# A Lattice-Based Threshold Ring Signature Scheme (TRSS-L)

Pierre-Louis Cayrel[1] Richard Lindner [2]
Markus Rückert [2] **Rosemberg Silva** [3]

[1]Center for Advanced Security Research Darmstadt (CASED)
[2]Technische Universität Darmstadt (TUD)
[3]State University of Campinas (UNICAMP) [3]

LatinCrypt 2010

[3]Funded by FAPESP Grant 2008/07949-8

# Outline

# General Goal: Reuse cryptographic schemes

## Approach

Explore similarities and differences between Codes and Lattices in order to enhance cryptographic schemes:

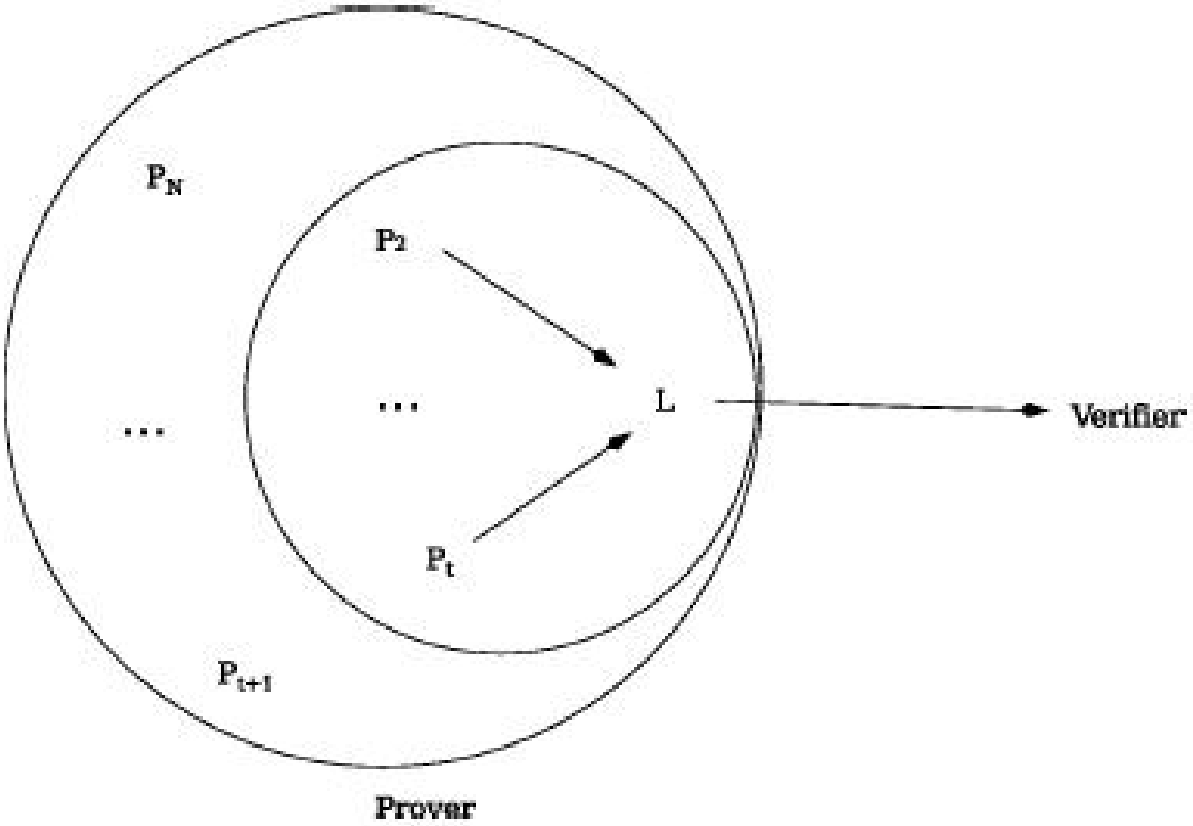
- converting schemes between Lattices and Codes.
- replacing hard problems as security basis
- enhancing performance and security

# Specific Goal: build a TRSS-L

## Lattice-Based Threshold Ring Signature

- Reuse structure from code-based solution: TRSS-C.
- Apply CLRS identification scheme as basis.
  - Lattice hard problem as security assumption.
  - Worst-case to average-case reduction, typical of lattice-base schemes.

# Threshold Ring Signature



# Threshold Ring Signature

## Description

- Subset $S$ of a group of users $U$ jointly sign a document.
- Size of $S$ is above a threshold $t$.
- Anonymity of members of $S$ is preserved.
- Property of unforgeability.

# From

## Starting Point

- **TRSS-C**: code-based threshold ring signature, generalizing Stern's ID scheme.
  - Generalization of Stern ID Scheme (3-pass, 2/3 soundness error)
  - Fiat-Shamir heuristic
- **Cayrel-Véron**: code-based ID scheme from which CLRS was derived.
  - 5-pass construction
  - Permutations of a q-ary code
  - Soundness error approximately 1/2

# Construction

## Sequence

1. Stern ID Scheme
2. Duality -> Véron ID Scheme
3. 5-Pass + Permutation + q-ary code -> Cayrel-Véron ID Scheme
4. Using the hardness of SIS as security assumption -> CLRS (to be presented at ProvSec 2010)
5. Generalization of CLRS + FS Shamir -> TRSS-L

# Identification Schemes

## Security Goals

- **ID**: authentication, access control.

## Common Constructions

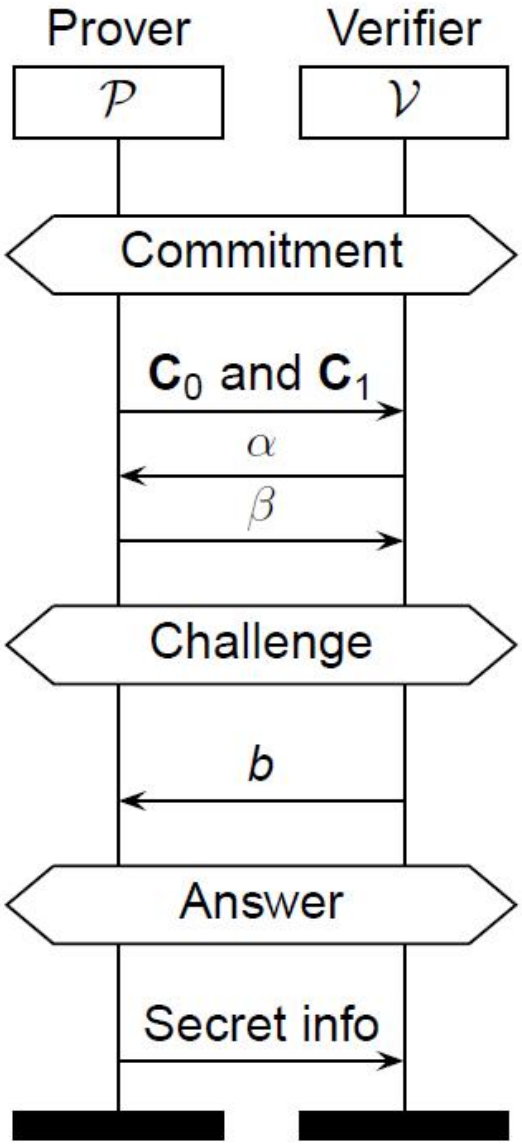
- **3 phases**: commitment, challenge, answer.
- **FS Heuristics**: converting an ID scheme into a signature one.

# ID Schemes as Interactive Proof Systems

## Properties

- **Completeness**: honest prover can always demonstrate his identity.
- **Soundness**: impersonator can never successfully fake an identity.
- **Zero-Knowledge**: verifier is only convinced about the prover's id, without gaining knowledge on how to fake it.

# Lattice ID Scheme



# Security

## Notes

- Concurrently secure
- Assumptions
  - Existence of a string commitment scheme
  - Hardness of SIS problem

# CLRS Scheme: algorithms

## Key Generation

- Private Key: binary vector $\mathbf{x} \in \mathbb{F}_2^m$ with Hamming weight $m/2$
- Public Key: $\mathbf{y} \in \mathbb{Z}_q^m$ such that $\mathbf{y} = \mathbf{A}\mathbf{x} \bmod q$

## Interactive Proof

The prover convinces the verifier that he knows a solution $\mathbf{x}$ to $\mathbf{y} = \mathbf{A}\mathbf{x} \bmod q$, with Hamming weight $m/2$, without revealing it.

# Signatures from ID

## Fiat-Shamir Heuristics

- Generalize Stern ID scheme.
  - $\text{Signer}_i$ as prover, leader as verifier.
  - Leader as prover, V as verifier.
  - Convince V that the group of signers knows a codeword of Hamming weight $tw$.
- Replace the verifier by a random oracle.
- Apply the document, or a hash thereof.

# Code-based TRSS

## Generalization of an ID Scheme

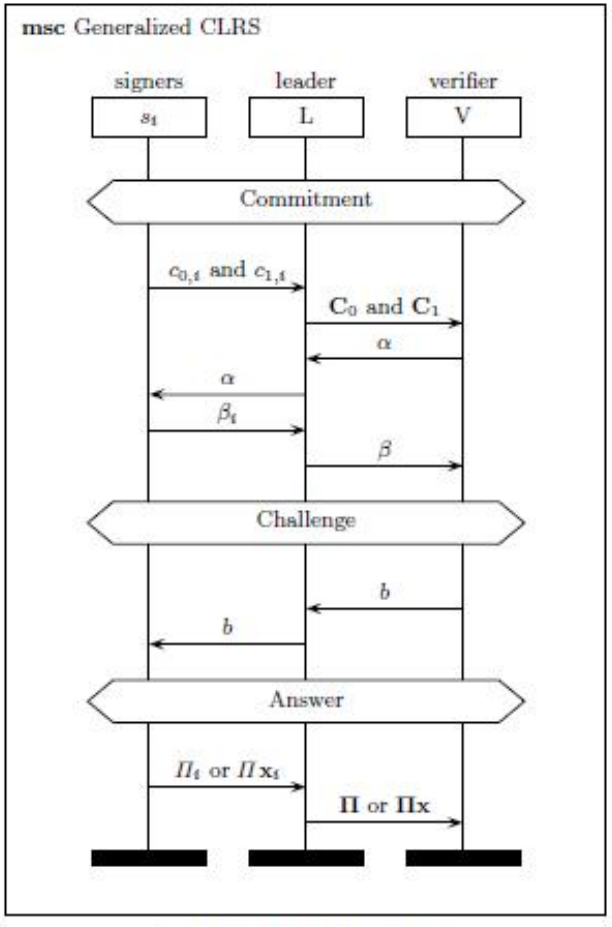
- The leader acts as verifier to each of the other signers (individual codes).
- The leader acts as prover to the final verifier (direct sum of codes).
- Individual private keys are codewords of Hamming weight $w$.
- The group private key is composed by blocks of small private keys. It has Hamming weight $tw$.
- Properties of unforgeability and anonymity.
- Hardness of Syndrome Decoding Problem as security assumption.

# Generalized CLRS

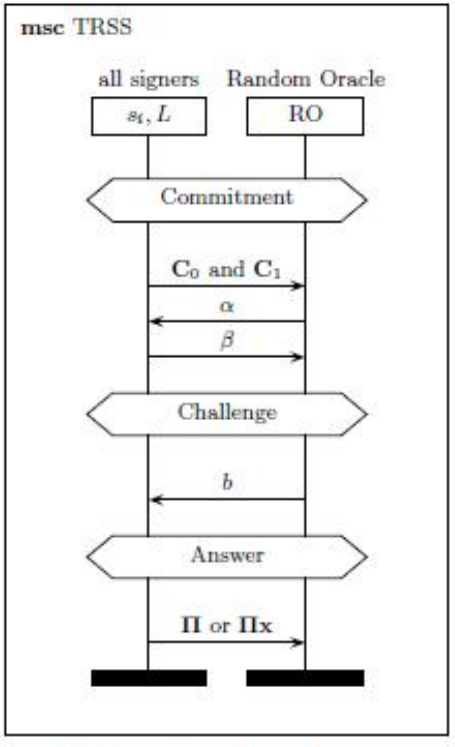
## Notes

- Leader (as verifier) $L$ and each signer $s_i$ (as prover) execute CLRS.
- Non-signers are considered to have null-vectors as private keys.
- Verifier and Leader (this time, as prover) execute CLRS.
- Block permutations allow anonymity.
- Applying FS heuristic to the generalized CLRS results in TRSS-L

# Generalized CLRS




(a) Generalized CLRS




(b) TRSS from FS Heuristic

# Generalized CLRS: key generation

1. $A_i \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$
2. $x_i \xleftarrow{\$} \mathbb{F}_2^m$, such that $\mathbf{weight}(x_i) = m/2$
3. $y_i = A_i x_i \bmod q$
4. $[A_i; -y_i]\,[x_i; 1]^T = 0 \bmod q$
5. $sk = [x_i; 1]$, with length $m + 1$ and $\mathbf{weight}(x_i) = m/2 + 1$
6. $pk = A'_i = [A_i; -y_i]$

# Generalized CLRS: key generation

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}'_0 & 0 & \cdots & 0 \\ 0 & \mathbf{A}'_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{A}'_{N-1} \end{bmatrix}$$

# Signatures from ID

## Fiat-Shamir Heuristics

- Generalize CLRS ID scheme.
  - Signer$_i$ as prover, leader as verifier.
  - Leader as prover, V as verifier.
  - Convince V that the group of signers knows a codeword of Hamming weight $t(m/2 + 1)$.
- Replace the verifier by a random oracle.
- Apply the document, or a hash thereof.

## Generalized CLRS scheme

- honest verifier zero-knowledge proof of knowledge;
- soundness error limited by $\frac{q+1}{2q}$;
- sentence: a group of $t$ signers knows a vector $v$ of length $N(m+1)$ and Hamming weight $t(m/2+1)$, such that each of the $N$ blocks of size $m+1$ either weighs $m/2+1$ or zero.

# Estimates

| Scheme | Signature Size (Mbytes) | Number of Rounds |
|---|---|---|
| TRSS-C | 42 | 170 |
| TRSS-L | 40 | 100 |

**Table:** Comparing TRSS Schemes for N=100, and security=100 bits

# Lines to Investigate

- Try different underlying ID schemes (e.g. Lyubashevsky's).
- Lower soundness error of CLRS scheme.
- Build TRSS on top of a signature scheme.

# Summary

## Combined Approach: Codes and Lattices

- Similarities between the areas allow transposition/replacement of security assumptions, keeping the scheme's structure.
- Differences allow gains in terms of performance and security.
- Resulting constructs may provide enhancements (more variety in security assumptions, better performance)

**Thanks!**

# CLRS Scheme: key generation

KeyGen:

$\mathbf{x} \xleftarrow{\$} \{0,1\}^m$, s.t. $\mathrm{wt}(\mathbf{x}) = m/2$

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$

$\mathbf{y} \longleftarrow \mathbf{A}\mathbf{x} \bmod q$

Com $\xleftarrow{\$}$ $\mathcal{F}$, suitable family of commitment functions

**Output** $(\mathrm{sk}, \mathrm{pk}) = (\mathbf{x}, (\mathbf{y}, \mathbf{A}, \mathrm{Com}))$

◂ Return

# CLRS Scheme: id protocol

| **Prover** $\mathcal{P}(\mathrm{sk}, \mathrm{pk})$ | | **Verifier** $\mathcal{V}(\mathrm{pk})$ |
|---|---|---|
| $(\mathrm{sk}, \mathrm{pk}) = (\mathbf{x}, (\mathbf{y}, \mathbf{A}, \textsc{Com})) \longleftarrow \textsc{KeyGen}$ | | |
| $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m, \sigma \xleftarrow{\$} S_m, \mathbf{z} \longleftarrow \mathbf{P}_\sigma \mathbf{x}$ | | |
| $\mathbf{r}_0 \xleftarrow{\$} \{0,1\}^n, \mathbf{r}_1 \xleftarrow{\$} \{0,1\}^n$ | | |
| $c_0 \longleftarrow \textsc{Com}(\sigma \,\|\|\, \mathbf{A}\mathbf{u}; \mathbf{r}_0)$ | | |
| $c_1 \longleftarrow \textsc{Com}(\mathbf{z} \,\|\|\, \mathbf{P}_\sigma \mathbf{u}; \mathbf{r}_1)$ | $\xrightarrow{c_0, c_1}$ | |
| | $\xleftarrow{\alpha}$ | $\alpha \xleftarrow{\$} \mathbb{Z}_q$ |
| $\beta \longleftarrow \mathbf{P}_\sigma(\mathbf{u} + \alpha \mathbf{x})$ | $\xrightarrow{\beta}$ | |
| | $\xleftarrow{\text{Challenge } b}$ | $b \xleftarrow{\$} \{0,1\}$ |
| **If** $b = 0$: | $\xrightarrow{\sigma, \mathbf{r}_0}$ | **Check** $c_0 \stackrel{?}{=} \textsc{Com}(\sigma \,\|\|\, \mathbf{A}\mathbf{P}_\sigma^{-1}\beta - \alpha\mathbf{y}; \mathbf{r}_0)$ |
| | | $\sigma \stackrel{?}{\in} S_m$ |
| **Else**: | $\xrightarrow{\mathbf{z}, \mathbf{r}_1}$ | **Check** $c_1 \stackrel{?}{=} \textsc{Com}(\mathbf{z} \,\|\|\, \beta - \alpha\mathbf{z}; \mathbf{r}_1)$ |
| | | $\mathbf{z} \stackrel{?}{\in} \{0,1\}^m, \mathrm{wt}(\mathbf{z}) \stackrel{?}{=} m/2$ |

Return

## CLRS Parameters

| Bit-security | n | m | q | Commitment Length (bits) |
|---|---|---|---|---|
| 100 | 64 | 2048 | 257 | 256 |

Return

# CLRS Performance

| Scheme | Secret key [Kbyte] | Public key [Kbyte] | Rounds | Total communication [Kbyte] | SIS norm bound |
|---|---|---|---|---|---|
| Lyubashevsky [11] | 0,25 | 2,00 | 11 | 110,00 | $\tilde{O}(n^2)$ |
| Kawachi et al. [7] | 0,25 | 0,06 | 27 | 58,67 | 1 |
| CLRS | 0,25 | 0,06 | 16 | 35,29 | 1 |

Return

# Generalized Stern (Codes)

1) Commitment Step:

- Each of the signers chooses $y_i \in \mathbb{F}_2^n$ randomly and a random permutation $\sigma_i$ of $\{1, 2, \ldots, n\}$ and sends to $L$ the commitments $c_{1,i}, c_{2,i}$ and $c_{3,i}$ such that :

$$c_{1,i} = h(\sigma_i | H_i y_i^t); \; c_{2,i} = h(\sigma_i(y_i));$$

$$c_{3,i} = h(\sigma_i(y_i \oplus s_i))$$

- $L$ sets the secret $s_i$ of the $N - t$ missing users at 0 and computes the $N - t$ corresponding commitments by choosing random $y_i$ and $\sigma_i$ $(t + 1 \leq i \leq N)$.
- $L$ chooses a random constant $n$-block permutation $\Sigma$ on $N$ blocks $\{1, \cdots, N\}$ in order to obtain the *master commitments*:

$$C_1 = h(\Sigma | c_{1,1} | \ldots | c_{1,N}), C_2 = h(\Sigma(c_{2,1}, \ldots, c_{2,N})),$$

$$C_3 = h(\Sigma(c_{3,1}, \ldots, c_{3,N})).$$

- $L$ sends $C_1, C_2$ and $C_3$ to $V$.

◂ Return

# Generalized Stern (Codes)

2) Challenge Step: $V$ sends a challenge $b \in \{0, 1, 2\}$ to $L$ which sends $b$ to the $t$ signers.
3) Answer Step: Let $P_i$ be one of the $t$ signers. The first part of the step is between each signer and $L$.

- Three possibilities :
  - if $b = 0$ : $P_i$ reveals $y_i$ and $\sigma_i$.
  - if $b = 1$ : $P_i$ reveals $(y_i \oplus s_i)$ (denoted by $(y \oplus s)_i$) and $\sigma_i$.
  - if $b = 2$ : $P_i$ reveals $\sigma_i(y_i)$ (denoted by $(\sigma(y))_i$) and $\sigma_i(s_i)$ (denoted by $(\sigma(s))_i$).
- $L$ simulates the $N - t$ others Stern's protocol with $s_i = 0$ and $t+1 \leq i \leq N$ and sets $s = (s_1, \ldots, s_N)$.
- $L$ computes the answer for $V$ (and sends it) :
  - if $b = 0$ : $L$ constructs $y = (y_1, \cdots, y_N)$ and $\Pi = \Sigma \circ \sigma$ (for $\sigma = (\sigma_1, \cdots, \sigma_N)$) and reveals $y$ and $\Pi$.
  - if $b = 1$ : $L$ constructs $y \oplus s = ((y \oplus s)_1, \cdots, (y \oplus s)_N)$ and reveals $y \oplus s$ and $\Pi = \Sigma \circ \sigma$.
  - if $b = 2$ : $L$ constructs and reveals $\Pi(y)$ and $\Pi(s)$.

◂ Return

# Generalized Stern (Codes)

4) Verification Step:
   - if $b = 0 : V$ verifies that $\Pi(s)$ is a $n$-block permutation and that $C_1, C_2$ have been honestly calculated.
   - if $b = 1 : V$ verifies that $\Pi(s)$ is a $n$-block permutation and that $C_1, C_3$ have been honestly calculated.
   - if $b = 2 : V$ verifies that $C_2, C_3$ have been honestly calculated, and that the weight of $\Pi(s)$ is $t\omega$ and that $\Pi(s)$ is formed of $N$ blocks of length $n$ and of weight $\omega$ or 0.
5) Iterate the steps 1,2,3,4 until the expected security level is reached.

◂ Return

# Lattice Concepts

## Short Integer Solution - SIS

Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a prime number $q$, find a vector $\mathbf{v}$ in the lattice $\Lambda_q^{\perp} = \{\mathbf{x} \in \mathbb{Z}^m \ : \ \mathbf{Ax} = \mathbf{0} \bmod q\}$ with length limited by $\|\mathbf{v}\| \leq L$.

Return

# Lattice Concepts

## Short Integer Solution - SIS

Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a prime number $q$, find a vector $\mathbf{v}$ in the lattice $\Lambda_q^{\perp} = \{\mathbf{x} \in \mathbb{Z}^m \ : \ \mathbf{Ax} = \mathbf{0} \bmod q\}$ with length limited by $\|\mathbf{v}\| \leq L$.

Return

# Stern

## Security Assumptions

- Existence of a collision resistant hash function $h$.
- Hardness of Syndrome-Decoding Problem: Is there $s = He^T$, with $\|e\| \leq w$?

## Zero-Knowledge Interactive Proof System

- Perfect completeness.
- Soundness: cheater can succeed in a given round with at most 2/3 probability.
- Zero-Knowledge: prover convinces verifier that he knows $e$, without revealing its value.

◂ Return

# Stern

## Commitments

- $c_1 \leftarrow \mathrm{h}(\sigma || Hy^T)$, where $y \xleftarrow{\$} \mathbb{F}_2^n$ and $\sigma$ is a random permutation.
- $c_2 \leftarrow \mathrm{h}(y \cdot \sigma)$.
- $c_3 \leftarrow \mathrm{h}((y \oplus sk) \cdot \sigma)$, where $pk \leftarrow H(sk)^T$

## Challenge

- $ch \xleftarrow{\$} \{0, 1, 2\}$

## Answer

- If $ch = 0$ : P reveals $y$ and $\sigma$. Then V checks $c_1$ and $c_2$.
- If $ch = 1$ : P $y \oplus sk$ and $\sigma$. Then V checks $c_1$ and $c_3$
- If $ch = 2$ : P reveals $y \cdot \sigma$ and $sk \cdot \sigma$. Then V checks $c_2$, $c_3$ and weight of $sk$.

◂ Return

# Véron's Improvement of Stern Scheme

## Main Ideas

- Dual construction. Generator matrix $G \leftrightarrow$ Check matrix $H$.
- Is there $(m, e)$ such that $x = mG + e$, with $wt(e) = p$?
- Better transmission rate.
- Techniques from finite fields to lower complexity of prover's task and size of stored data. Improve product by choosing a basis $\mathbb{F}_{2^k}$ that results in more sparse matrices.

Return

# Cayrel and Véron's 5-pass Scheme

Prover

$c_1 \leftarrow h(\Sigma, \gamma, Hu^T)$
$c_2 \leftarrow h(\Pi_{\gamma,\Sigma}(u), \Pi_{\gamma,\Sigma}(s))$
**send** $\{c_1, c_2\}$

$\beta \leftarrow \Pi_{\gamma,\Sigma}(u + \alpha s)$
**send** $\beta$

When 0, **reveals** $\gamma, \Sigma$
When 1, **reveals** $\Pi_{\gamma,\Sigma}(s)$

Where $\Pi_{\gamma,\Sigma}(u) = (\gamma_{\Sigma(1)} u_{\Sigma(1)}, \ldots, \gamma_{\Sigma(n)} u_{\Sigma(n)})$
and $y = Hs^T$

Verifier

$\alpha \xleftarrow{\$} \mathbb{F}_q^n$
**send** $\alpha$

**send** challenge in $\{0, 1\}$

0: check $c_1 = h(\Sigma, \gamma, H\Pi^{-1}_{\gamma,\Sigma}(\beta)^T - \alpha y)$
1: check $c_2 = h(\beta - \alpha\Pi_{\gamma,\Sigma}(s), ), \Pi_{\gamma,\Sigma}(s))$
and $wt(\Pi_{\gamma,\Sigma}(s)) = w$

◂ Return

# Computer evolution: 1970 to 2005

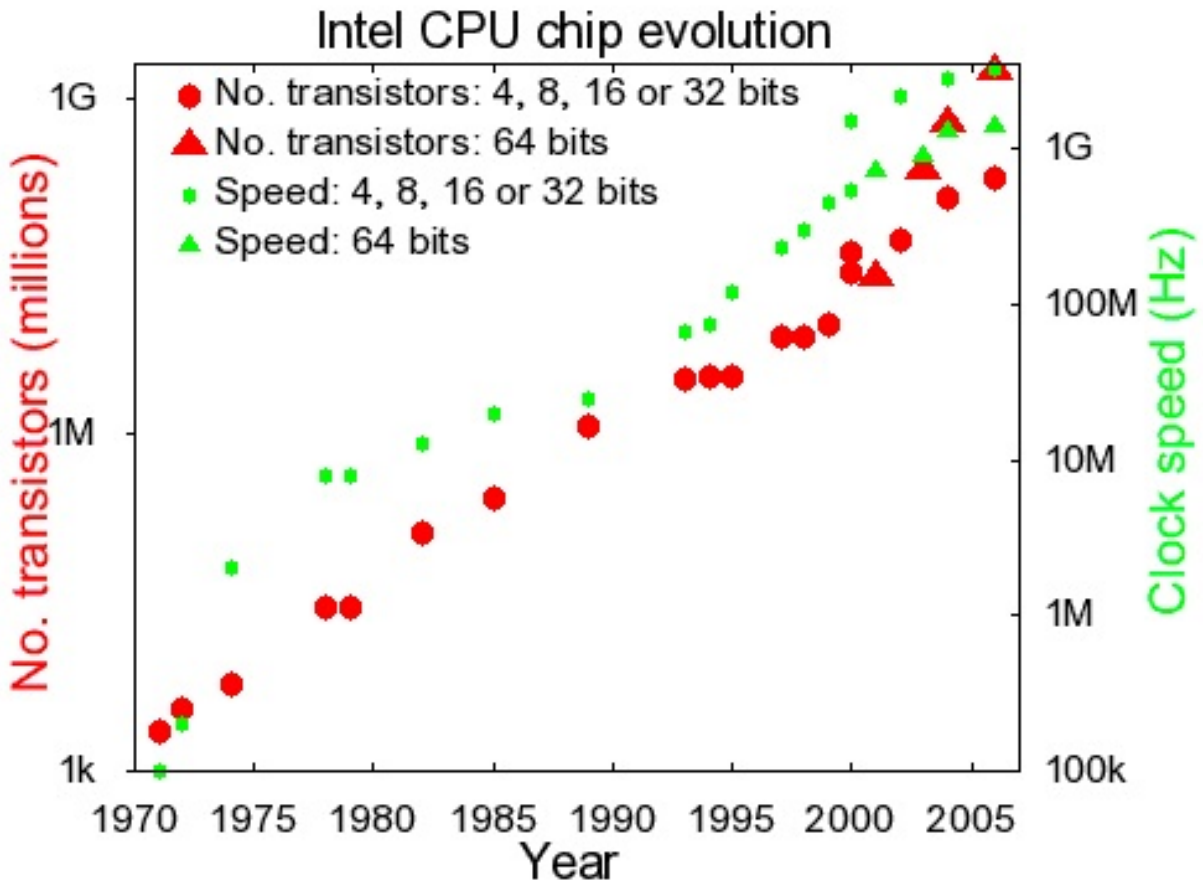



**Figure:** CPU Evolution - log scale

Return

# Computer evolution: 1970 to 2005

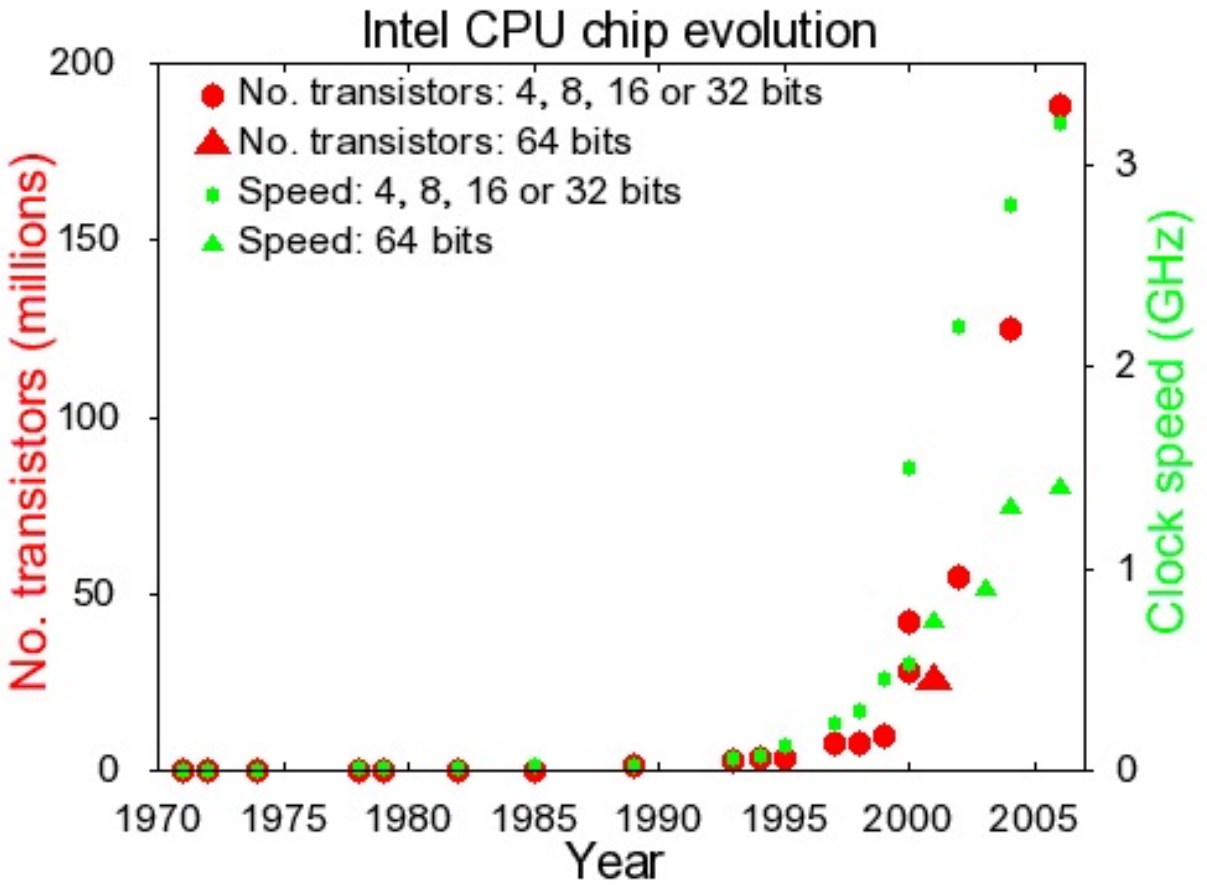



**Figure:** CPU Evolution

Return

## Worst-case to average-case

### Theorem

*For any polynomially bounded functions* $\beta(n), m(n), q(n) = n^{O(1)}$, *with* $q(n) \geq 4\sqrt{m(n)}n^{1.5}\beta(n)$ *and* $\gamma(n) = 14\pi\sqrt{n}\beta(n)$, *there is a probabilistic polynomial time reduction from solving* $GapCVP_\gamma$ *in the worst-case to solving* $SIS_{q,m,\gamma}$ *on the average with non-negligible probability. In particular, for any* $m = \Theta(n \log n)$, *there exists* $q(n) = O(n^{2.5} \log n)$ *and* $\gamma = O(n\sqrt{\log n})$, *such that solving* $SIS_{q,m}$ *on the average is at least as hard as solving* $GapSVP_\gamma$ *in the worst-case.*

◂ Return


Miklós Ajtai.
Generating hard instances of lattice problems.
*Electronic Colloquium on Computational Complexity (ECCC)*, 3(7), 1996.


Miklós Ajtai and Cynthia Dwork.
A public-key cryptosystem with worst-case/average-case equivalence.
*Electronic Colloquium on Computational Complexity (ECCC)*, 3(65), 1996.


John Horton Conway and Neil James Alexander Sloane.
*Sphere packings, lattices and groups*.
Springer, New York, 1988.


Uriel Feige, Amos Fiat, and Adi Shamir.
Zero knowledge proofs of identity.
In *STOC*, pages 210–217. ACM, 1987.


Amos Fiat and Adi Shamir.
How to prove yourself: Practical solutions to identification and signature problems.
In *CRYPTO*, pages 186–194, 1986.


Ishay Haviv and Oded Regev.
Tensor-based hardness of the shortest vector problem to within almost polynomial factors.
In *STOC*, pages 469–477, 2007.


Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa.
Concurrently secure identification schemes based on the worst-case hardness of lattice problems.
In *ASIACRYPT '08: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security*, pages 372–389, Berlin, Heidelberg, 2008. Springer-Verlag.


Vadim Lyubashevsky.

Lattice-based identification schemes secure under active attacks.
In *Public Key Cryptography*, pages 162–179, 2008.


Vadim Lyubashevsky.
Fiat-shamir with aborts: Applications to lattice and factoring-based signatures.
In Mitsuru Matsui, editor, *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science*, pages 598–616. Springer, 2009.


Vadim Lyubashevsky and Daniele Micciancio.
Generalized compact knapsacks are collision resistant.
In *ICALP (2)*, pages 144–155, 2006.


Vadim Lyubashevsky and Daniele Micciancio.
Asymptotically efficient lattice-based digital signatures.

In *Theory of Cryptography Conference (TCC 2008)*, pages 37–54. Springer Berlin / Heidelberg, LNCS 4948, March 2008.


D. Micciancio.
Generalized compact knapsacks, cyclic lattices, and efficient one-way functions.
In *Computational Complexity*. Springer, 2007.


Daniele Micciancio and Shafi Goldwasser.
*Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*.
Kluwer Academic Publishers, Boston, Massachusetts, March 2002.


Daniele Micciancio and Salil P. Vadhan.
Statistical zero-knowledge proofs with efficient provers: Lattice problems and more.

In Dan Boneh, editor, *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 282–298. Springer, 2003.


Minh-Huyen Nguyen and Salil P. Vadhan.
Zero knowledge with efficient provers.
In *STOC*, pages 287–295, 2006.


Peter W. Shor.
Polynominal time algorithms for discrete logarithms and factoring on a quantum computer.
In Leonard M. Adleman and Ming-Deh A. Huang, editors, *ANTS*, volume 877 of *Lecture Notes in Computer Science*, page 289. Springer, 1994.


Jacques Stern.
A new identification scheme based on syndrome decoding, 1994.


Zaharina Velikova.

*A Lattice Attack on the McEliece Public Key Cryptosystem: Lattice Basis Reduction Algorithms in Cryptography*.
VDM Verlag, 2008.

Pascal Véron.
Improved identification schemes based on error-correcting codes.
*Appl. Algebra Eng. Commun. Comput.*, 8(1):57–69, 1996.