Post-Quantum Cryptography & Privacy

Andreas Hülsing

PQCRYPTO
ICT-645622

Privacy?


... the Panopticon must not be understood as a dream building: it is the diagram of a mechanism of power reduced to its ideal form.
Michel Foucault, *Discipline and Punish*, 1977

How to achieve privacy?

Tor

DuckDuckGo

Under the hood...

Asymmetric Crypto

- ECC
- RSA
- DSA

Symmetric Crypto

- AES
- SHA2
- SHA1
- ...

Combination of both needed!

We need symmetric and asymmetric crypto to achieve privacy!

Quantum Computing

Quantum Computing

“Quantum computing studies theoretical computation systems (quantum computers) that make direct use of quantum-mechanical phenomena, such as superposition and entanglement, to perform operations on data.”

-- Wikipedia

Qubits

- Qubit state: $\alpha_0 |0\rangle + \alpha_1 |1\rangle$ with $\alpha_i \in \mathbb{C}$ such that $|\alpha_0|^2 + |\alpha_1|^2 = 1$
- Ket: $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
- Qubit can be in state $\frac{|0\rangle + |1\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

- Computing with 0 and 1 at the same time!

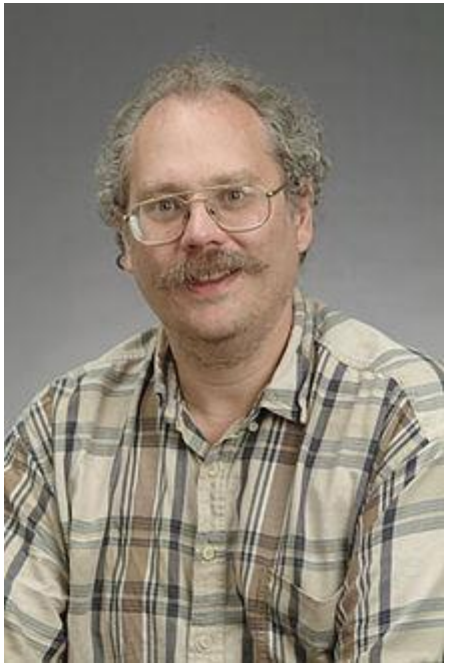
Quantum computers are not almighty

- To learn outcome one has to measure.
  - Collapses state
  - 1 qubit leads 1 classical bit of information
  - Randomized process
- Only invertible computation.
- Impossible to clone (copy) quantum state.

The Quantum Threat

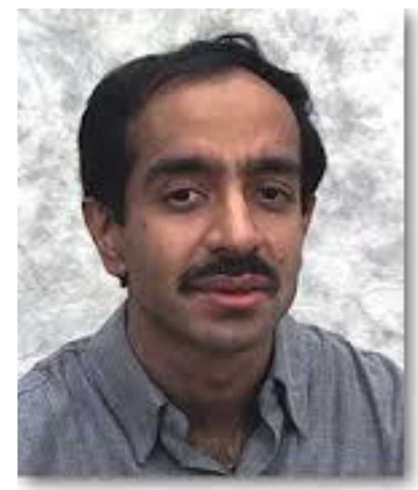
Shor's algorithm (1994)

- Quantum computers can do FFT very efficiently
- Can be used to find period of a function
- This can be exploited to factor efficiently (RSA)
- Shor also shows how to solve discrete log efficiently (DSA, DH, ECDSA, ECDH)



Grover‘s algorithm (1996)

- Quantum computers can search N entry DB in $\Theta(\sqrt{N})$
- Application to symmetric crypto
- Nice: Grover is provably optimal (For random function)
- Double security parameter.



To sum up

- All asymmetric crypto is broken by QC
  - No more digital signatures
  - No more public key encryption
  - No more key exchange

- Symmetric crypto survives
  (with doubled key size / output length)
  - NOT ENOUGH!

Why care today?

Quantum Computing

"Quantum computing studies theoretical computation systems (quantum computers) that make direct use of quantum-mechanical phenomena, such as superposition and entanglement, to perform operations on data."

-- Wikipedia

Bad news

I will not tell you when a quantum computer will be built!

Europe plans giant billion-euro quantum technologies project

Third European Union flagship will be similar in size and ambition to graphene and human brain initiatives.

Elizabeth Gibney

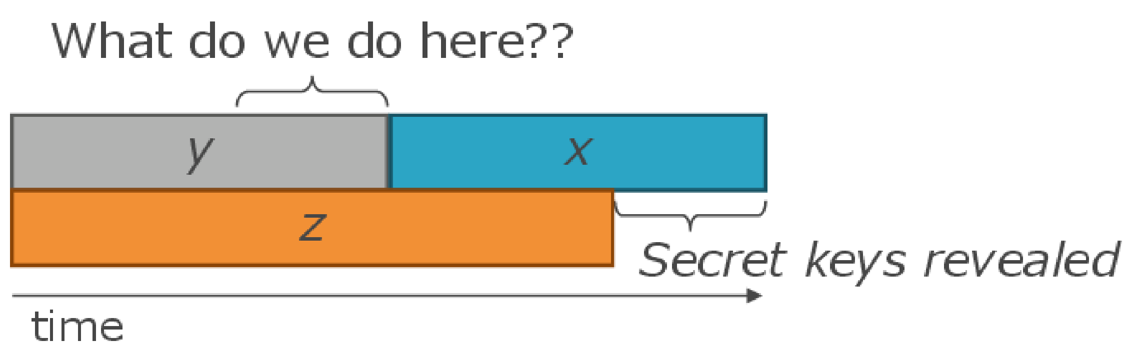
It‘s a question of risk assessment

How soon do we need to worry?

Depends on:

- How long do you need your keys to be secure? (x years)
- How much time will it take to re-tool the existing infrastructure with large-scale quantum-safe solution? (y years)
- How long will it take for a large-scale quantum computer to be built (or for any other relevant advance? (z years)

Theorem 1: If $x + y > z$, then worry.



Who would store all encrypted data traffic? That must be expensive!

Defending Our Nation. Securing The Citizens.

Quantum Cryptography

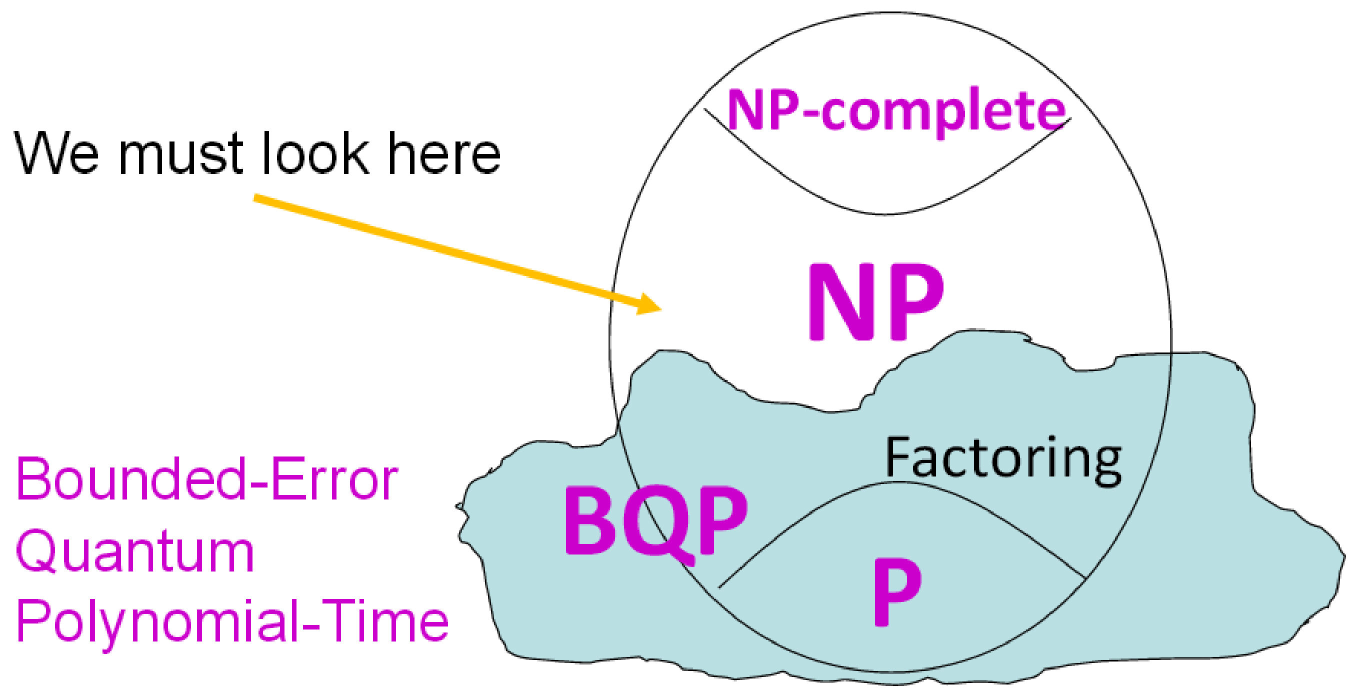
Why not beat ‘em with their own weapons?

- QKD: Quantum Key distribution.
  - Based on some nice quantum properties: entanglement & collapsing measurments
  - Information theoretic security (at least in theory) -> Great!
  - For sale today!
- So why don‘t we use this?
- Only short distance, point-to-point connections!
  - Internet? No way!
- Longer distances require „trusted-repeaters“ ☺
  - We all know where this leads...

PQCRYPTO to the rescue

Quantum-secure problems

No provably quantum resistant problems



Credits: Buchmann, Bindel 2015

Conjectured quantum-secure problems

- Solving multivariate quadratic equations (MQ-problem)
  -> Multivariate Crypto
- Bounded-distance decoding (BDD)
  -> Code-based crypto
- Short(est) and close(st) vector problem (SVP, CVP)
  -> Lattice-based crypto
- Breaking security of symmetric primitives (SHAx-, AES-, Keccak-,... problem)
  -> Hash-based signatures / symmetric crypto

Multivariate Crypto

$$4x + x^2 + y^2 z \equiv 1 \mod 13$$

$$7y^2 + 2xz^2 \equiv 12 \mod 13$$

$$x + y^2 + 12xz^2 \equiv 4 \mod 13$$

Solution: $x = 15,\ y = 29,\ z = 45$

Credits: Buchmann, Bindel 2015

MQ-Problem

Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ and $\mathbf{MQ}(n, m, \mathbb{F}_q)$ denote the family of vectorial functions $\boldsymbol{F}\colon \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^m$ of degree 2 over $\mathbb{F}_q$:

$\mathbf{MQ}(n, m, \mathbb{F}_q)$

$$= \left\{ \boldsymbol{F}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x}) | f_s(\boldsymbol{x}) = \sum_{i,j} a_{i,j} x_i x_j + \sum_i b_i x_i \,, \qquad s \in [1, m] \right\}$$

The $\mathbf{MQ}$ Problem $\mathbf{MQ}(\boldsymbol{F}, \boldsymbol{v})$ is defined as given $\boldsymbol{v} \in \mathbb{F}_q^m$ find, if any, $\boldsymbol{s} \in \mathbb{F}_q^n$ such that $\boldsymbol{F}(\boldsymbol{s}) = \boldsymbol{v}$.

Decisional version is NP-complete [Garey, Johnson´79]

Multivariate Signatures

$P: F^n \rightarrow F^m$, easily invertible non-linear

$S: F^n \rightarrow F^n$, $T: F^m \rightarrow F^m$, affine linear

Public key: $G = S \circ P \circ T$, hard to invert

Secret Key: S, P, T allows to find G^{-1}

$$G^{-1} = T^{-1} \circ P^{-1} \circ S^{-1}$$

Signing: $s = T^{-1} \circ P^{-1} \circ S^{-1}(m)$

Verifying: $G(s) =^? m$

Forging signature: Solve $G(s) - m = 0$

Fast

Large keys:
100 kBit for 100 bit security
Compared to 1776 bit RSA modulus

- UOV , Goubin et al., 1999
- Rainbow, Ding, et al. 2005
- pFlash, Cheng, 2007
- Gui, Ding, Petzoldt, 2015

Credits: Buchmann, Bindel 2015

Multivariate Cryptography

- Breaking scheme $\not\Leftrightarrow$ Solving MQ-Problem
  -> Not a random instance
  -> Not NP-hard (there might be easy instances)
  -> New proposal with security reduction, small keys, but large signatures.
- Many broken proposals
  -> Oil-and-Vinegar, SFLASH, MQQ-Sig, (Enhanced) TTS, Enhanced STS.
  -> Security somewhat unclear
- Only signatures
  -> (new proposal for encryption exists but too recent)
- Really large keys

Coding-based cryptography - BDD

Given:
- Linear code $\mathbb{C} \subseteq \mathbb{F}_2^n$
- $y \in \mathbb{F}_2^n$
- $t \in \mathbb{N}$

Find:
- $x \in \mathbb{C}$: $\operatorname{dist}(x, y) \leq t$

BDD is NP-complete (Berlekamp et al. 1978) (Decisional version)

Credits: Buchmann, Bindel 2015

McEliece PKE (1978)

S, G, P matrices over F

G generator matrix for Goppa code ← Allows to solve BDD

Public key: $G' = S \circ G \circ P$, t

Secret Key: P, S, G

Encryption: $c = mG' + z \in F^n$

Decryption: $x = cP^{-1} = mSG + zP^{-1}$

solve BDD to get $y = mSG$

decode to obtain m

Fast

Large public keys!
500 kBits for 100 bit security
Compared to 1776 bit RSA modulus

IND-CPA secure version

Credits: Buchmann, Bindel 2015

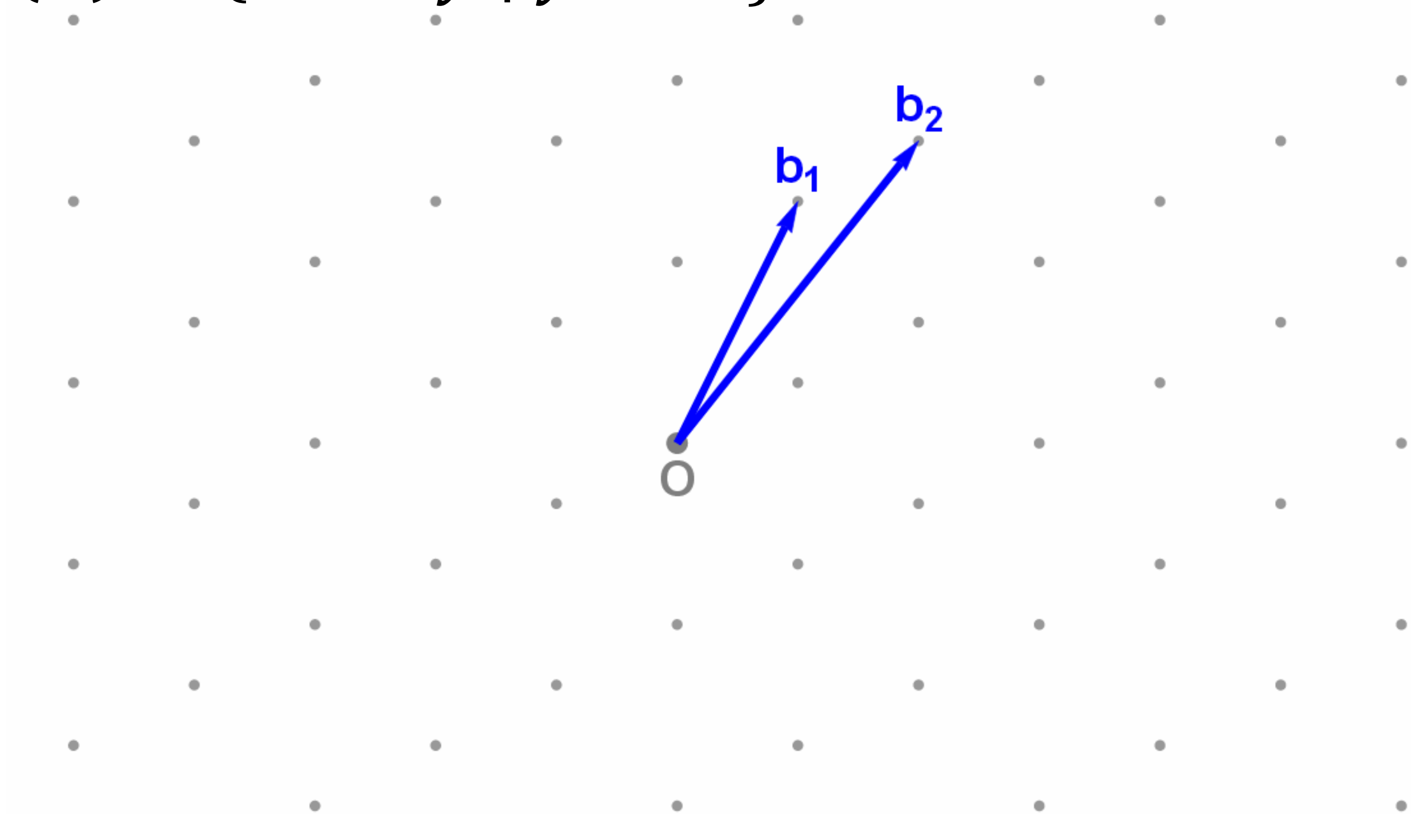
Code-based cryptography

- Breaking scheme $\not\Leftrightarrow$ Solving BDD
  -> Not a random instance
  -> Not NP-hard (there might be easy instances)
- However, McEliece with binary Goppa codes survived for almost 40 years (similar situation as for e.g. AES)
- Using more compact codes often leads to break
- So far, no practical signature scheme
- Really large public keys

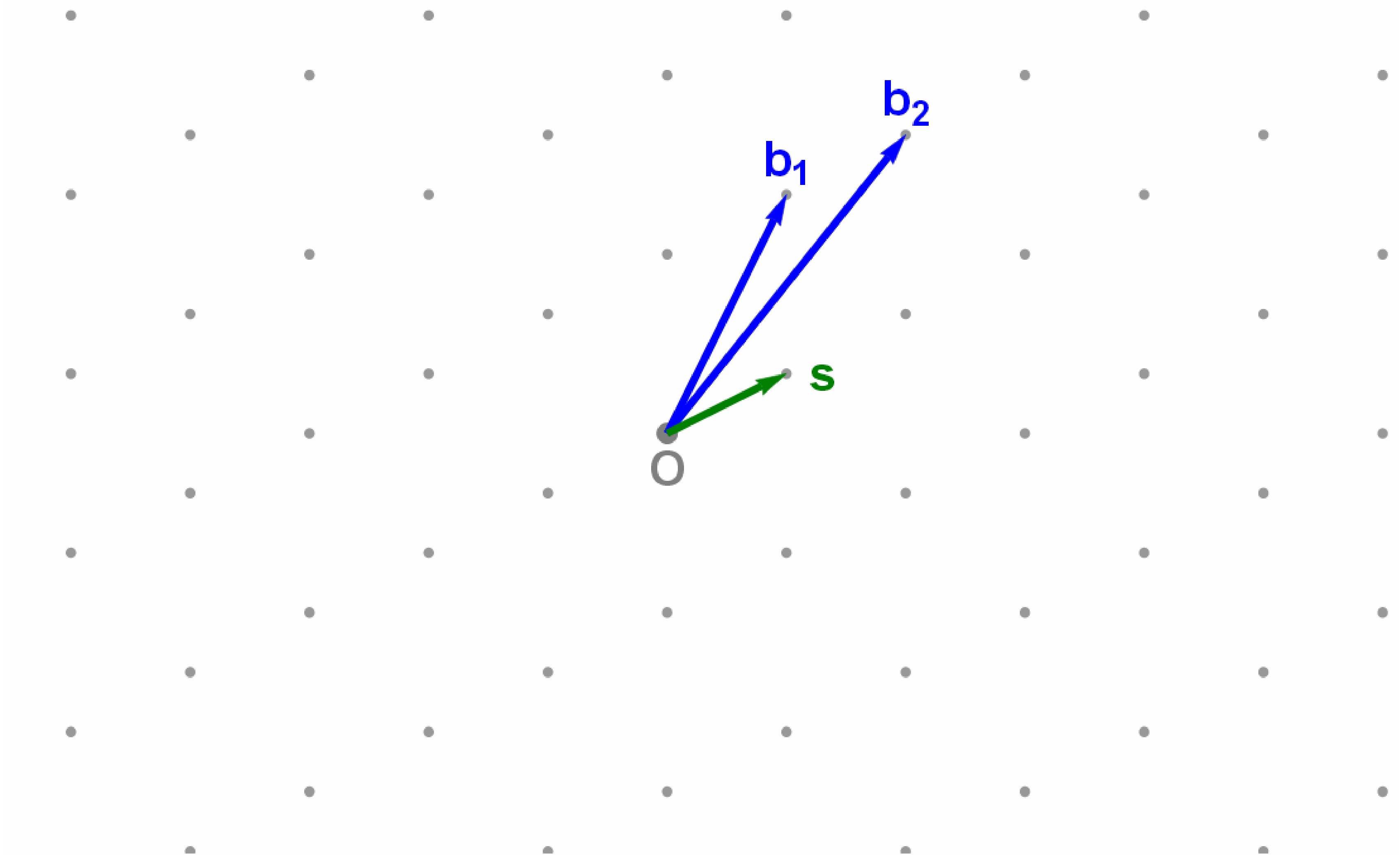
Lattice-based cryptography

Basis: $B = (b_1, b_2) \in \mathbb{Z}^{2\times 2}; b_1, b_2 \in \mathbb{Z}^2$

Lattice: $\Lambda(B) = \{x = By \mid y \in \mathbb{Z}^2\}$



Shortest vector problem (SVP)



(Worst-case) Lattice Problems

- **SVP:** Find shortest vector in lattice, given random basis. NP-hard (Ajtai'96)
- **Approximate SVP (αSVP):** Find short vector (norm < α times norm of shortest vector). Hardness depends on α (for α used in crypto not NP-hard).
- **CVP:** Given random point in underlying vectorspace (e.g. $\mathbb{Z}^n$), find the closest lattice point. (Generalization of SVP, reduction from SVP)
- **Approximate CVP (αCVP):** Find a „close" lattice point. (Generalization of αSVP)

(Average-case) Lattice Problems Short Integer Solution (SIS)

$\mathbb{Z}_p^n = \text{n-dim.}$ vectors with entries mod p ($\approx n^3$)

Goal:

Given $\boldsymbol{A} = (\boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_m}) \in \mathbb{Z}_p^{n \times m}$

Find „small“ $\boldsymbol{s} = (s_1, \dots, s_m) \in \mathbb{Z}^m$ such that

$$\boldsymbol{As} = \boldsymbol{0} \bmod p$$

Reduction from worst-case αSVP.

Hash function

Set $m > n \log p$ and define $f_{\boldsymbol{A}}: \{0,1\}^m \to \mathbb{Z}_p^n$ as

$$f_{\boldsymbol{A}}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} \bmod p$$

Collision-resistance: Given short $\boldsymbol{x}_1, \boldsymbol{x}_2$ with $\boldsymbol{A}\boldsymbol{x}_1 = \boldsymbol{A}\boldsymbol{x}_2$ we can find a short solution as

$$\boldsymbol{A}\boldsymbol{x}_1 = \boldsymbol{A}\boldsymbol{x}_2 \Rightarrow \boldsymbol{A}\boldsymbol{x}_1 - \boldsymbol{A}\boldsymbol{x}_2 = \boldsymbol{0}$$

$$\boldsymbol{A}(\boldsymbol{x}_1 - \boldsymbol{x}_2) = \boldsymbol{0}$$

So, $\boldsymbol{z} = \boldsymbol{x}_1 - \boldsymbol{x}_2$ is a solution and it is short as $\boldsymbol{x}_1, \boldsymbol{x}_2$ are short.

Lattice-based crypto

- SIS: Allows to construct signature schemes, hash functions, ... , basically minicrypt.
- For more advanced applications: Learning with errors (LWE)
  - Allows to build PKE, IBE, FHE,...
- Performance: Sizes can almost reach those of RSA (just small const. factor), really fast (for lattices defined using polynomials).
- BUT: Exact security not well accessed, yet. Especially, no good estimate for quantum computer aided attacks.

Hash-based Signature Schemes

[Mer89]

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

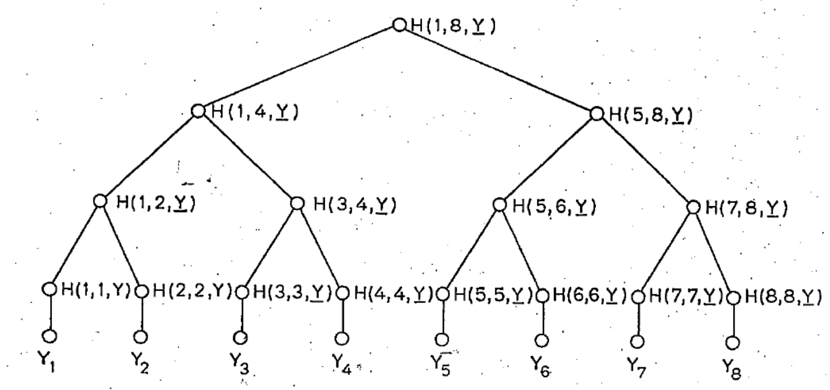
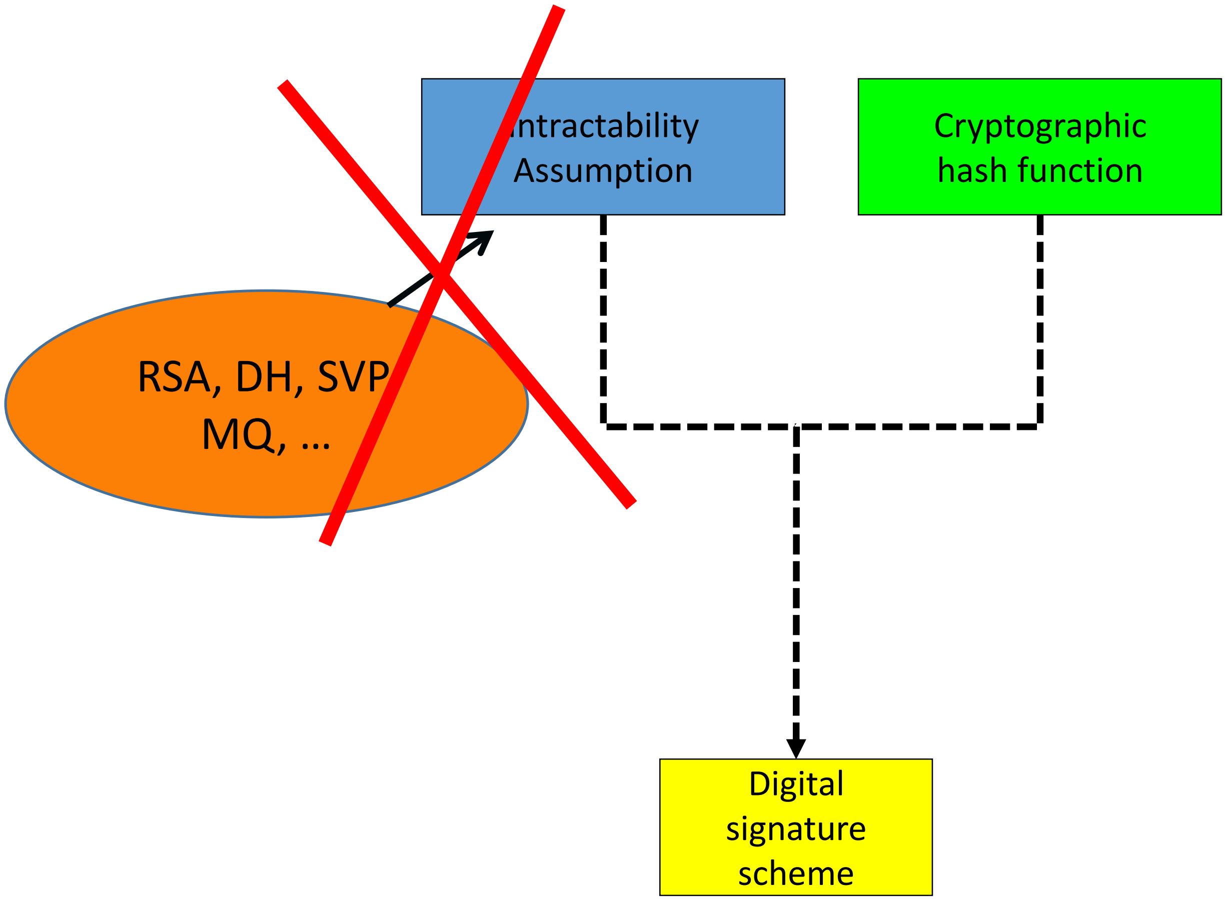


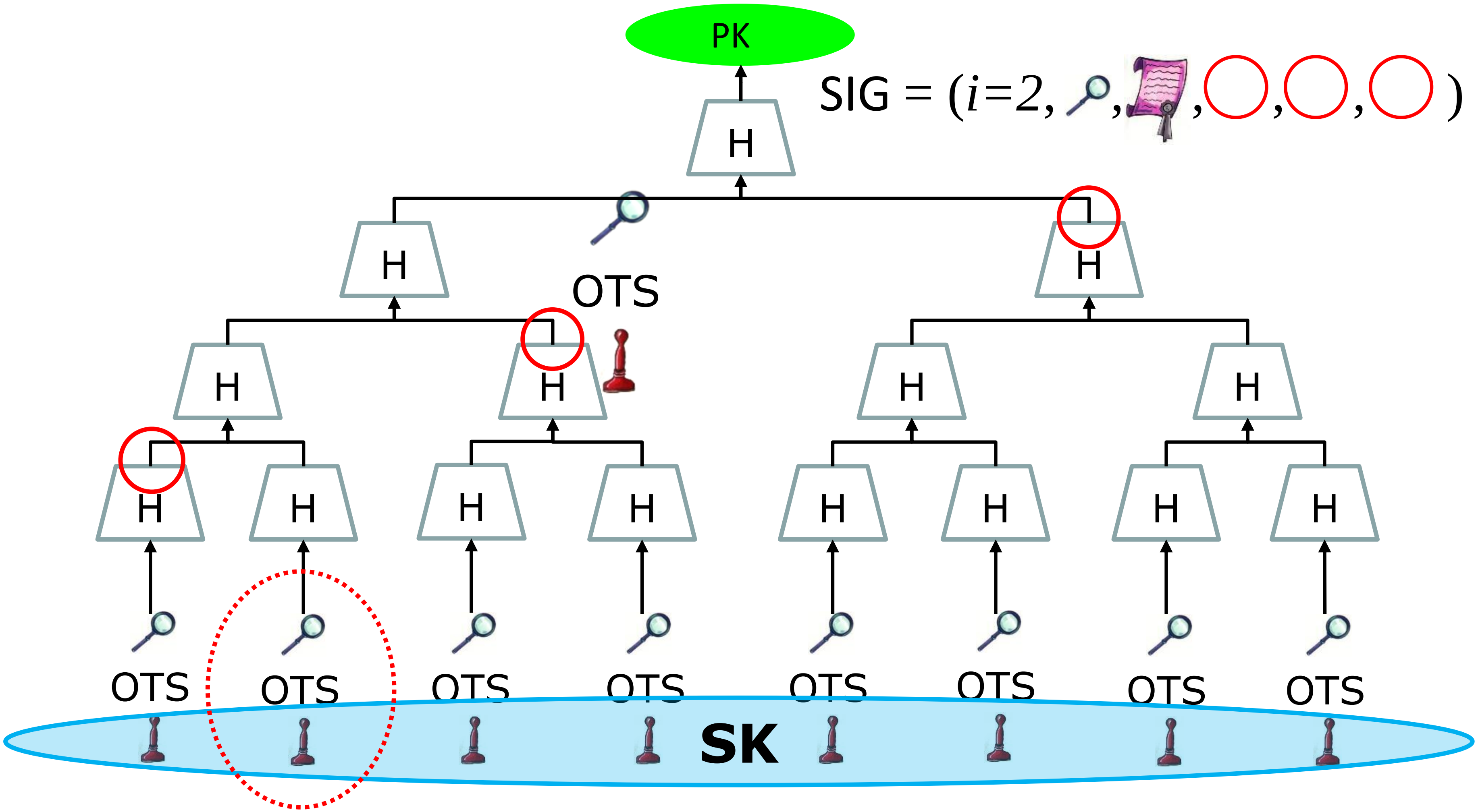
FIG 1
AN AUTHENTICATION TREE WITH N = 8.

PAGE 41B

RSA – DSA – EC-DSA…



Merkle's Hash-based Signatures



Hash-based signatures

- Only signatures
- Minimal security assumptions
- Well understood
- Fast & compact (2kB, few ms), but stateful, or
- Stateless, bigger and slower (41kB, several ms).

PQCRYPTO
ICT-645622

PQCrypto

Initial recommendations

- **Symmetric encryption** Thoroughly analyzed, 256-bit keys:
  - AES-256
  - Salsa20 with a 256-bit key

  Evaluating: Serpent-256, . . .
- **Symmetric authentication** Informati[...]retic MACs:
  - GCM using a 96-bit nonce an[...]bit authenticator
  - Poly1305
- **Public-key encrypti**[...]liece with binary Goppa codes:
  - length $n =$ [...]nension $k = 5413$, $t = 119$ errors

  Evaluatin[...]DPC, Stehlé-Steinfeld NTRU, . . .
- **Pu**[...]**signatures** Hash-based (minimal assumptions):
  - XMSS with any of the parameters specified in CFRG draft
  - SPHINCS-256

  Evaluating: HFEv-, . . .

Confidence inspiring solutions are slow, too big, ...

PQCRYPTO
ICT-645622

TODOs

- Increase confidence for other schemes: (Quantum) cryptanalysis
- Improve existing schemes
- Create code-base

Basis for standards, certification, ... , deployment

Thank you!
Questions?