industry input appreciated**

McGrew & Curcio‘2014

Crypto Forum Research Group — D. McGrew
Internet-Draft — M. Curcio
Intended status: Informational — Cisco Systems
Expires: January 5, 2015 — July 4, 2014

Hash-Based Signatures
draft-mcgrew-hash-sigs-02

Abstract

This note describes a digital signature system based on cryptographic hash functions, following the seminal work in this area. It specifies a one-time signature scheme based on the work of Lamport, Diffie, Winternitz, and Merkle (LDWM), and a general signature scheme, Merkle Tree Signatures (MTS). These systems provide asymmetric authentication without using large integer mathematics and can achieve a high security level. They are suitable for compact implementations, are relatively simple to implement, and naturally resist side-channel attacks. Unlike most other signature systems, hash-based signatures would still be secure even if it proves feasible for an attacker to build a quantum computer.

McGrew & Curcio‘2014

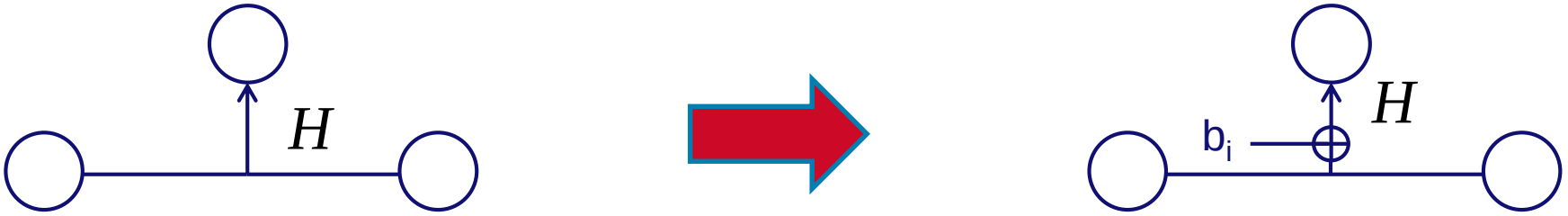
- **Merkle Tree + Winternitz OTS**
- **Parameter Sets = Cipher Suites**
- **Security = collision resistance**

XMSS

eXtended Merkle Signature Scheme

Reduced Security Requirements

- **Change WOTS -> WOTS+**
- **Change Tree**



Security from second-preimage resistance

„Collision-resilient“ scheme

No birthday-attacks

Size reduction

Hash function $h:\{0,1\}^* \rightarrow \{0,1\}^m$

Assume:

- **only generic attacks,**
- **security level n**

Collisio

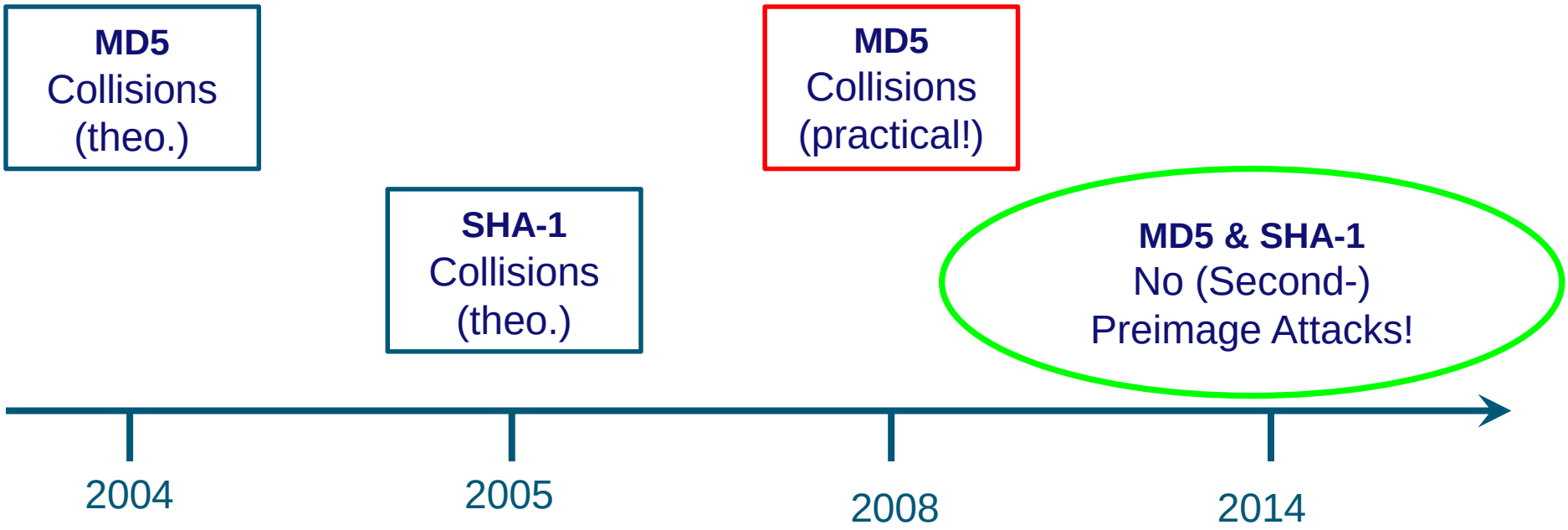
Halfes Signature Size!

⟶ g[illegible] = birthday attack ⟶ m = $2n$

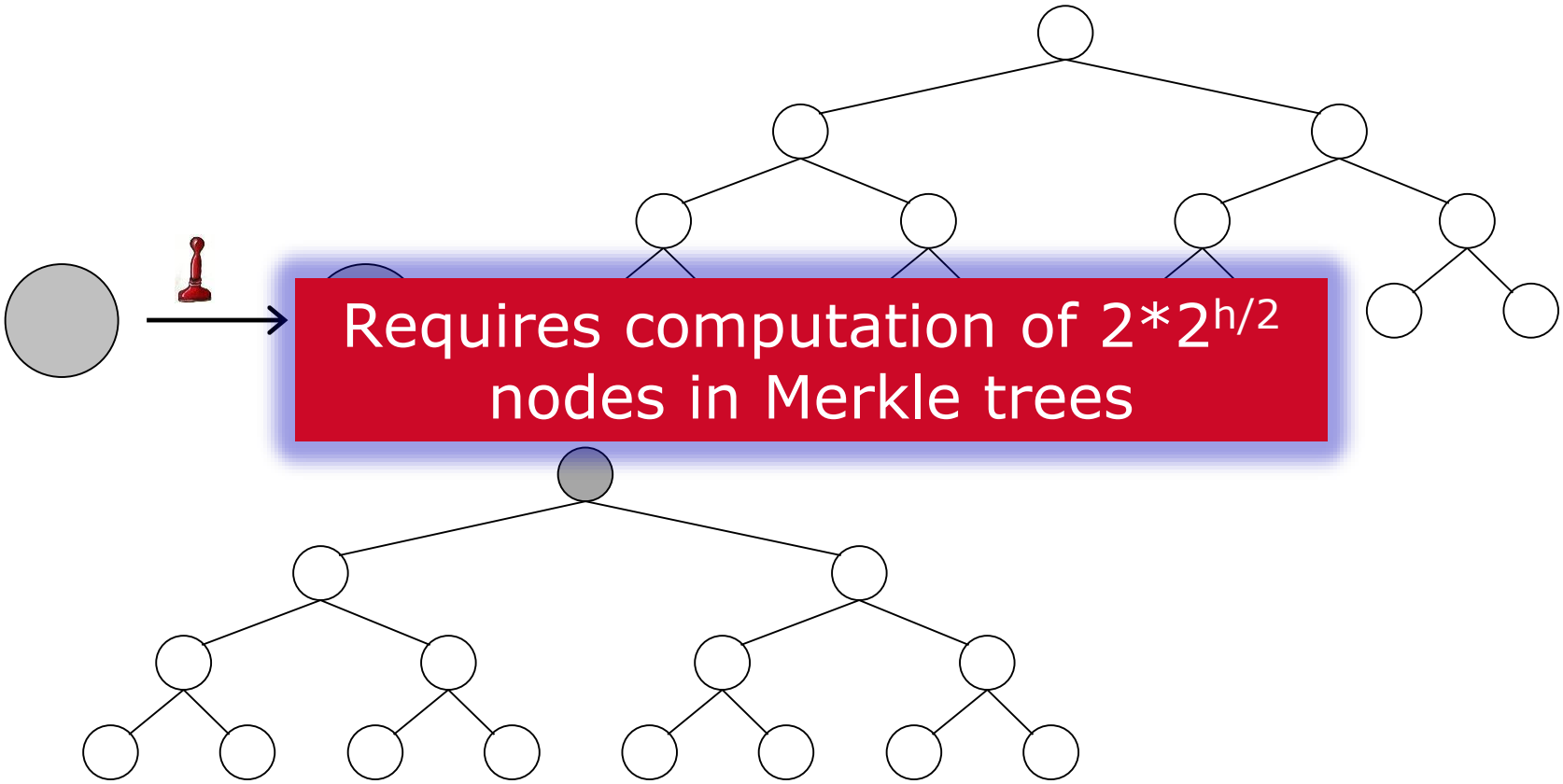
Second-preimage resistance required:

⟶ generic attack = exhaustive search ⟶ $m = n$

Early warning system



Tree Chaining



Tree Chaining

- **Can be extended to d layers**
- **Reduces signature and key generation time**
- **Necessary for smartcards & h >> 20**

Tree Chaining

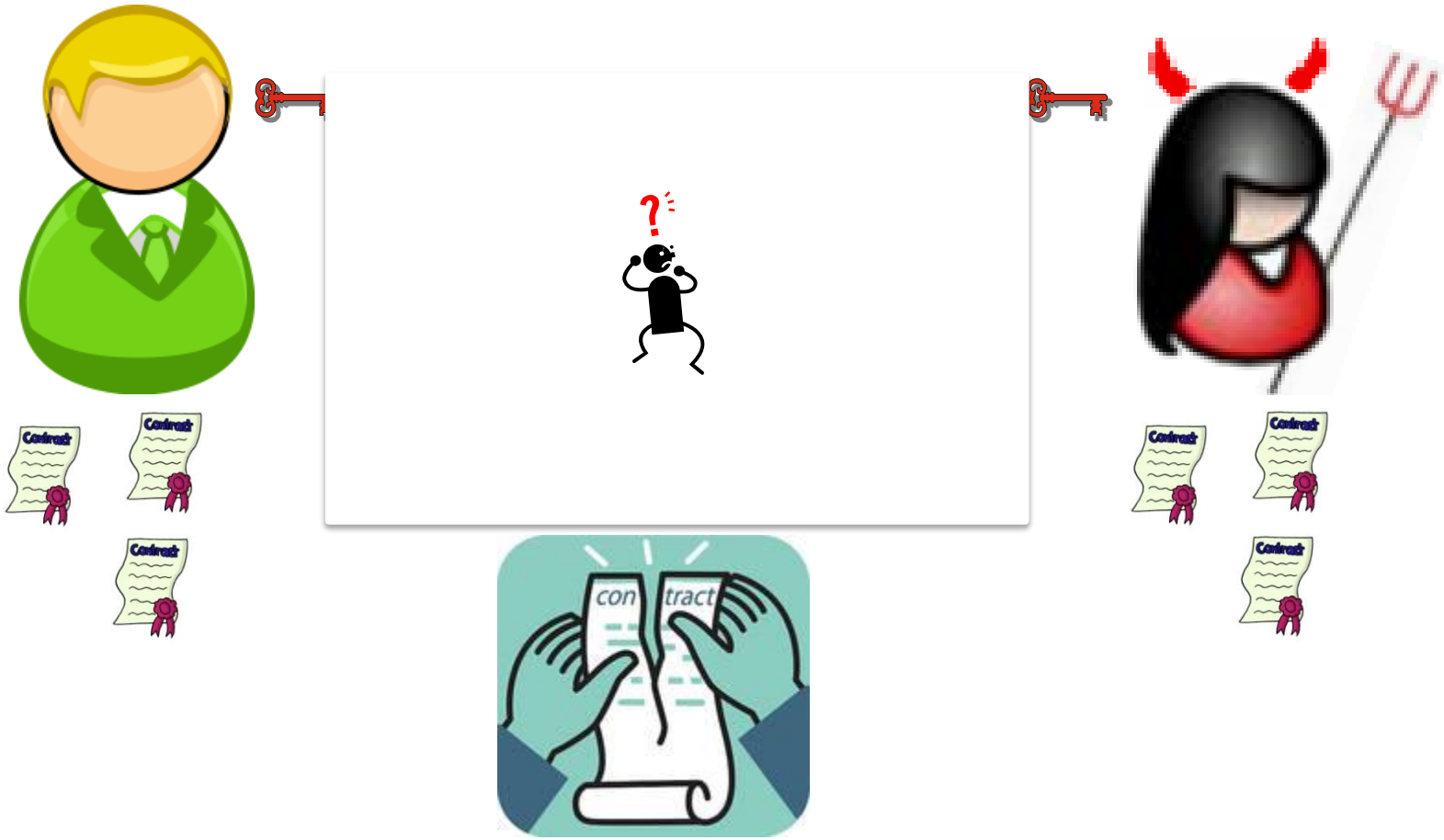
	Sign (ms)	Verify (ms)	Keygen (ms)	Signature (byte)	Public Key (byte)	Secret Key (byte)	Bit Sec.	Comment
XMSS	**134**	23	**925,400**	2,388	800	2,448	92	H = 16, w = 4
XMSS$^+$	**106**	25	**5,600**	3,476	544	3,760	94	H = 16, w = 4
RSA 2048	190	7	11,000	≤ 256	≤ 512	≤ 512	87	

Infineon SLE78 16Bit-CPU@33MHz, 8KB RAM, TRNG, sym. & asym. co-processor

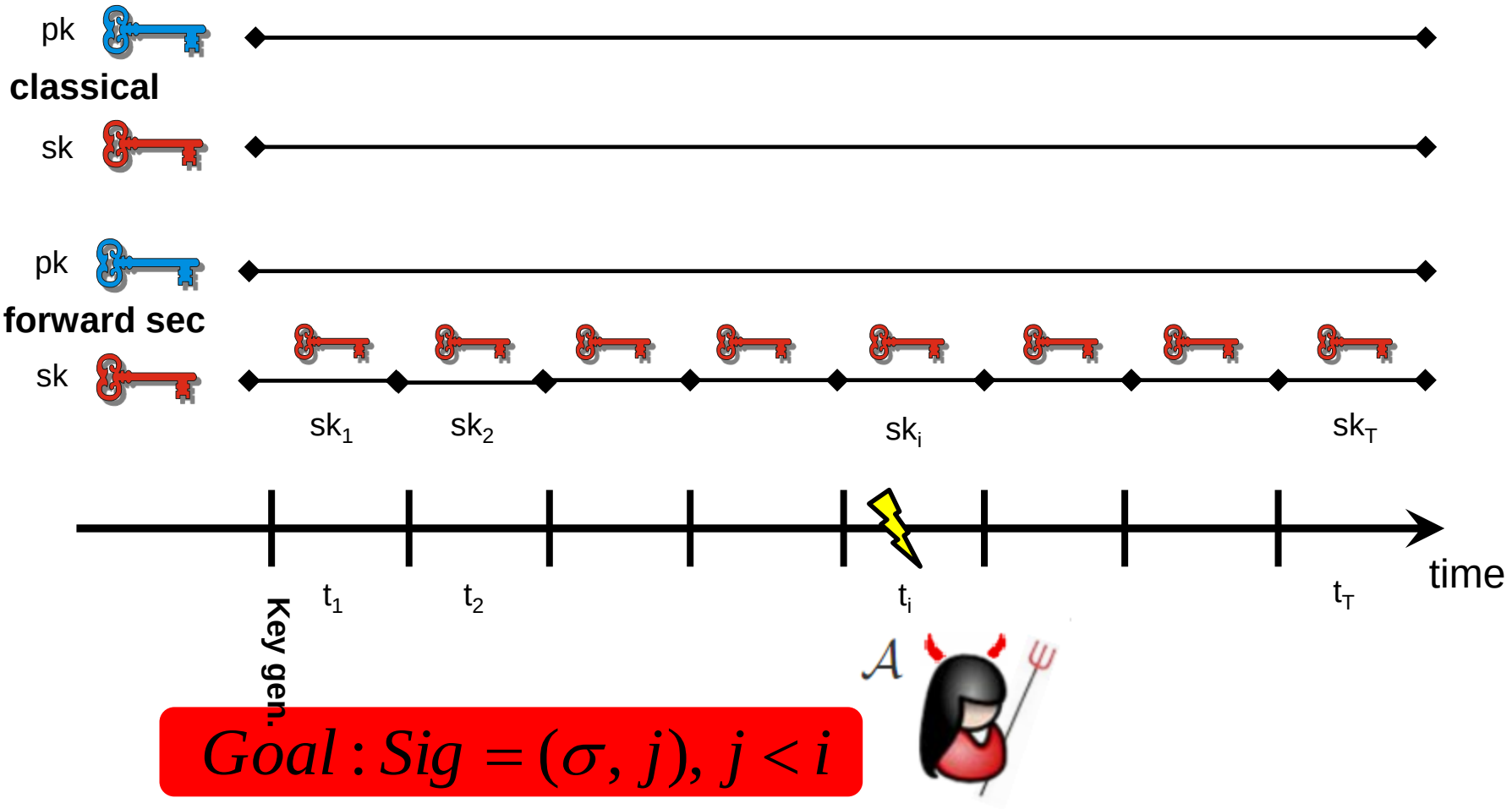
NVM: Card 16.5 million write cycles/ sector,
XMSS$^+$ < 5 million write cycles (h=20)

[HBB12]

Forward Security

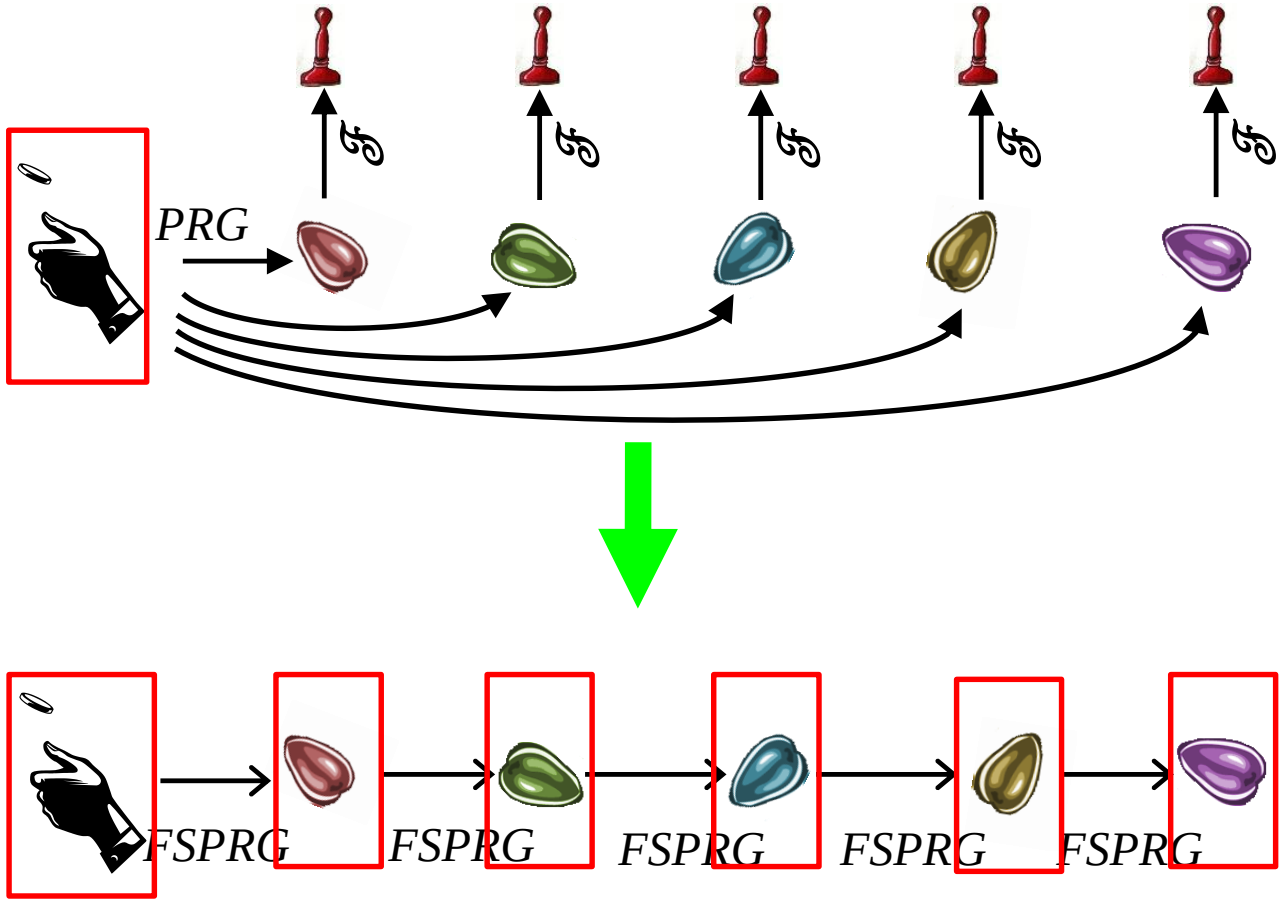


Forward Security



$$Goal : Sig = (\sigma, j), j < i$$

Requires special KeyGen



PoC Implementation

C Implementation, using OpenSSL [BDH2011]

	Sign (ms)	Verify (ms)	Signature (bit)	Public Key (bit)	Secret Key (byte)	Bit Security	Comment
XMSS-SHA-2	35.60	1.98	**16,672**	13,600	3,364	157	h = 20, w = 64,
XMSS-AES-NI	**0.52**	**0.07**	19,616	7,328	1,684	84	h = 20, w = 4
XMSS-AES	1.06	0.11	19,616	7,328	1,684	84	h = 20, w = 4
RSA 2048	**3.08**	**0.09**	≤ 2,048	≤ 4,096	≤ 512	87	

Intel(R) Core(TM) i5-2520M CPU @ 2.50GHz with Intel AES-NI

Conclusion

- **Current draft: Great first step**

... BUT ...

- **XMSS: Additional important features**
  - **More efficient**
  - **Stronger Security Guarantees**
  - **Forward-security**

Add-on to draft required.

Thank you!

Questions?