MPTS 2020 — Virtual event November 4-6, 2020
NIST Workshop on Multi-Party Threshold Schemes

Publicly Verifiable Secret Sharing and Its Use in Threshold Cryptography

Berry Schoenmakers
Coding & Crypto group
Dept of Mathematics & Computer Science

November 4, 2020

PRIVLEDGE

TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Outline

I. Threshold secret sharing

- Shamir *t*-out-of-*n* threshold scheme

II. Publicly verifiable secret sharing (PVSS)

- Focus on ElGamal-based construction, CRYPTO'99

"A Simple Publicly Verifiable Secret Sharing Scheme and its Application to Electronic Voting"

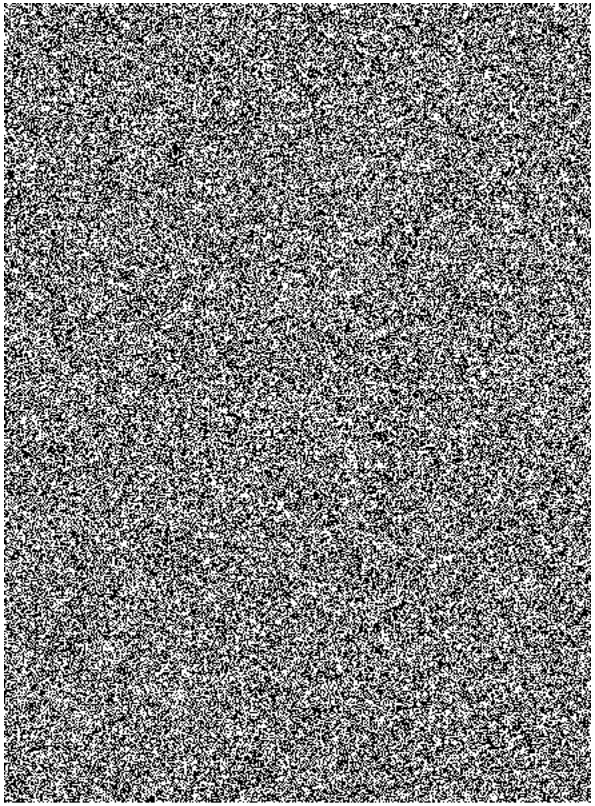
III. Applications in threshold cryptography

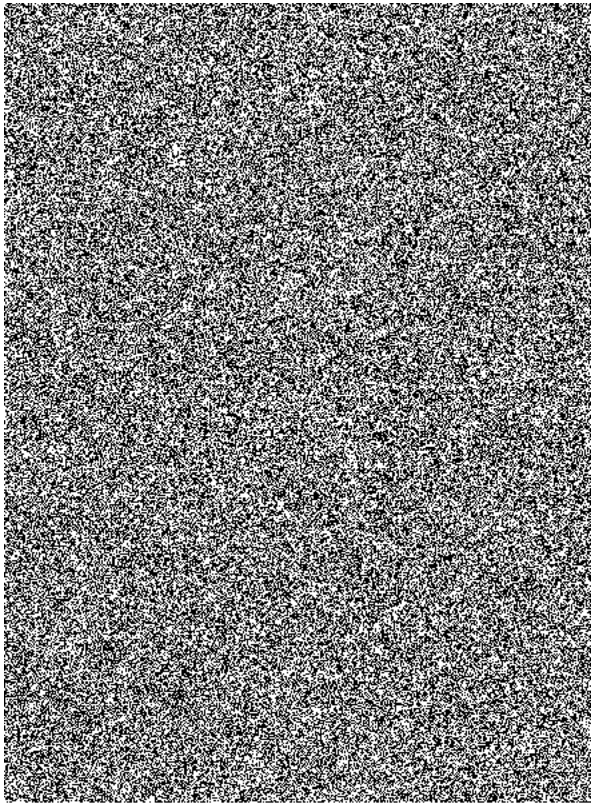
Part I

Threshold Secret Sharing

Visual secret sharing

Naor & Shamir, Eurocrypt‘97





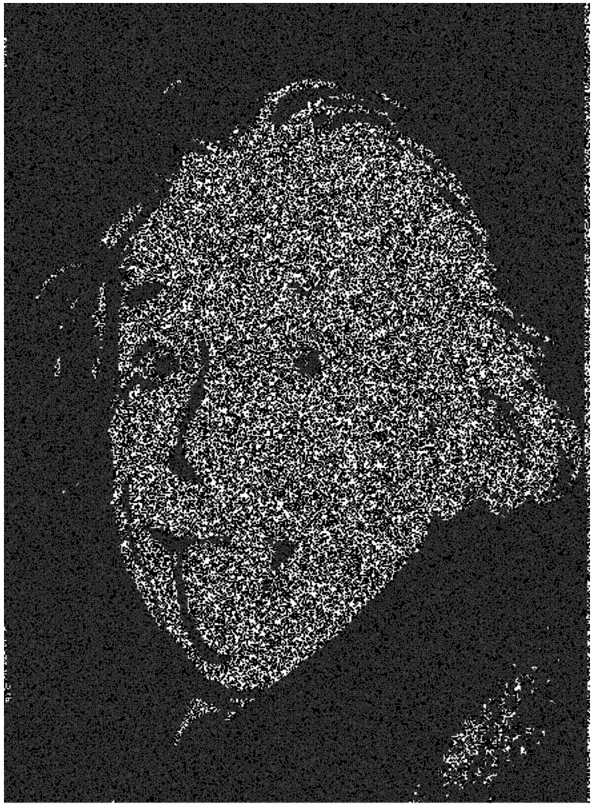
No information at all in individual slides !

Perfect security

Courtesy Benne de Weger

Visual secret sharing

Naor & Shamir, Eurocrypt‘97



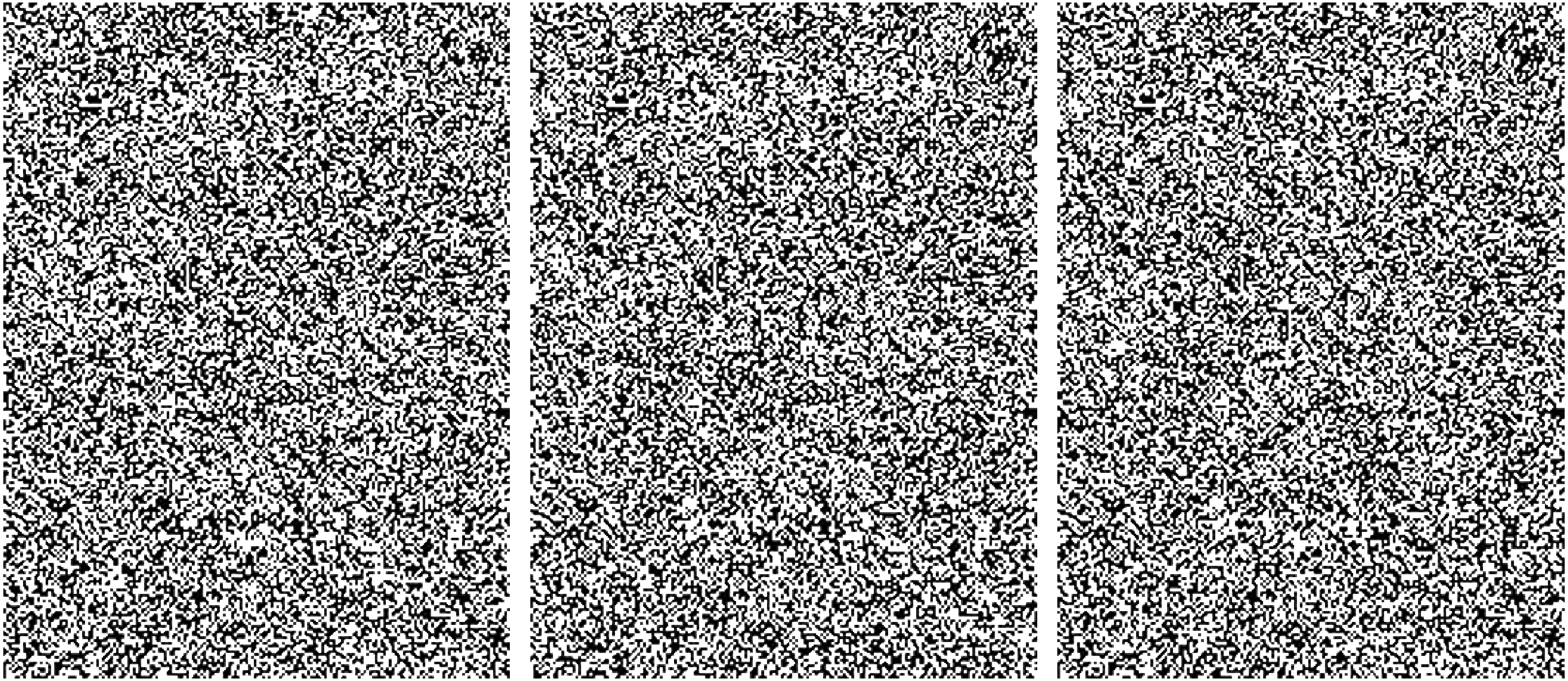
No information at all in individual slides !

Perfect security

Courtesy Benne de Weger

Visual secret sharing

Naor & Shamir, Eurocrypt‘97



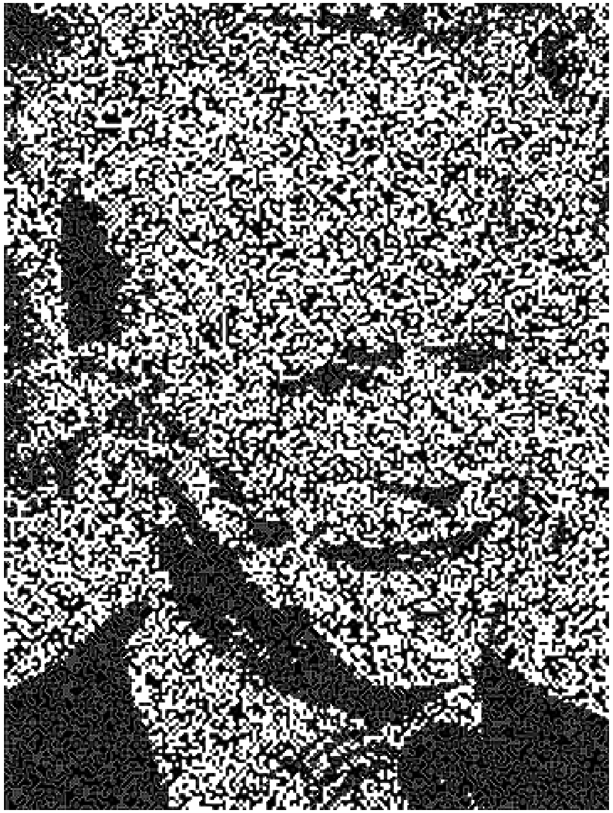
Middle slide turns into anything
we like with forged second slide,
left or right!

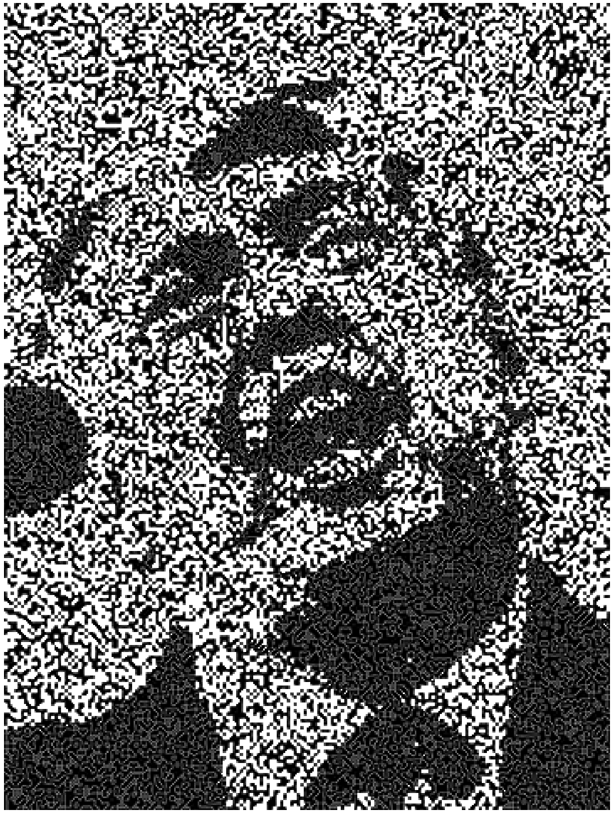
Perfect forgeries

Courtesy Benne de Weger

Visual secret sharing

Naor & Shamir, Eurocrypt'97



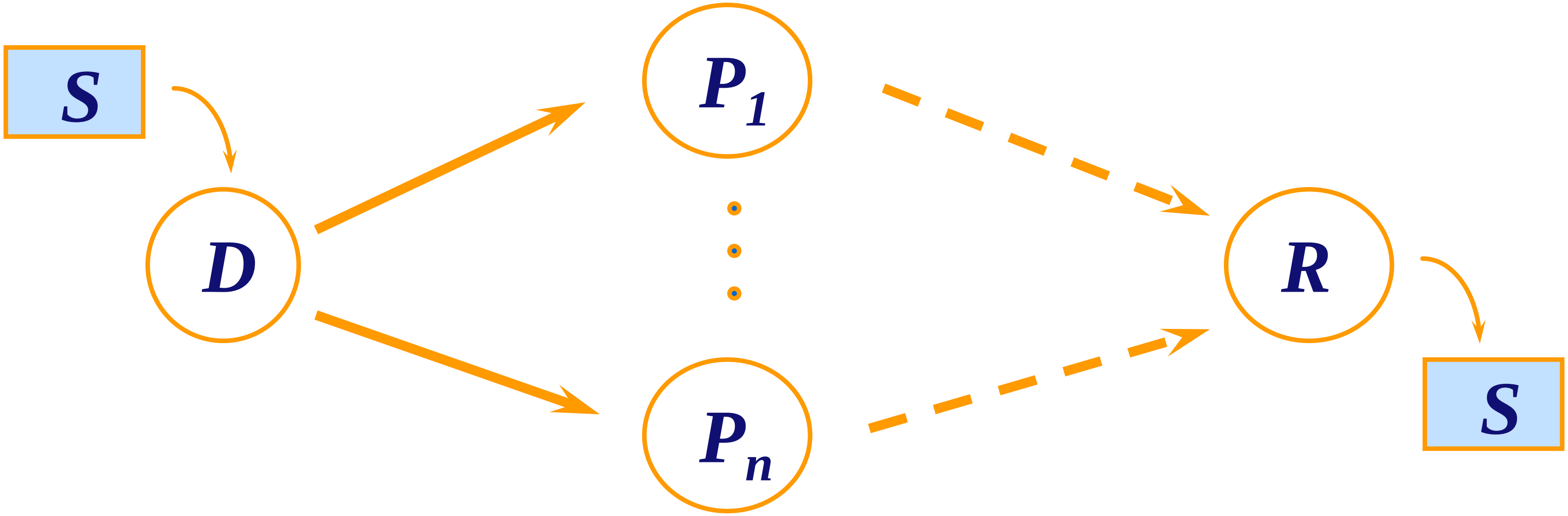


Middle slide turns into anything
we like with forged second slide,
left or right!

Perfect forgeries

Courtesy Benne de Weger

Threshold secret sharing, $1 \leq t \leq n$



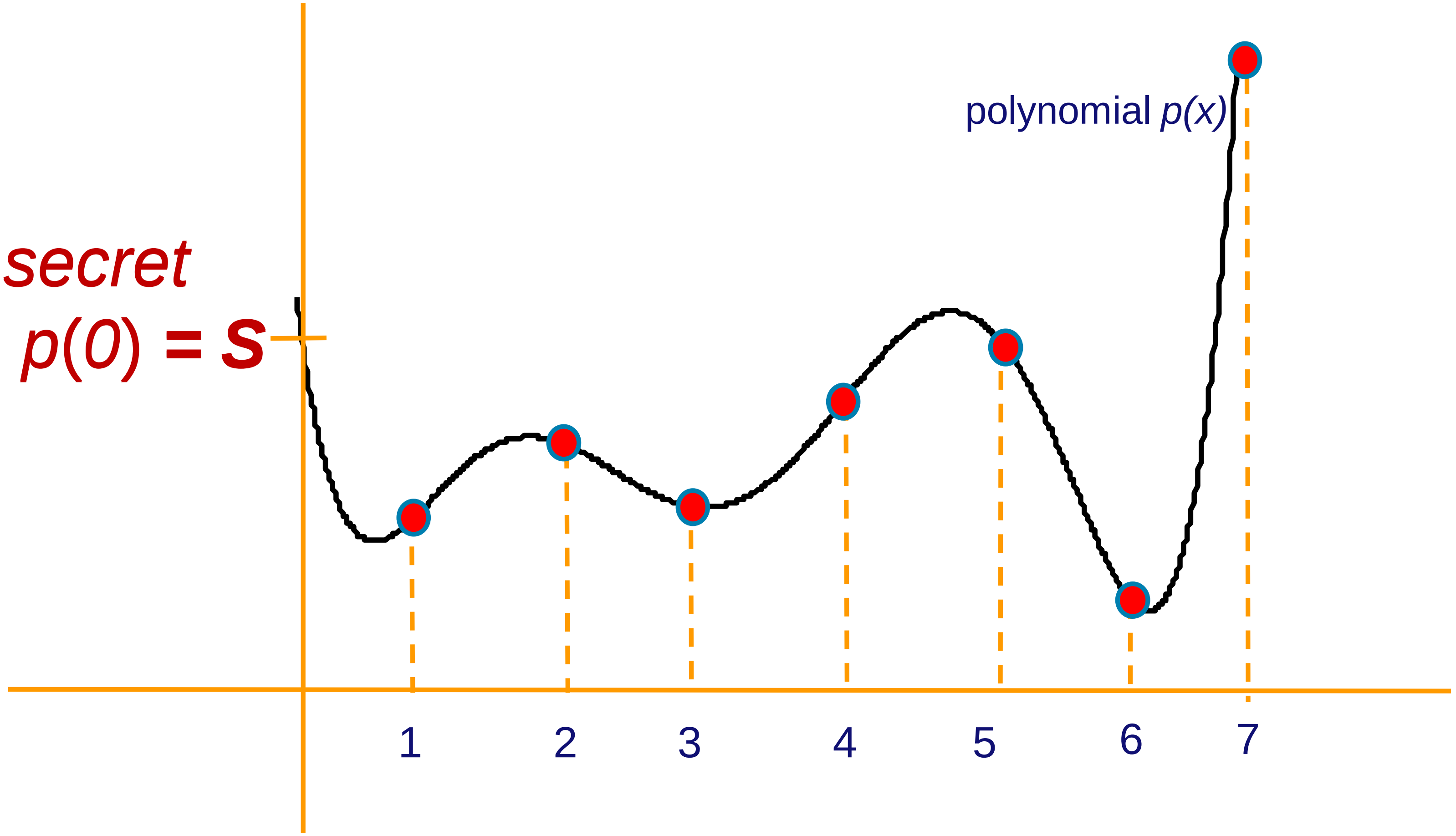
- **Distribution**: *dealer D* distributes shares S_i of secret S among *participants* $P_1, \ldots, P_n$

- **Reconstruction**: any size-t set of participants get secret S from their shares S_i

Shamir’s scheme with threshold *t*

Distribute *S*: give ● $S_i = p(i)$ to participant P_i for random $p(x)$ of degree < t

Reconstruct *S*: Lagrange interpolation of any *t* points ●



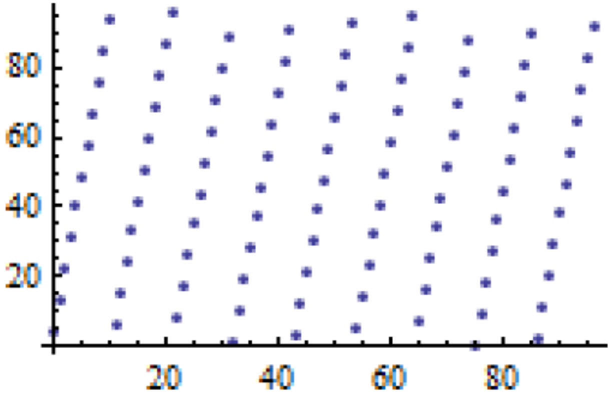
secret

p(*0*) = ***S***

Shamir threshold scheme (actual version)

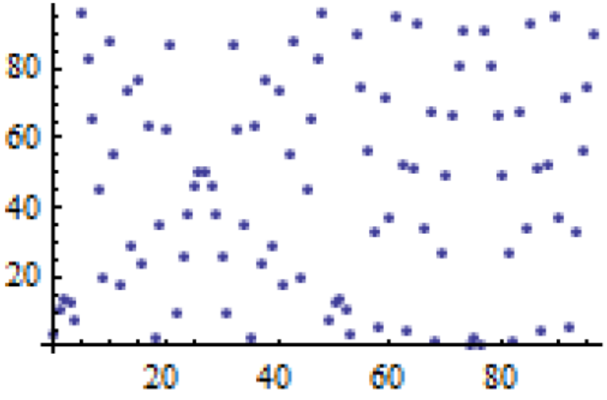
Polynomials over finite field $\mathbb{Z}_q$

prime ***q* = 97**



straight line (mod 97)

degree ***d* = 1**



parabola (mod 97)

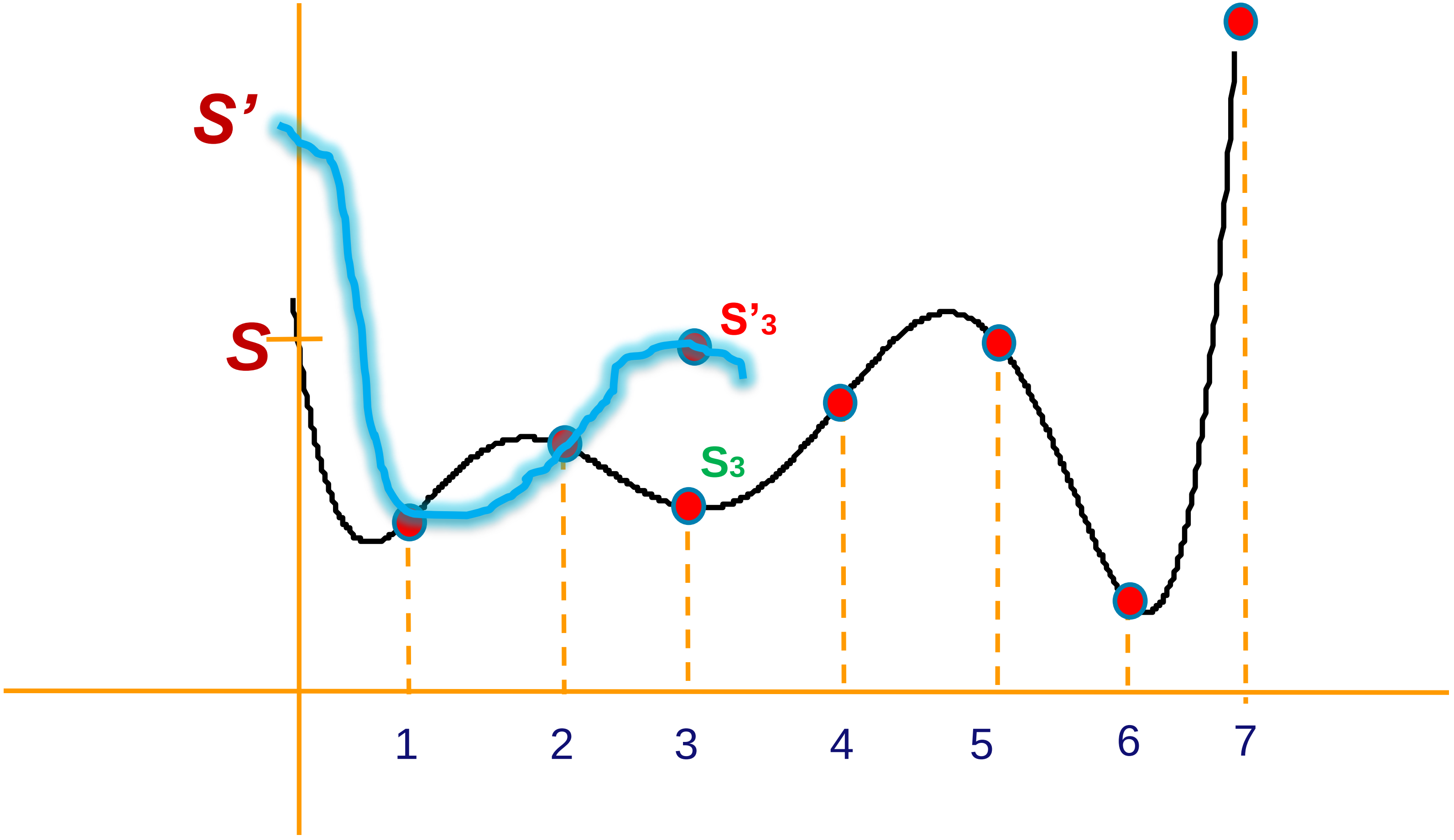
degree ***d* = 2**

Mathematically, same behavior as polynomials over $\mathbb{R}$:

- Polynomial of degree $d > 0$ has at most d roots.
- Polynomial of degree d is uniquely determined by $d + 1$ points.

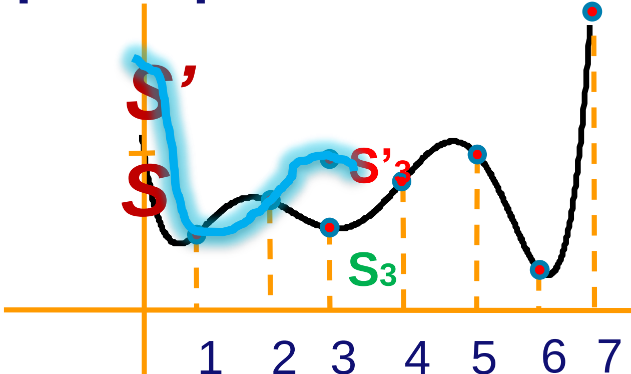
Cheating: modify point on polynomial (by dealer or by a participant)

Example: replace share S_3 by a different share S'_3
to change secret from *S* to *S'*



Shamir threshold scheme

- Pros:
  - Simple and efficient
  - Perfect secret sharing:
    - if you lack a single share, no information on the secret
    - also for finite fields of low order *q*
  - Information-theoretically secure
    - hence quantum-secure
- Cons:
  - Requires private channels from dealer to participants
  - No protection against active attacks
    - cheating dealer: inconsistent shares
    - cheating participants: forged shares



Part II

Publicly Verifiable Secret Sharing

Verifiable Secret Sharing (VSS)

- To stop cheating dealers/participants

- Error-correcting codes to deal with cheating participants [*McEliece&Sarwate'81*]

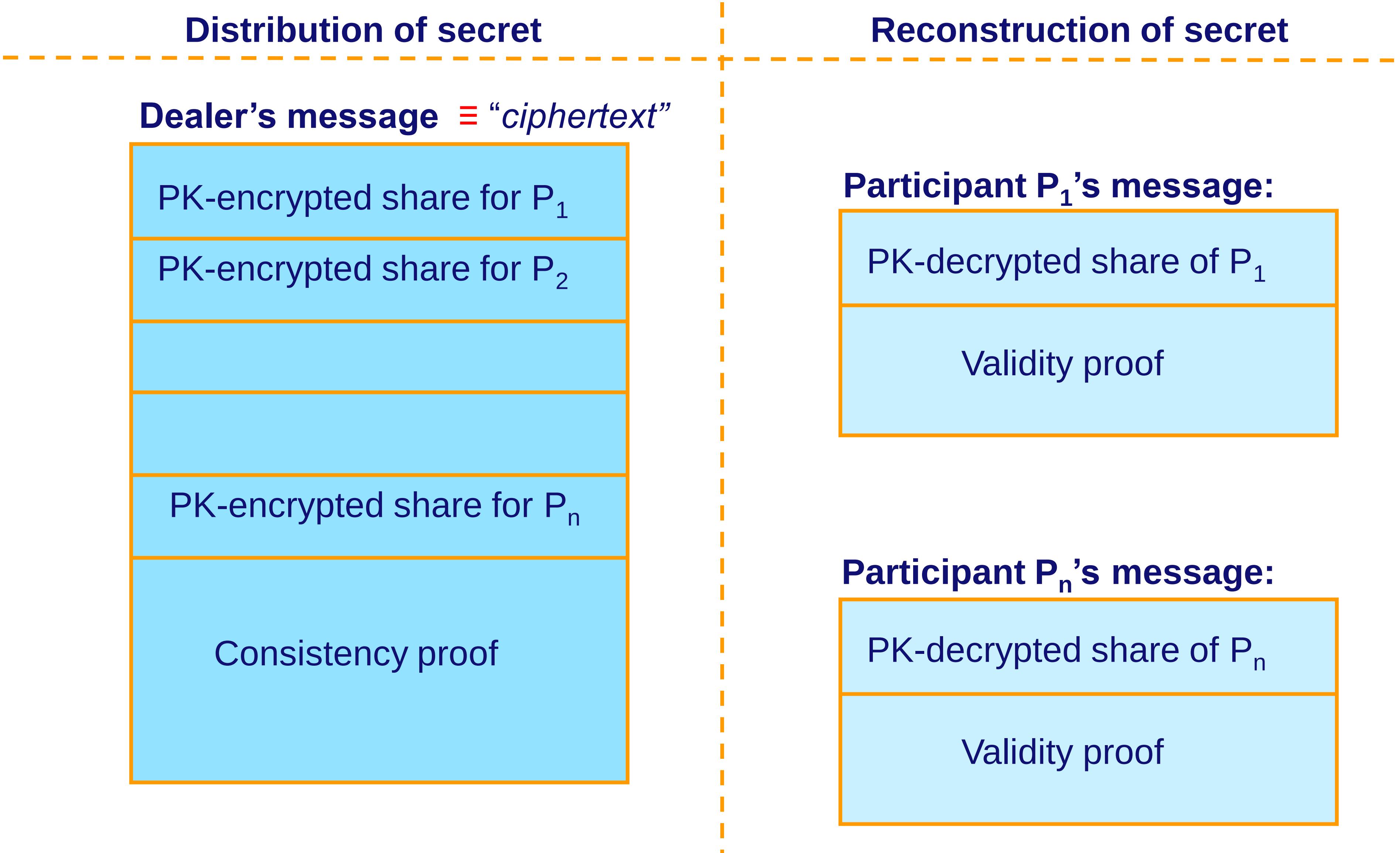
- VSS scheme to deal also with a cheating dealer [*Chor&Goldwasser&Micali&Awerbuch'85*]
- Noninteractive VSS [*Feldman'87*]
- Information-theoretic VSS [*Pedersen'91*]

- Publicly VSS [*Stadler'96*]

Publicly Verifiable Secret Sharing (PVSS)

- Public verifiability
  - anybody is able to check dealer and participants
- Use public broadcast channel
  - e.g., blockchain
  - private channels are not publicly verifiable
- Resort to computational protection of shares
  - e.g., Decision Diffie-Hellman assumption
- Noninteractive scheme
  - “no complaining” by individual participants

PVSS ≡ *“public key cryptosystem with threshold decryption”*



Distribution of secret

Dealer’s message ≡ *“ciphertext”*

PK-encrypted share for P_1

PK-encrypted share for P_2

PK-encrypted share for P_n

Consistency proof

Reconstruction of secret

Participant P_1’s message:

PK-decrypted share of P_1

Validity proof

Participant P_n’s message:

PK-decrypted share of P_n

Validity proof

ElGamal-based PVSS (CRYPTO ‘99)

- **ElGamal encryption for random messages (like key encapsulation in KEMs)**
- **Works with any cyclic group <*g*> of prime order *q***
  - **let *h* be a group element s.t. nobody knows $\log_g h$**

1. **Initialization:**
   - **P_i registers public key $y_i = h^{x_i}$**
2. **Distribution:**
   - **commitments $C_j = g^{\alpha_j}$ to coefficients α_j of polynomial $p(x)$**
   - **encrypted shares $Y_i = y_i^{p(i)}$ plus ∑-proof**
3. **Reconstruction:**
   - **P_i decrypts Y_i and releases $S_i = h^{p(i)}$ plus ∑-proof**
   - **recover S by Lagrange interpolation in the exponent**

Properties ≈ threshold-ElGamal cryptosystem

- Works in any Discrete-Log setting with DDH assumption
  - secure against passive attacks
- Dynamic selection of sets of participants
  - reuse already registered public keys
  - no need for distributed key generation (DKG)
- Homomorphic properties
  - ZeroKnowledge and MPC friendly
- Security against active attacks from ∑-proofs
- Performance roughly linear in *n*

Part III

Applications in Threshold Cryptography

(1/4) Direct applications

- In any application of threshold secret sharing
  - second layer on top of Shamir’s scheme
    - to protect Shamir’s scheme against active attacks
    - to replace “plain” Verifiable Secret Sharing

- For (public) storage of secret information
  - in escrow, on a blockchain
    - threshold property for more resilience
    - ad hoc/dynamic selection of participants

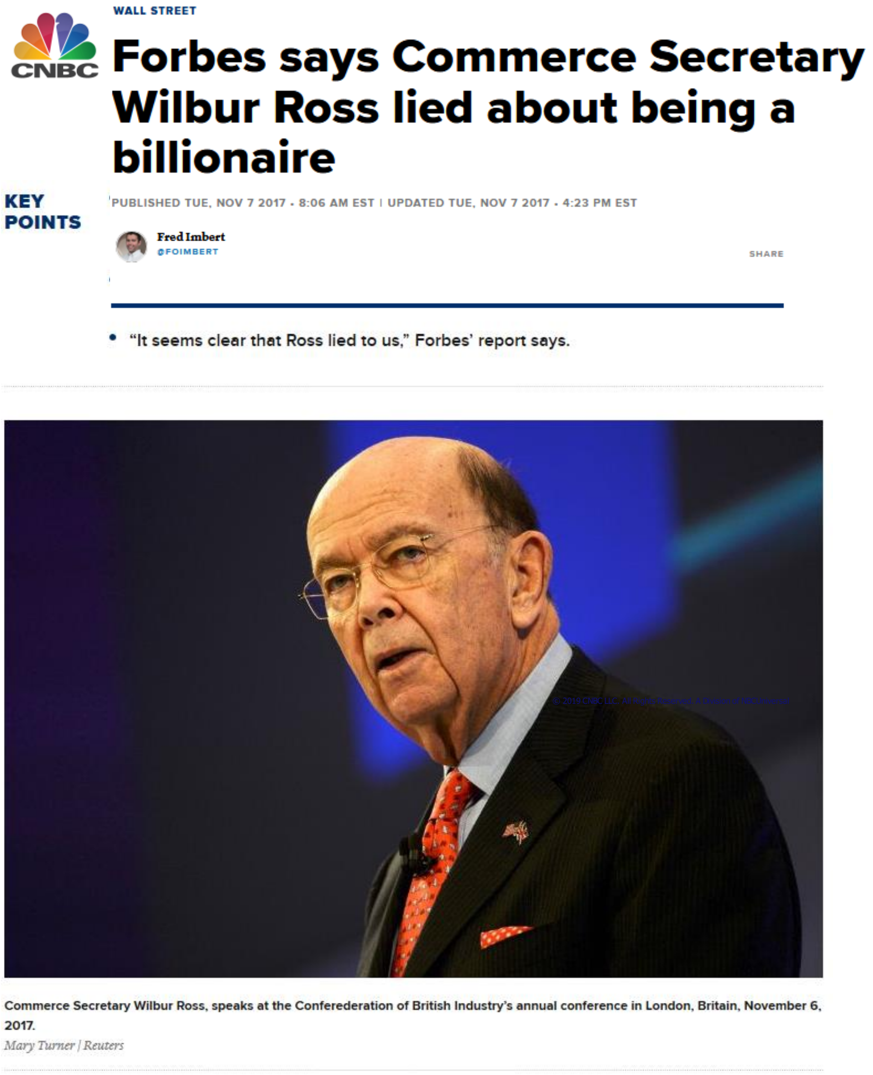
(2/4) An alternative threshold homomorphic cryptosystem

- **For use in e-voting:**
  - **to encrypt the votes**
  - **allows homomorphic tallying**
  - **compatible with efficient ∑-proofs**
- **For use in e-auctions**
- **…**

- **More generally, for use with verifiable (auditable) MPC:**
  - **outsource MPC to (dynamically) selected parties**
  - **verifiable input and output for MPC**
  - **implemented in PRIVILEDGE using MPyC framework**

To solve World's Billionaires **Problem**

Wannabe billionaires …



WALL STREET

Forbes says Commerce Secretary Wilbur Ross lied about being a billionaire

KEY POINTS

PUBLISHED TUE, NOV 7 2017 • 8:06 AM EST | UPDATED TUE, NOV 7 2017 • 4:23 PM EST

Fred Imbert
@FOIMBERT

- “It seems clear that Ross lied to us,” Forbes' report says.

Commerce Secretary Wilbur Ross, speaks at the Conferederation of British Industry's annual conference in London, Britain, November 6, 2017.

Mary Turner | Reuters

Courtesy Toon Segers

The *World's Billionaires* Problem

priviledge-project.eu

Upgrade of Yao's *Millionaires' problem*

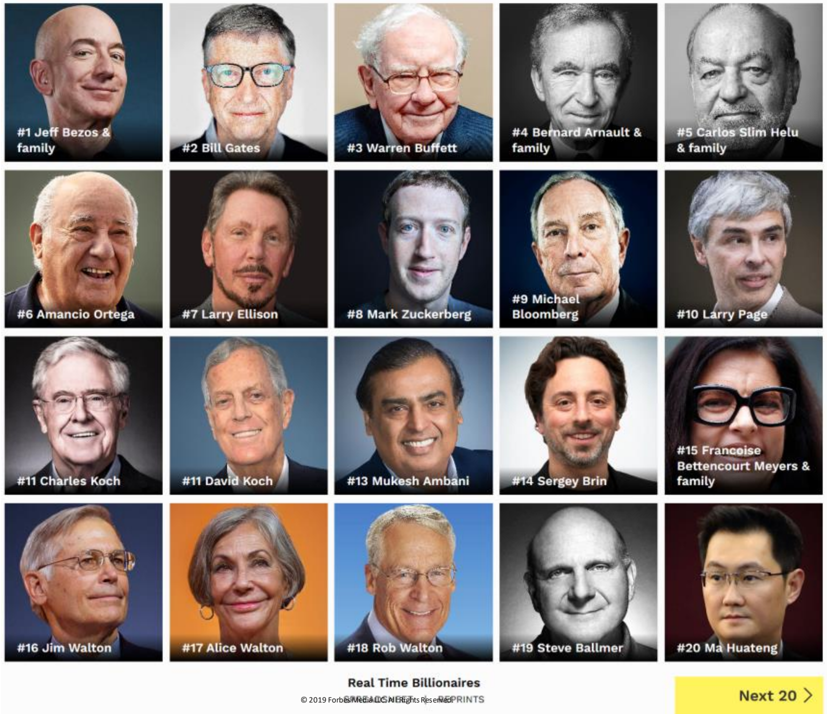
Verifiable input:

- Encrypted tax returns
- Signed by tax authority
- Posted on blockchain

Verifiable output:

- Top 400 billionaires world-wide

Privacy for all outside top 400



© 2019 Forbes Media LLC. All Rights Reserved.

Source: Forbes.com,
Published March 5, 2019 (link)

(3/4) Randomness beacons

- **Verifiable randomness, for blockchains like Cardano**
- **PVSS extensions like SCRAPE [*Cascudo&David, ACNS ’17*]**

- **Note: SCRAPE speed up for consistency proof may kick in a bit later than suggested**
  - **CRYPTO’99: recompute** $X_i = \prod_{j=0}^{t-1} C_j^{i^j} = h^{p(i)}$ **(*)**
  - **SCRAPE: check** $\prod_{i=1}^{n} X_i^{c_i^{\perp}} = 1$ **(**)** ***with*** $(c_1^{\perp}, \ldots, c_n^{\perp})$ ***in dual code***

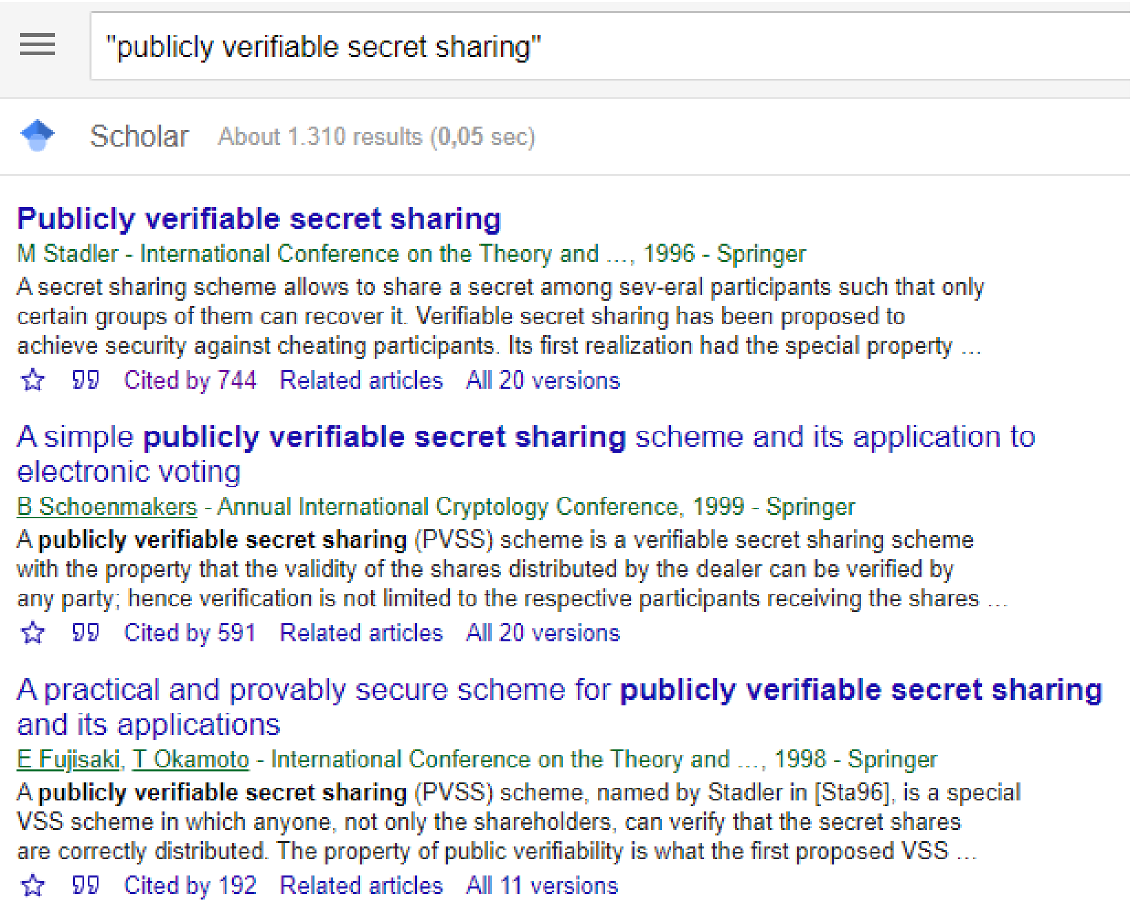
Claimed to reduce #exponentiations from O(nt) to O(n).

However exponentiations in (*) are not full ones, unlike in (**) :

- **Horner’s scheme for (*):** ***O*(*n t* log *n*) multiplications**
- **(**) takes:** ***O*(*n* log *q*) multiplications**
- **break even around *n* = 100 participants, say, as *q* is a large prime**

(4/4) What else?

- **(Ad hoc) group key management, key distribution**
- **… many more applications**



"publicly verifiable secret sharing"

Scholar About 1.310 results (0,05 sec)

Publicly verifiable secret sharing
M Stadler - International Conference on the Theory and ..., 1996 - Springer
A secret sharing scheme allows to share a secret among sev-eral participants such that only certain groups of them can recover it. Verifiable secret sharing has been proposed to achieve security against cheating participants. Its first realization had the special property ...
☆ 99 Cited by 744 Related articles All 20 versions

A simple **publicly verifiable secret sharing** scheme and its application to electronic voting
B Schoenmakers - Annual International Cryptology Conference, 1999 - Springer
A **publicly verifiable secret sharing** (PVSS) scheme is a verifiable secret sharing scheme with the property that the validity of the shares distributed by the dealer can be verified by any party; hence verification is not limited to the respective participants receiving the shares ...
☆ 99 Cited by 591 Related articles All 20 versions

A practical and provably secure scheme for **publicly verifiable secret sharing** and its applications
E Fujisaki, T Okamoto - International Conference on the Theory and ..., 1998 - Springer
A **publicly verifiable secret sharing** (PVSS) scheme, named by Stadler in [Sta96], is a special VSS scheme in which anyone, not only the shareholders, can verify that the secret shares are correctly distributed. The property of public verifiability is what the first proposed VSS ...
☆ 99 Cited by 192 Related articles All 11 versions

Conclusion

- PVSS relatively simple
  - behaves as ElGamal encryption
  - with dynamically defined threshold decryption
  - noninteractive --> “no complaining”
  - ZK and MPC friendly
  - performance reasonable (linear in *n*)

- PVSS versatile cryptographic primitive
  - as second layer on top of Shamir scheme
  - for threshold cryptography

- Questions ? To berry@win.tue.nl

H2020 EU-project

priviledge-project.eu

This work is part of a project that received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 780477 (PRIVILEDGE).