Math-for-industry
Education & Research Hub

ISSN2188-1200

IMI Workshop of the Joint Research Projects

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

Editors: Kirill Morozov, Hiroaki Anada, Yuji Suga

九州大学マス･フォア･インダストリ研究所

MI Lecture Note Vol.80 : Kyushu University

IMI Workshop of the Joint Research Projects

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

Editors: Kirill Morozov, Hiroaki Anada, Yuji Suga

About MI Lecture Note Series

The Math-for-Industry (MI) Lecture Note Series is the successor to the COE Lecture Notes, which were published for the 21st COE Program “Development of Dynamic Mathematics with High Functionality,” sponsored by Japan’s Ministry of Education, Culture, Sports, Science and Technology (MEXT) from 2003 to 2007. The MI Lecture Note Series has published the notes of lectures organized under the following two programs: “Training Program for Ph.D. and New Master’s Degree in Mathematics as Required by Industry,” adopted as a Support Program for Improving Graduate School Education by MEXT from 2007 to 2009; and “Education-and-Research Hub for Mathematics-for-Industry,” adopted as a Global COE Program by MEXT from 2008 to 2012.

In accordance with the establishment of the Institute of Mathematics for Industry (IMI) in April 2011 and the authorization of IMI’s Joint Research Center for Advanced and Fundamental Mathematics-for-Industry as a MEXT Joint Usage / Research Center in April 2013, hereafter the MI Lecture Notes Series will publish lecture notes and proceedings by worldwide researchers of MI to contribute to the development of MI.

October 2014
Yasuhide Fukumoto
Director
Institute of Mathematics for Industry

IMI Workshop of the Joint Research Projects
Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

MI Lecture Note Vol.80, Institute of Mathematics for Industry, Kyushu University
ISSN 2188-1200
Date of issue: March 30, 2018
Editors: Kirill Morozov, Hiroaki Anada, Yuji Suga
Publisher:
Institute of Mathematics for Industry, Kyushu University
Graduate School of Mathematics, Kyushu University
Motooka 744, Nishi-ku, Fukuoka, 819-0395, JAPAN
Tel +81-(0)92-802-4402, Fax +81-(0)92-802-4405
URL http://www.imi.kyushu-u.ac.jp/

Printed by
Kijima Printing, Inc.
Shirogane 2-9-6, Chuo-ku, Fukuoka, 810-0012, Japan
TEL +81-(0)92-531-7102 FAX +81-(0)92-524-4411

IMI Workshop of the Joint Research Projects

Workshop on Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

June 12th – 13th, 2017

Industry-University-Government Collaboration Innovation Plaza

3-8-34 Momochihama Sawara-ku Fukuoka 814-0001, Japan

Sponsored by

Institute of Mathematics for Industry (IMI),

Kyushu University

Organized by

Kirill Morozov, Hiroaki Anada, and Yuji Suga

Acknowledgements

One of the organizers, Kirill Morozov, was partially supported by a *kakenhi* Grant-in-Aid for Scientific Research (C) 15K00186 from Japan Society for the Promotion of Science concerning his invitation of Prof. Beimel and Prof. Desmedt, as well as his own participation to this workshop.

One of the organizers, Hiroaki Anada, was partially supported by a kakenhi Grant-in-Aid for Scientific Research (C) 15K00029 from Japan Society for the Promotion of Science concerning his invitation of Prof. Kushilevitz to this workshop.

Preface

Rapid development of computer systems and networks emphasized importance of application of cryptographic technologies. Confidentiality and reliability can be naturally attained using the cryptographic technology of secret-sharing, which has been more and more widely applied for secure storage. However, data must not only be securely stored but also securely processed, and therefore search and computation over secured data becomes an increasingly important problem that finds applications in digital payment systems, medical data processing, and other important areas – these functionalities are achieved using secure multi-party computation technologies. Acceptance of these concepts for practical deployment requires a thorough security evaluation, involving mathematical modeling of the implemented systems as well as their rigorous security proofs. The purpose of this workshop was to discuss the above aspects. The program included 3 keynote lectures, 6 invited lectures and a panel discussion, gathering over 40 attendees in total. The goal of these lecture notes is to raise awareness about the topics and results discussed at the workshop, especially among researchers in mathematics and developers in cloud computing and cybersecurity.

Kirill Morozov, Representative of the Organizers

Table 1. List of attendees.

Hiroaki Anada	Tushar Kanti Saha	Shinichi Matsumoto	Nobuyuki Sugio
Amos Beimel	Ryo Kikuchi	Tomoko Matsushima	Yasushi Takahashi
Bernardo David	Dong-In Kim	Toshiyasu Matsushima	Tadanori Teruya
Yvo Desmedt	Eitaro Kohno	Shota Nakasato	Junting Xiao
Tsumbuukhuu Dulguun	Takeshi Koshiba	Naohisa Nishida	Masato Yamanouchi
Goichiro Hanaoka	Noboru Kunihiro	Koji Nuida	Masaya Yasuda
Keisuke Hara	Naruhiro Kurokawa	Kazuma Ohara	Kenji Yasunaga
Masahiro Ishii	Eyal Kushilevitz	Kazuo Ohta	Maki Yoshida
Makoto Ishikawa	Hyungu Lee	Miyo Okada	Yusuke Yoshida
Mitsugu Iwamoto	Shincheol Lee	Eriko Osakabe	Ye Yuan
Hyungrok Jo	Niklas Lemcke	Yuji Suga	Kirill Morozov

Photograph 1. Group photo in front of the venue.

KYUSHU UNIVERSITY

IMI Joint Research Project in 2017

Institute of Mathematics for Industry Kyushu University

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

Date:

June 12(Mon)-13(Tue), 2017

http://www.imi.kyushu-u.ac.jp/eng/events/view/1240

Keynote speakers:

Amos Beimel, Ben-Gurion University
"Graph Secret Sharing"

Yvo Desmedt, The University of Texas at Dallas
"Human Recomputable Secret Shares and their Applications in E-Voting"

Eyal Kushilevitz, Technion
"Ad-hoc MPC"

Invited speakers:

Bernardo David, Tokyo Institute of Technology
Mitsugu Iwamoto, The University of Electro-Communications
Ryo Kikuchi, NIPPON TELEGRAPH AND TELEPHONE CORPORATION
Takeshi Koshiba, Waseda University
Naruhiro Kurokawa, Bank of Japan
Kazuma Ohara, NEC Corporation

Venue: AirIMaQ (Momochi), Seminar Room, 2F
Industry-University-Government Collaboration Innovation Plaza
3-8-34 Momochihama Sawara-ku Fukuoka 814-0001, JAPAN
https://airimaq.kyushu-u.ac.jp/en/airimaq/access.php

■Organizing Committee ▸
Hiroaki Anada (University of Nagasaki)
Kirill Morozov (Tokyo Institute of Technology)
Yuji Suga (Internet Initiative Japan Inc.)

■Sponsored by ▸ Institute of Mathematics for Industry, Kyushu University
■Registration fee ▸ Free

Contact : ct-sns-info@imi.kyushu-u.ac.jp
(For general inquiries) Institute of Mathematics for Industry, Kyushu University
TEL: 092-802-4402 E-mail: kyodo_riyou@imi.kyushu-u.ac.jp

Program

June 12 (Monday)

10:00-10:10 (Opening)

[1] 10:10-10:50 [keynote] Amos Beimel, Ben-Gurion University, Israel
"Graph Secret Sharing"

[2] 11:10-11:50 [keynote] Yvo Desmedt, The University of Texas at Dallas, USA
"Human Recomputable Secret Shares and their Applications in E-Voting"

[3] 14:00-14:40 Mitsugu Iwamoto, The University of Electro-Communications, Japan
"Secret Sharing Schemes under Guessing Secrecy"

[4] 15:00-15:40 Naruhiro Kurokawa, Bank of Japan, Japan
"Function Secret Sharing Using Fourier Basis"

16:00-16:30 (Panel Discussion) Panelists: Bernardo David, Yvo Desmedt, Mitsugu Iwamoto, Ryo Kikuchi, Naruhiro Kurokawa, Eyal Kushilevitz and Kazuma Ohara.
Moderator: Kirill Morozov

June 13 (Tuesday)

[5] 10:10-10:50 [keynote] Eyal Kushilevitz, Technion, Israel
"Ad-hoc MPC"

[6] 11:10-11:50 Takeshi Koshiba, Waseda University, Japan
"Secure Message Transmission against Rational Adversaries"

[7] 14:00-14:40 Kazuma Ohara, NEC Corporation, Japan
"Optimized Honest-Majority MPC for Malicious Adversaries
- Breaking the 1 Billion-Gate Per Second Barrier"

[8] 14:50-15:30 Ryo Kikuchi, NTT CORPORATION, Japan
"Key components in MEVAL"

[9] 15:40-16:20 Bernardo David, Tokyo Institute of Technology, Japan
"A Provably Secure Proof-of-Stake Blockchain Protocol"

16:20-16:30 (Closing)

Table of Contents

Linear Secret-Sharing Schemes for Forbidden Graph Access Structures

Amos Beimel (Joint work with Oriol Farràs, Yuval Mintz, and Naty Peter)

Ben Gurion University of the Negev
amos.beimel@gmail.com

A secret-sharing scheme realizes the forbidden graph access structure determined by a graph $G = (V, E)$ if a pair of vertices can reconstruct the secret if and only if it is and edge of G. An important property of these schemes is that they can be used to construct schemes for the conditional disclosure of secrets.

We study the complexity of realizing a forbidden graph access structure by linear secret-sharing schemes. A secret-sharing is linear if the reconstruction of the secret from the shares is a linear mapping. In many applications of secret sharing, it is required that the scheme is linear. We provide efficient constructions and lower bounds on the share size of linear secret-sharing schemes for sparse and dense graphs, closing the gap between upper and lower bounds: Given a sparse graph with n vertices and at most $n^{1+\beta}$ edges, for some $0 \leq \beta < 1$, we construct a linear secret-sharing scheme realizing the forbidden graph access structure in which the total size of the shares is $\tilde{O}(n^{1+\beta/2})$. We provide an additional construction showing that every dense graph with n vertices and at least $\binom{n}{2} - n^{1+\beta}$ edges can be realized by a linear secret-sharing scheme with the same share size.

We prove lower bounds on the share size of linear secret-sharing schemes realizing forbidden graph access structures. We prove that for most forbidden graphs access structures, the total share size of every linear secret-sharing scheme realizing the graph is $\Omega(n^{3/2})$, this shows that construction of [Gay, Kerenidis, and Wee, CRYPTO 2015] is optimal. Furthermore, we show that for every $0 < \beta \leq 1$ there exist a graph with at most $n^{1+\beta}$ edges and a graph with at least $\binom{n}{2} - n^{1+\beta}$ edges, such that the total share size of every linear secret-sharing scheme realizing these forbidden graph access structures is $\Omega(n^{1+\beta/2})$. This shows that our constructions are optimal (up to poly-logarithmic factors).

Secret Sharing
for Forbidden Graphs

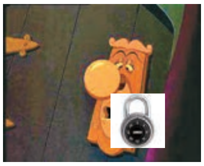
Amos Beimel, Ben-Gurion University

Based on works with
Oriol Farras, Universitat Rovira i Virgili
Yuval Mintz, Naty Peter, Ben-Gurion University

Cryptographic Technologies for Securing Network Storage
June 12, 2017

1

Secret Sharing



3742

6634 3441 2538 1329




Secret Sharing

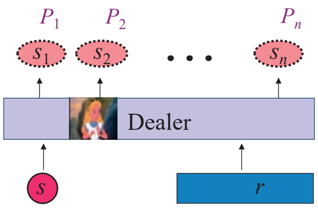
3742


6634 3441 2538 1329




3

Secret Sharing [Shamir79,Blakley79,ItoSaitoNishizeki87]



- Parties: $P = \{P_1, \ldots, P_n\}$
- Access Structure $\Gamma \subseteq 2^P$ (collection of sets of parties)
- A scheme realizes Γ if:
  - Correctness: every authorized set $B \in \Gamma$ can recover s
  - Privacy: every unauthorized set $B \notin \Gamma$ cannot learn anything about s

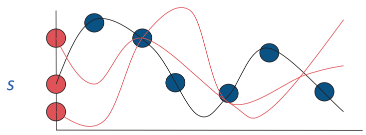
4

Shamir's t-out-of-n Secret Sharing

- Access structure: $\Gamma = \{ A \subseteq P : |A| \geq t \}$
- Scheme:
  - Input: secret $s \in \mathbb{F}_p$ where $p > n$ is a prime
  - Dealer chooses a random polynomial

$$Q(x) = s + r_1 x + r_2 x^2 + \cdots + r_{t-1} x^{t-1}$$

  - Share of P_j: $s_j = Q(j) \mod p$



5

Linear Secret Sharing

- Input: secret $s \in \mathbb{F}_q$
- Dealer chooses random elements $r_1, \ldots, r_m \in \mathbb{F}_q$
- Share :
  - A vector over $\mathbb{F}_q$
  - Each coordinate: a linear combination of s and $r_1, \ldots, r_m$

- Example 1: Shamir's scheme:
  - $s_j = Q(j) = s + j^1 \cdot r_1 + j^2 \cdot r_2 + \cdots + j^{t-1} \cdot r_{t-1} \bmod p$
- Example 2: $s \in \mathbb{F}_2$
  - Dealer chooses $r_1, r_2 \in \mathbb{F}_2$
  - $s_1 = (r_1, r_1 \oplus r_2)$
  - $s_2 = (s \oplus r_1)$
  - $s_3 = (r_1, s \oplus r_1 \oplus r_2)$

6

Why Secret Sharing?

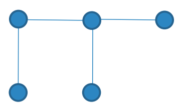
- Storing sensitive information – Robust key management
- Used in many secure protocols:
  - multiparty computation
  - threshold cryptography
  - attribute-based encryption (ABE)
  - access control
  - oblivious transfer
- Most applications require linear secret-sharing schemes
- Most known schemes are linear

7

Schemes for Forbidden Graphs [SunShieh97]

A scheme realizes a forbidden graph $G = (V, E)$ if:

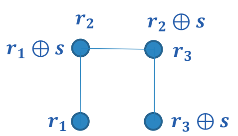
- The parties are the set of vertices V
- The authorized sets are:
  - The edges in E
  - Every set of size at least 3
- The unauthorized sets are:
  - The non-edges
  - A single party (vertex)



8

A Scheme Realizing a Forbidden Graph

- $s \in \{0, 1\}$
- For every edge $e_i = (u, v) \in E$,
  - Give a random bit r_i to u and $r_i \oplus s$ to v
- u, v can reconstruct the secret by performing xor on their shares.



- In addition, share s using a 3-out-of-n secret-sharing scheme
- Total share size: $O(|V| + |E|) = O(n^2)$

9

Upper Bounds for Forbidden Graphs

- Every graph can be realized by a secret-sharing scheme with share size $n^{1+\sqrt{\log\log n/\log n}} = n^{1+o(1)}$ [LiuVaikuntanathanWee17]
- Every graph can be realized by a *linear* secret-sharing scheme with share size $O(n^{3/2})$ [GayKerenidisWee15]

- We consider linear secret sharing schemes
- Questions:
  - If G contains few edges, can we realize it more efficiently?
    - Few = $n^{1+\beta}$. Goal: better than $\min\{n^{1+\beta}, n^{3/2}\}$
  - If G contains many edges, can we realize it more efficiently?
    - Many = $\binom{n}{2} - n^{1+\beta}$ **.** Goal: better than $n^{3/2}$
  - If G has an efficient scheme and we add and remove few edges, can we realize it efficiently?

10

Motivation

- Secret sharing for forbidden bipartite graphs are equivalent to conditional disclosure of secrets
  - Used to construct symmetric private information retrieval and attribute based encryption

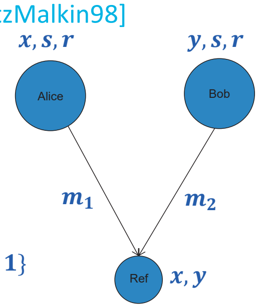
- Our goal: construct efficient linear secret-sharing schemes for specific families of forbidden graphs

- We want to understand if, for forbidden graphs, linear secret sharing requires shares of size $\Omega(n^{3/2})$
  - Which graphs require large shares?

11

Conditional Disclosure of Secrets (CDS)
[GertnerIshaiKushilevitzMalkin98]

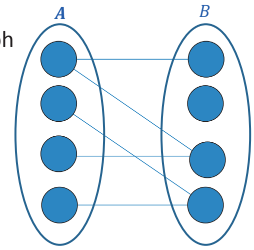
- Each party has a private input
- Both parties know a secret $\boldsymbol{s}$
- Shared randomness $\boldsymbol{r}$
- Referee knows $\boldsymbol{x}, \boldsymbol{y}$
- A condition: $\boldsymbol{P}\colon \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{N}} \times \{\mathbf{0},\mathbf{1}\}^{\boldsymbol{N}} \to \{\mathbf{0},\mathbf{1}\}$
- Each party sends one message
- Correctness: If $\boldsymbol{P}(\boldsymbol{x},\boldsymbol{y}) = \mathbf{1}$, Ref learns $\boldsymbol{s}$
- Security: If $\boldsymbol{P}(\boldsymbol{x},\boldsymbol{y}) = \mathbf{0}$, Ref learns nothing



Learns $\boldsymbol{s}$ iff $\boldsymbol{P}(\boldsymbol{x},\boldsymbol{y}) = \mathbf{1}$

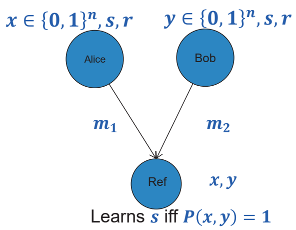
CDS and Forbidden Bipartite Secret Sharing

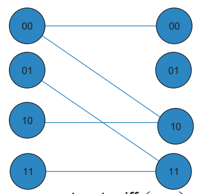
- Bipartite Graph: $\boldsymbol{G = (A, B, E)}$
  - Vertices: $\boldsymbol{A \cup B}$
  - Edges: Only between sets $\boldsymbol{E \subseteq A \times B}$
- Secret sharing for forbidden bipartite graph
  - Every $\boldsymbol{(a, b) \in E}$ can reconstruct $\boldsymbol{s}$
  - Every $\boldsymbol{a \in A, b \in B}$ s.t. $\boldsymbol{(a, b) \notin E}$ should not learn information about $\boldsymbol{s}$



CDS and Forbidden Bipartite Secret Sharing

- Given a CDS define:
  - $\boldsymbol{A, B = \{0,1\}^n}$
  - $\boldsymbol{E = \{(x, y): P(x, y) = 1\}}$
- To share a secret s:
  - $\boldsymbol{s_x = m_1(x, s, r), s_y = m_2(y, s, r)}$
- $\boldsymbol{x, y}$ can reconstruct $\boldsymbol{s}$ iff $\boldsymbol{P(x, y) = 1}$ iff $\boldsymbol{(x, y) \in E}$





$\boldsymbol{x, y}$ can reconstruct $\boldsymbol{s}$ iff $\boldsymbol{(x, y) \in E}$

Main Result: Upper Bounds

Thm 1:

If a graph with n vertices contains for some $0 \leq \beta \leq \mathbf{1}$
- either at most $n^{1+\beta}$ edges or
- at least $\binom{n}{2} - n^{1+\beta}$ edges,

Then there is a linear secret-sharing scheme realizing the graph with total share size $\tilde{O}(n^{1+\beta/2})$.

Thm 2:

If
- G can be realized with a scheme with total share size m.
- G' obtained from G by removing and adding at most $n^{1+\beta}$ edges.

Then there is a linear secret-sharing scheme realizing G' with share size $\tilde{O}(m + n^{1+\beta/2})$.

15

Main Result: Lower Bounds

- Thm 3: There exists a graph with n vertices such that in any linear secret-sharing scheme realizing it with a one-bit secret the size of the shares is $\Omega(n^{3/2})$
- Conclusion 1: The construction of Gay et al. is optimal
- Conclusion 2: Gap between linear and non-linear schemes for forbidden graphs

- Thm 4: There exists a graph with n vertices and at most $n^{1+\beta}$ edges such that in any linear secret-sharing scheme realizing it with a one-bit secret the size of the shares is $\Omega(n^{1+\beta/2})$
  - Same result for a graph with at least $\binom{n}{2} - n^{1+\beta}$ edges

- Conclusion 3: Our constructions are optimal up to a poly-log factor.

16

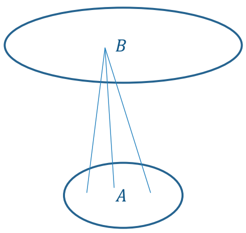
A Scheme for a Graph with $n^{1+\beta}$ Edges

- Basic Construction: for a bipartite graph $G = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{E})$ such that $\boldsymbol{A}$ is small and every vertex in $\boldsymbol{B}$ has degree at most $\boldsymbol{d}$
- Share size $\boldsymbol{O}(|\boldsymbol{B}| + |\boldsymbol{A}| \cdot \boldsymbol{d})$
- Second construction: for a bipartite $\boldsymbol{G} = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{E})$ such that every vertex in $\boldsymbol{B}$ has degree at most $\boldsymbol{d}$
  - Share size $\boldsymbol{O}(\boldsymbol{n} \cdot \sqrt{\boldsymbol{d}})$
- Third construction: for a bipartite graph $\boldsymbol{G} = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{E})$ that has at most $\boldsymbol{n}^{1+\beta}$ edges
  - Share size $\boldsymbol{O}(\boldsymbol{n}^{1+\beta/2})$
- Final construction: for a graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ that has at most $\boldsymbol{n}^{1+\beta}$ edges
  - Share size $\boldsymbol{O}(\boldsymbol{n}^{1+\beta/2})$

17

Basic Construction

- If $\boldsymbol{G} = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{E})$ is bipartite graph s.t. every vertex in $\boldsymbol{B}$ has degree at most $\boldsymbol{d}$
- Then $\boldsymbol{G}$ has a linear secret-sharing with total share size is $\boldsymbol{O}(|\boldsymbol{B}| + |\boldsymbol{A}| \cdot \boldsymbol{d})$



Example: $|\boldsymbol{A}| = \sqrt{\boldsymbol{n}}, |\boldsymbol{B}| = \boldsymbol{n}$

⇨ Every $\boldsymbol{b} \in \boldsymbol{B}$ has degree at most $\boldsymbol{d} = \sqrt{\boldsymbol{n}}$

⇨ The total share size is $\boldsymbol{O}(\boldsymbol{n})$

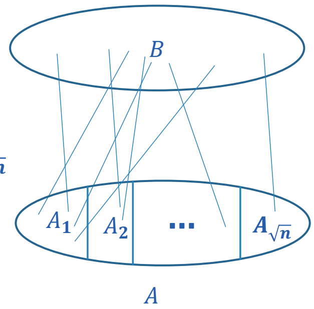
18

A Scheme with share size $O(n^{3/2})$

- If $\boldsymbol{G = (A, B, E)}$ is bipartite graph
- Then $\boldsymbol{G}$ has a linear secret-sharing with total share size is $\boldsymbol{O(n^{3/2})}$

Scheme:

- Partition $\boldsymbol{A}$ into sets $\boldsymbol{A_1, \dots, A_{\sqrt{n}}}$ of size $\boldsymbol{\sqrt{n}}$
- Define $\boldsymbol{G_i = (A_i, B, E \cap (A_i \times B))}$
- Realize each $\boldsymbol{G_i}$ with a scheme with total share size $\boldsymbol{O(n)}$
- The total share size is $\boldsymbol{O(n \cdot \sqrt{n}\,)}$



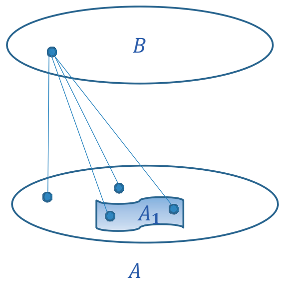
21

A Scheme with share size $O(nd^{1/2})$

- If $G = (\boldsymbol{A, B, E})$ is bipartite graph s.t.
  - The degree of every $\boldsymbol{b \in B}$ is at most $\boldsymbol{d}$
- Then $\boldsymbol{G}$ has a linear secret-sharing with total share size is $O(\boldsymbol{nd^{1/2}})$

With different parameters :

- Randomly partition $\boldsymbol{A}$ into: $\boldsymbol{A_1, \dots, A_{\sqrt{d}}}$ of size $n/\sqrt{d}$
- Define $\boldsymbol{G_i = (A_i, B, E \cap (A_i \times B))}$
  - With high prob. the degree of every $\boldsymbol{b \in B}$ in $\boldsymbol{G_i}$ is at most $\boldsymbol{\sqrt{d}}$
- Realize each $\boldsymbol{G_i}$ with a scheme with total share size $\boldsymbol{O\left(n + (n/\sqrt{d}) \cdot \sqrt{d}\right) = O(n)}$
- The total share size is $\boldsymbol{O(n \cdot \sqrt{d})}$



22

Bipartite with Few Edges

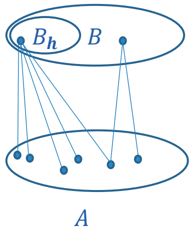
If $\boldsymbol{G = (A, B, E)}$ is bipartite with at most $\boldsymbol{O(n^{1+\beta})}$ edges

Then $\boldsymbol{G}$ has a linear secret-sharing with total share size is $\boldsymbol{O(n^{1+\beta/2})}$

- In this talk: $O(\boldsymbol{n^{5/4+\beta/4}})$

Scheme:

- Let $\boldsymbol{B_h = \{b \in B : \deg(b) > n^{1/2+\beta/2}\}}$
- $\boldsymbol{|B_h| \le \frac{n^{1+\beta}}{n^{1/2+\beta/2}} = n^{1/2+\beta/2}}$
- Realize $\boldsymbol{G_{\text{high}} = (A, B_h, E \cap (A \times B_h))}$
  - Share size $\boldsymbol{O\left(\sqrt{|A| \cdot |B_h| \cdot n}\right) = O\left(\sqrt{n \cdot n^{1/2+\beta/2} \cdot n}\right)}$
- Realize $\boldsymbol{G_{\text{low}} = (A, B \setminus B_h, E \cap (A \times B \setminus B_h))}$
  - Share size $\boldsymbol{O\left(\sqrt{|A| \cdot |B| \cdot n^{1/2+\beta/2}}\right) = O\left(\sqrt{n \cdot n \cdot n^{1/2+\beta/2}}\right)}$
- In the paper: Reduce degree in $\boldsymbol{\log\ n}$ steps



23

Conclusions

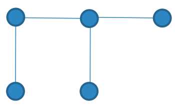
- Forbidden graph secret sharing is equivalent to CDS ⇨ SPIR, Atribute based encryption
- Every forbidden graph can be realized by a linear secret-sharing scheme with share size $\boldsymbol{O}(n^{1.5})$.
- We show that every forbidden graph with $n^{1+\beta}$ edges can be realized by a linear secret-sharing scheme with share size $\boldsymbol{O}(n^{1+\beta/2})$.
  - Same result for with $\binom{n}{2} - n^{1+\beta}$ edges
- There exists a forbidden graph such that in any linear secret-sharing scheme realizing it the share size is $\Omega(n^{1.5})$
- There exists a forbidden graph with $n^{1+\beta}$ edges such that in any linear secret-sharing scheme realizing it the share size is $\Omega(n^{1+\beta/2})$
- Open: graph access structures

24

Schemes for Graphs

A scheme realizes a graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ if:

- The parties are the set of vertices V
- The authorized sets are:
  - The edges in $\boldsymbol{E}$
  - Every set that contains an edge
- The unauthorized sets are:
  - The non-edges
  - Every set that doesn't contain an edge
- Every graph can be realized by a linear scheme with share size $\boldsymbol{O(n^2/\log n)}$
  - Sparse graph: $\boldsymbol{E}$
  - Dense graph: $\boldsymbol{O(n^{5/4+3\beta/4})}$



25

Thanks!

26

Human Recomputable Secret Shares and their Applications in E-Voting

Yvo Desmedt

The University of Texas at Dallas

Yvo.Desmedt@utdallas.edu

The classical approach of secret sharing is to consider the secret to be in a finite field. Computers are used by the dealer to make shares, and computers are used to reconstruct the secret. Since the invention of Visual Cryptography by Kafri and Keren in 1987, many researchers have stepped away from these restrictions.

In 2007, Desmedt-Pieprzyk-Steinfeld-Wang considered secrets that belong to a non-Abelian group, such as the symmetric group (i.e., permutations), to obtain secure multiparty computation.

In this talk, we consider secret and shares that are permutations, wonder how good humans can do computations with these and consider them in the context of e-voting, but then e-voting secure against hacking of the voter's computer.

Human Recomputable Secret Shares and their Applications in E-Voting

Yvo Desmedt

Univ. of Texas at Dallas, US

June 12, 2017

Yvo Desmedt's work on anonymity was partially supported by: the US NSF ANI-0087641. The work on voting was partially sponsored by the UK EPSRC EP/C538285/1, by BT as BT Chair of Information Security and partly done while being Invited Senior Research Scientist at RCIS (AIST, Japan).

A part of this research was done while Yvo Desmedt visited AT&T Shannon Research, Tsinghua University (while funded by the National Natural Science Foundation of China Grant 60553001, and the National Basic Research Program of China Grant 2007CB807900 and 2007CB807901).

Part of this presentation is based on:

- unpublished research with Rebecca Wright (with her permission),
- a joint paper with Josef Pieprzyk, Ron Steinfeld and Huaxiong Wang (Crypto 2007)

- a joint paper with Stelios Erotokritou at SCN 2012.
- a joint paper with Stelios Erotokritou at Vote ID 2015.

Special thanks to Rene Peralta whose November 9, 2011 suggestion to consider $Z_{10}(+)$ as an Abelian subgroup of S_{10}, allowed us to make a more user-friendly scheme.

Overview

1. Special Secret Sharing Schemes
2. Our setting: Post Snowden elections
3. A pioneering approach: Chaum's Code Voting
4. Advantages/disadvantages of Code Voting
5. Our setting, assumptions and their impacts
6. The voting: passive adversary only
7. Some usability tests (SCN 2012)
8. High level description
9. Details: technical background
10. The mixing for the single-seat: Efficiency improvement
11. The mixing for the single-seat MIX-friendly case

12. The mixing for the multi-seat election
13. The active case: An announcement
14. Variants
15. Conclusions

1. Special Secret Sharing Schemes

The most known secret sharing scheme is Shamir's secret sharing scheme (over 11,000 citations). His approach was to consider:

1. the secret and shares to be in a finite field,
2. to have the dealer use a computer to generate shares, and
3. to use computers to reconstruct the secret.

Since the invention of Visual Cryptography by Kafri and Keren in 1987, many researchers have stepped away from these restrictions (note that this was reinvented by Naor and Shamir in 1994 and that Kafri-Keren have 225 citations and Naor-Shamir have 2741).

Generalizing from finite field to Abelian Groups was initiated by

Desmedt-Frankel, published in 1994 (see also: Cramer-Fehr, Cramer-Fehr-Stam and the Cramer-Fehr-Ishai-Kushilevitz application to MPC).

After many years of research, in 2007 Desmedt-Pieprzyk-Steinfeld-Wang succeeded in making black-box “MPC” computations over non-Abelian groups. The motivation was purely theoretical. Today we will see an application of the situation in which:

the secret and shares belongs to a non-Abelian group,

i.e., S_n (or a subgroup of S_n, such as Z_n).

2. Our setting: Post Snowden elections

Post Snowden: today most people understand that computers, laptops can be hacked and may have trapdoors, malware, etc.

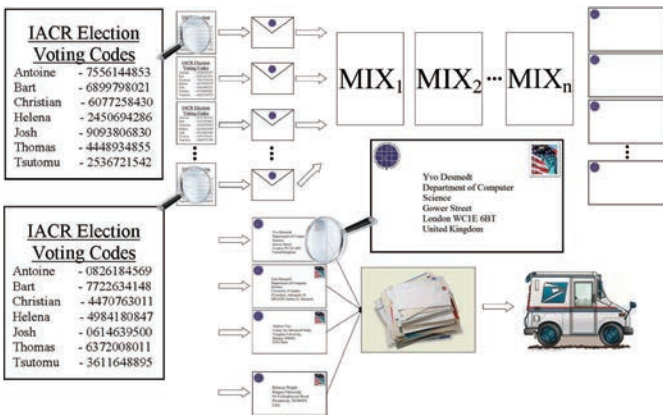
Potential solutions:

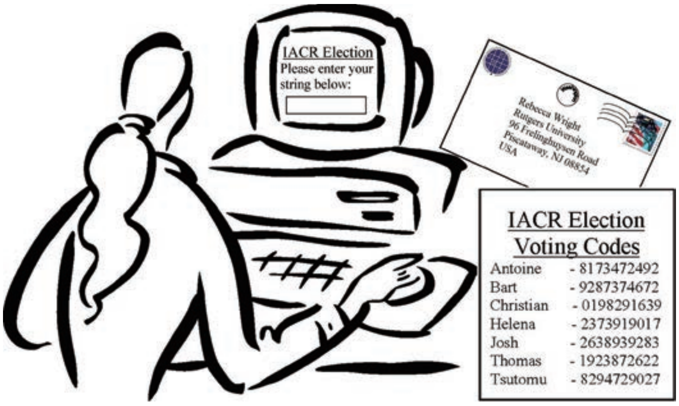
- Halderman (2015) recommended to stop using Internet Voting.
- We believe we need to restart/encourage a line of research in which we wonder how to vote assuming that the device you use for voting has been hacked.

Our model (high level): we assume we can not trust:

- any single party,
- any single device, etc.

3. A pioneering approach: Chaum’s Code Voting





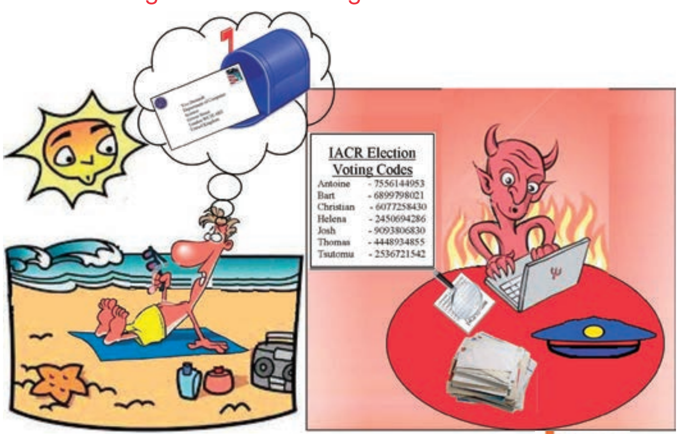
4. Advantages/disadvantages of code voting

Advantages of Code Voting: secure even if voter's machine hacked.

Disadvantages:

- requires IACR to send random numbers by postal mail, and
- no collusion between postal system (or sender of envelopes) and the party receiving the vote.
- authorities do not like the system because it differs too much from what is used today!

Ballot stuffing with Code Voting



5. Our setting, assumptions and their impacts

Our setting:

1. Voter votes using an untrusted device
2. The voter has access to many communication devices/media (e.g., home PC, mobile, at work, in the library, postal)
3. Voter uses "human computations," which we checked on reliability (see further).
4. Authorities use untrusted computers, potentially with state sponsored malware.

Our first model:

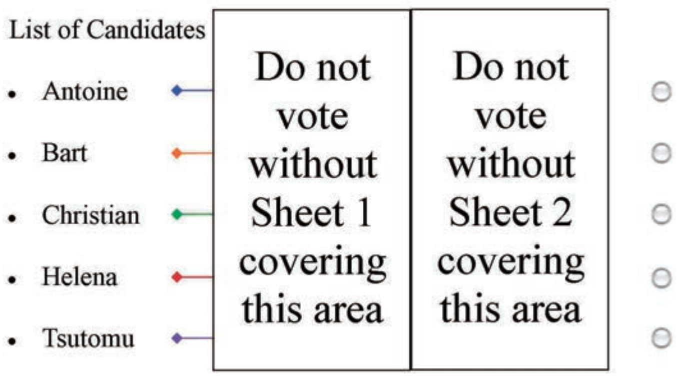
1. at most t devices/parties are infected.
2. our adversary is passive, curious, but not interested in: modifying the vote, in a DoS, etc. (see further)

Impact:

- Many cryptographic tools become useless, such as: AES, ElGamal, ZKIP, NIZK.
- So, we need to make a new MIX

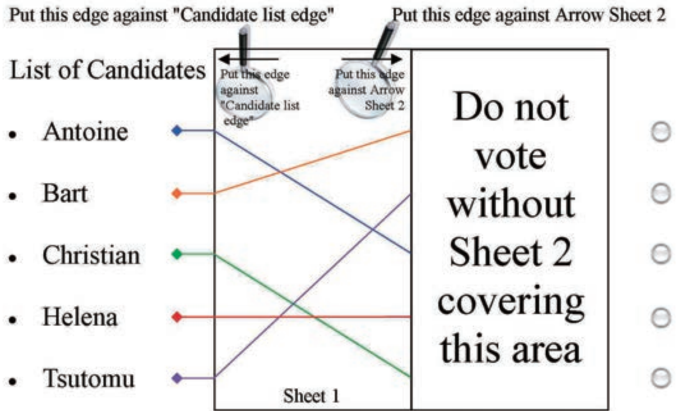
6. The voting: passive adversary only

A user friendly approach: (multi-seat, not "code-voting", $t = 1$)



6. THE VOTING: PASSIVE ADVERSARY ONLY

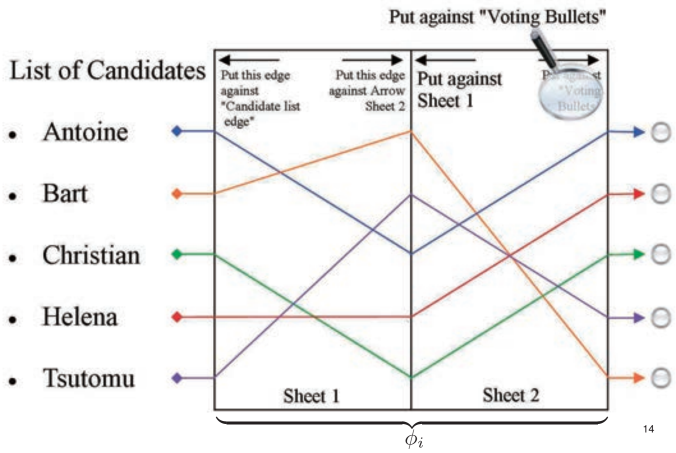
A user friendly approach: (multi-seat, not “code-voting”, $t = 1$)



©Yvo Desmedt and Stelios Erotokritou 14

6. THE VOTING: PASSIVE ADVERSARY ONLY

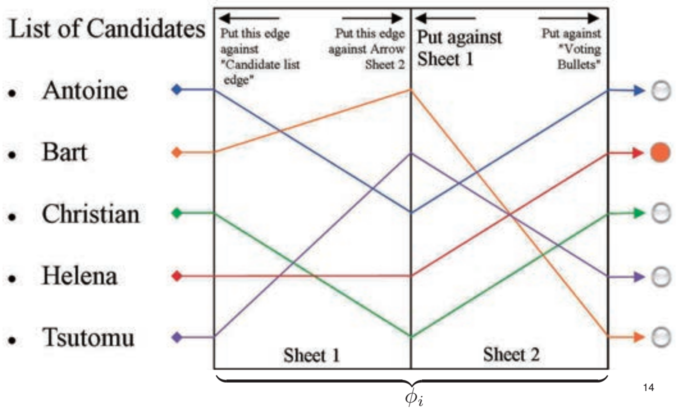
A user friendly approach: (multi-seat, not “code-voting”, $t = 1$)



14

6. THE VOTING: PASSIVE ADVERSARY ONLY

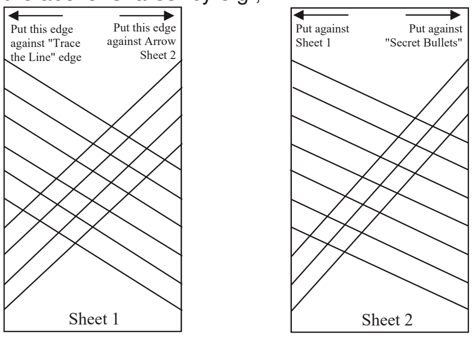
A user friendly approach: (multi-seat, not “code-voting”, $t = 1$)



14

In the single-seat election (mix friendly), we use code-voting ($t = 1$)

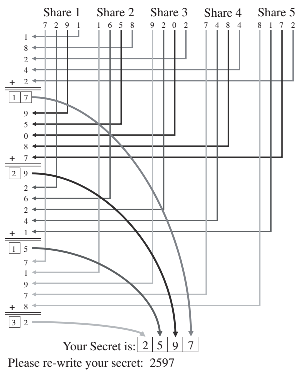
We regard the Abelian group $Z_{10}(+)$ as a subgroup of S_{10} and replace the above "shares" by e.g.,



These corresponding to an addition plus 4 $\mathrm{mod}10$ and plus 3 $\mathrm{mod}10$ respectively. We assume there are 10 candidates.

7. Some usability tests (SCN 2012)

How good are users able to add strings of numbers, each $\mathrm{mod}10$? Our test show only 95% get this correct, even when helping users, as following:



Details:

We asked 100 participants to do several tests (their ages did not surpass 65).

Asking to add 5 shares of 4 digits $\mathrm{mod}10$, 95% of the people computed the correct result, using the above visual tool to avoid confusion.

However, when using the permutation based addition, 99% of the people computed the correct result.

A common comment from the participants was that the permutation based $\mathrm{mod}10$ addition was extremely easy - whereas the other experiment was rather challenging for some people.

8. HIGH LEVEL DESCRIPTION

Background: secret shares

Example: 2-out-of-2:

Goal: Give binary secret s to 2 parties, Alice and Bob.

How: Flip a coin. Give the result, s_1, to Alice.

Give Bob: $s \oplus s_1$.

Can be generalized to:

- work over any finite group,
- the case we do not trust t insiders.

Just let $s = s_1 \circ s_2 \circ \cdots \circ s_{t+1}$.

High level protocol description:

1. We use a Code Generation Entity (CGE), which will in the pre-voting stage choose initial one-time pad (informally, π_i) for each voter.
2. Our MIX network uses layers, each layer having at least $t+1$ shares.
3. The CGE sends shares $(t+1)$ of these π_i to the MIX servers in the first layer.
4. The MIX network anonymizes and modifies the shares of π_i. The permutations used are the same for all the shares of the same value. For this, each layer had a leader that remembers the permutation used and the modifications done at that layer.

5. Each server in the last layer of the MIX sends a share to each voter (communication paths used by different servers are vertex disjoint).
6. The voter combines the shares (see above) and votes.
7. The voter sends the “encrypted” vote back to the leader of the last layer of the MIX network.
8. Starting with the leader of the last layer, all permutations and modifications done at that layer are undone.
9. The leader of the first layer of the MIX sends the almost-unencrypted vote to the CGI.
10. The CGI uses the inverse of its one-time pad.

9. DETAILS: TECHNICAL BACKGROUND

We primarily use (besides MIX and shares):

- Concepts from secure multiparty computation
  Simplified goal: given shares of s and shares of u how to make shares of $s * u$, without computing s and u.

- Desmedt-Kurosawa 2000 introduced:
  Definition 1. We say that $(X, \mathcal{B})$ is an (n, b, t)-verifiers set system if:
  1. $|X| = n$,
  2. $|B_i| = t + 1$ for $i = 1, 2, \ldots, b$, and
  3. for any subset $F \subset X$ with $|F| \leq t$, there exists a $B_i \in \mathcal{B}$ such that $F \cap B_i = \emptyset$.

Vertex disjoint paths: paths p_1 and p_2 from S to R are vertex disjoint if the nodes on path p_1, and on p_2, except for S and R are disjoint.

10. THE MIXING FOR THE SINGLE-SEAT MIX-FRIENDLY CASE

We have several protocols, of which we describe the simplest.

In the simplest, we require that each server in layer i is physically different from each server in layer j ($i \neq j$).

Note: Our MIX-friendly protocols can also be used in situations in which we have a single receiver (can be generalized) and multiple senders. The receiver should not learn who the sender is. For simplicity we focus on voting.

In below protocol we assume that $b = t + 1$. We denote the servers in layer i by a "block" B_i.

Protocol 1. *Prevoting protocol*

Step 1 Let π_i^1 be the i^{th} one-time pad (where $1 \leq i \leq v$). The receiver

(CGI) shares each π_i^1 into $t+1$ shares $\pi_{i,j}^1 \in F_{2^l}$ (where $1 \leq j \leq t+1$) and privately sends $\pi_{i,j}^1$ to the corresponding MIX $MIX_{1,j}$ in block B_1.

Step 2 The *leader* of B_1 (we call $MIX_{1,1}$) informs all others MIX servers in B_1 how they have to permute the i-index of all above $\pi_{i,j}^1$. This permutation is defined by $\rho_1 \in_R S_v$.

Step 3 On the i indices all MIX servers in B_1 apply the permutation ρ_1. So, $\pi_{i,j}^1 := \pi_{\rho_1(i),j}^1$.

Step 4 The *leader* of B_1 chooses $t+1$ random bit string modifiers $\omega_{i,j}^1 \in_R F_{2^l}$ and privately sends $\omega_{i,j}^1$ to parties in B_1.

Step 5 For each (i,j) the $t+1$ values $\pi_{i,j}^1$ are regarded as shares of π_i^1. Similarly, the $t+1$ values $\omega_{i,j}^1$ are regarded as shares of ω_i^1.

The MIX server in B_1 computes $\pi_{ij}^2 = \omega_{ij}^1 + \pi_{ij}^1$.
$\pi_{i,j}^2$ are regarded as shares of π^2, the $\rho_1(i)$ permuted and modified one time pad.

Step 6 Steps 2-5 are repeated, incrementing by one the indices of B_1 and B_2 until the last block B_b is reached.

Step 7 Shares held by MIX-servers of block B_{t+1} are denoted as $\phi_{i,j}$. $MIX_{t+1,j} \in B_{t+1}$ then sends $\phi_{i,j}$ to the i^{th} sender.
The communication paths used by different servers in block B_{t+1} are vertex disjoint.

Voting

1. The vote recombines the shares (see above) to make its one-time-pad and then this is used to encrypt the number of the candidate chosen.

2. The voter sends the encrypted vote to the leader of the last layer of the MIX network.

MIXING the votes

1. The leader of block $j = t+1$ having received v votes, "decrypts" the votes using $-\omega_i^k$.
2. The leader of block j permutations using ρ_j^{-1} to undo the earlier permutations on the order of the votes.
3. The leader of block j sends all so obtained v "votes" to the leader of block $j-1$.
4. Above steps are repeated.
5. The leader of block 1 sends the final "decrypted" votes to the CGI.

Theorem 1. *The above protocol is a reliable, private and anonymous message transmission protocol.*

For the proof, see the paper for details.

11. THE MIXING FOR THE SINGLE-SEAT: EFFICIENCY IMPROVEMENT

We can improve on the number of servers and the number of layers we need, by using concepts of verifiers set system, and modeling the communication system between the different servers in the layers as a graph (as in PSMT). We modify the communication between two layers to maintain the security.

Concept: (see Burmester-Desmedt 2004, formalized by Desmedt-Wang-Burmester 2005)

Color-based adversary structure: computers running the same platform are given the same color. We assume at most t color are corrupted, i.e., nodes corrupted have at most t different colors.

In our context, we want to reuse as many times as the same MIX

servers.

When a MIX server appears twice in the Directed Acyclic Graph between the CGI and the voters, we color it with the same color. We then consider PSMT in which we have a general adversary structure defined by the color based one.

Solution proposed: see Erotokritou-Desmedt 2012 (SCN) and also Vote ID 2015.

12. The mixing for the multi-seat election

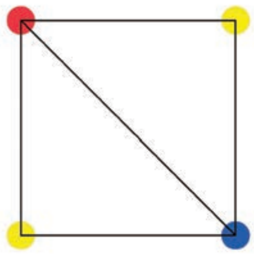
Sketch:

Above works well because we work over an Abelian group. In the case of multi-seat elections, the one-time-pad is a permutation, and so no longer an Abelian group.

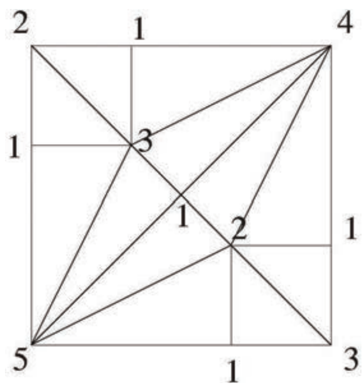
That means that Step 5 (in which we used +) in the last protocol does not work. We need to use a more complex protocol to modify the shares in the blocks. For this we use the work of Desmedt-Pieprzyk-Steinfeld-Wang of Crypto 2007.

Let us look at some nice graphs from this paper.

When $t = 1$:



and when $t = 2$:



13. AN ANNOUNCEMENT

We have a theoretical solution against active adversaries.

In this case, we consider:

- The mixing process: in which we can have active adversaries.
- The communication part: since different routes are used and since we do not use authentication, active adversaries could be in the communication protocol. Note that solving this using PSMT technology seems easy, however:
- The voter needs to deal with incorrect shares! The voter cannot even run Shamir's secret sharing!! So, certainly not a normal error-correction!

We use a variant of a repeat code to solve the last problem. (We

base this on the protocols for PSMT in SCN 2012 with an active adversary). While our test show that humans can combine permutations with roughly 99% being correct, we do not test whether humans can decode repeat codes correctly.

Therefore we call our solution (upcoming paper) theoretical.

14. VARIANTS

- Verification: Chaum allowed for voters to receive a confirmation that the vote was received, by giving the voters a second code for each candidate.
  We too can obtain this, i.e., our solution is a distributed secure version of Chaum confirmation which works among the lines of above.
- Better trust models: Our slides and text focuses on the case we do not trust t parties, devices, etc. We can generalize this to general access structure. That allows us to consider state sponsored hacking and state infected hardware/software.
  We can then assume at most t platforms have been hacked.

15. Conclusions

Achieving a good solution will not be easy. Indeed:

- Paranoid cryptographers assumed for 20 years that the servers used by authorities must be the bad guys!
- Cryptographers ignored for too long the fact politicians and the public want internet voting.
- Many cryptographers have no understanding of the weaknesses of modern PCs and what techniques hackers can deploy against voters.
- Theoreticians are not interested in secure Internet Voting.
- These promoting practical research do not understand it may take

10 years research with lots of interaction before a good solution might be presented. They want a solution now!

We showed that the disadvantages of Chaum's code voting can be addressed. We are aware that our solution is "Towards Secure Internet Voting."

It took 15 years to design reasonable voting schemes when using secure booths. So, we can expect that others will improve on our solutions.

Secret Sharing Schemes Under Guessing Secrecy

Mitsugu Iwamoto (Joint work with Junji Shikata)

The University of Electro-Communications

mitsugu@uec.ac.jp

Information theoretic security is a class of security notion to guarantee the security against adversaries with unbounded computing power. In particular, after seminal work by Shannon [5], *perfect secrecy* has been well investigated because of its importance. Recently, Alimomeni and Safavi-Naini introduced an information theoretic security notion called *guessing secrecy* for symmetric key encryption (SKE) [1].

In defining guessing secrecy, we assume that an adversary guesses a plaintext *only once* by using the corresponding ciphertext without a key. If the adversary tries to maximize the success probability of the guess and it is equivalent to the success probability in guessing the plaintext without the key, we can say that no advantage is given to the adversary from the ciphertext.

In the original guessing secrecy [1], the maximum success probability of guessing is averaged with respect to the ciphertexts, and hence, we call it *average* guessing secrecy. On the other hand, Iwamoto and Shikata later discussed the maximum probability of guessing in the worst case with respect to the ciphertext in defining guessing secrecy, which is called *worst-case* guessing secrecy. Intuitively, worst-case guessing secrecy offers intermediate level of security between average guessing secrecy and perfect secrecy. Iwamoto and Shikata also discussed average and worst case guessing secrecy for secret sharing schemes (SSS) as well as SKE [3, 4].

The aim of this talk is to shed light on the relations among perfect secrecy, average and worst case guessing secrecy by investigating several constructions of SKE and SSS. As a result, it turns out that the relations of the above-mentioned information theoretic security notions depend on the primitives, and the difference between SKE and $(2, 2)$–threshold SSSs becomes clearer.

The content of this talk is based on our previous work [2–4] and recent results.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Numbers JP15H02710, and JP17H01752.

References

[1] M. Alimomeni and R. Safavi-Naini. Guessing Secrecy. In *International Conference on Information Theoretic Security (ICITS)*, volume LNCS 7412, pages 1–13. Springer-Verlag, 2012.

[2] M. Iwamoto and J. Shikata. Information theoretic security for encryption based on conditional Rényi entropies. In C. Padró, editor, *International Conference on Information Theoretic Security (ICITS)*, volume LNCS8317, pages 103–121, 2013.

[3] M. Iwamoto and J. Shikata. Secret Sharing Schemes Based on Min-Entropies. *IEEE International Symposium on Information Theory - Proceedings*, pages 401–405, 2014.

[4] M. Iwamoto and J. Shikata. Constructions of Symmetric-key Encryption with Guessing Secrecy. In *International Symposi* 729, 2015.

[5] C. E. Shannon. Commun ., 28:656–715, Oct. 1949.

Secret Sharing Schemes under Guessing Secrecy

Mitsugu Iwamoto

The University of Electro-Communications, Japan.

12th June, 2017

IMI Workshop:
Cryptographic Technologies for Securing Network Storage
and Their Mathematical Modeling

Based on joint work with Junji Shikata, YNU
Appeared at ICITS2013, ISIT2014, 2015 & recent result.

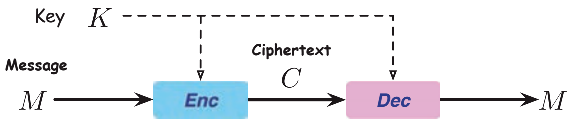
Outline

Introduction

Perfect Secrecy and Guessing Secrecy

Symmetric Key Encryption (SKE)

SKE: $\Sigma := (P_K, \mathrm{Enc}, \mathrm{Dec})$

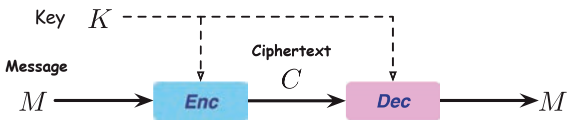


- Real values: key $k \in \mathcal{K}$, message $m \in \mathcal{M}$, ciphertext $c \in \mathcal{C}$
- Random variables: key K, message M, ciphertext C
  - ☞ $P_{KMC}(\cdot,\cdot,\cdot)$: joint probability distribution of K, M, C
  - ☞ $K \perp M$: K and M are independent
- No decryption error is assumed

Perfect Secrecy

[Shannon, 1950 (1945)]

Encryption: $\Sigma := (P_K, \mathrm{Enc}, \mathrm{Dec})$



Definition (Perfect Secrecy: PS)

Σ satisfies perfect secrecy (PS) if

$$\forall m \in \mathcal{M}, \forall c \in \mathcal{C}, \quad P_{M|C}(m|c) = P_M(m)$$

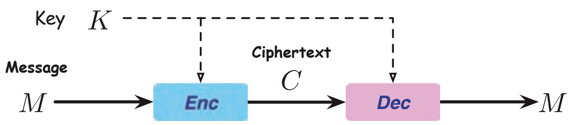
- i.e., M and C are statistically independent

- Σ is secure against adversaries with unbounded computing power

Guessing Secrecy for SKE

[Alimomeni, Safavi-Naini, ICITS2012]

SKE: $\Sigma := (P_K, \mathrm{Enc}, \mathrm{Dec})$



- Suppose that an adversary guesses m from c only once
- Best strategy: maximize success probabilities in guessing m
  - ☞ $\arg\max_m P_{M|C}(m|c)$: Most probable m when c is given
  - ☞ $\arg\max_m P_M(m)$: Most probable m when no information is given
- Two ways in treating the ciphertext c

Average / Worst-case Guessing Secrecy

Definition (Guessing Secrecy for SKE)

- Average GS, A-GS: [Alimomeni, Safavi-Naini, ICITS2012]

$$\mathbb{E}_C\left[\max_m P_{M|C}(m|C)\right] = \max_m P_M(m)$$

- Worst-case GS, W-GS: [I–Shikata, ICITS2013]

$$\max_c \max_m P_{M|C}(m|c) = \max_m P_M(m)$$

- Clearly,

[weaker] A-GS $\preceq$ W-GS $\preceq$ PS [stronger]

Our Interest

- Gaps among the security notions

Average / Worst-case Guessing Secrecy in Min-Entropies

Definition (Guessing Secrecy for SKE in Min-entropies)

- Average GS, A-GS: [Alimomeni, Safavi-Naini, ICITS2012]

$$R_\infty^{\mathsf{avg}}(M|C) = R_\infty(M)$$

- Worst-case GS, W-GS: [I–Shikata, ICITS2013]

$$R_\infty^{\mathsf{wst}}(M|C) = R_\infty(M)$$

where

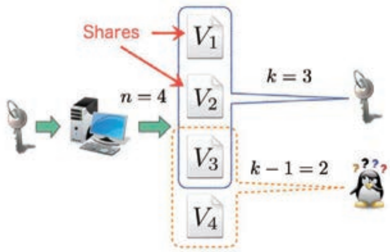
- $R_\infty(X) := -\log \max_x P_X(x)$
- $R_\infty^{\mathsf{avg}}(X|Y) := -\mathbb{E}_Y\left[\log \max_x P_X(x|Y)\right]$
- $R_\infty^{\mathsf{wst}}(X|Y) := -\log \max_{x,y} P_{X|Y}(x|y)$

Guessing Secrecy

for Secret Sharing Schemes

(k, n)-threshold Secret Sharing Schemes [Shamir, Blakley, 1979]

Example ($(3,4)$–SSS)



Definition (SSS under PS)

- S is decrypted from $\boldsymbol{A}$ without error if $|\boldsymbol{A}| \geq k$
- $\forall s \in \mathcal{S}, \forall v_{\boldsymbol{A}} \in \mathcal{V}^{|\boldsymbol{A}|}, P_{S|V_{\boldsymbol{A}}}(s|v_{\boldsymbol{A}}) = P_S(s)$ if $|\boldsymbol{A}| \leq k-1$

Guessing Secrecy for Secret Sharing

Definition (PS for Secret Sharing)

- $\forall s \in \mathcal{S}, \forall v_{\boldsymbol{A}} \in \mathcal{V}^{|\boldsymbol{A}|}, P_{S|V_{\boldsymbol{A}}}(s|v_{\boldsymbol{A}}) = P_S(s)$ if $|\boldsymbol{A}| \leq k-1$

Definition (GS for Secret Sharing)

- A-GS: $\mathbb{E}_{V_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|V_{\boldsymbol{A}})\right] = \max_{s \in \mathcal{S}} P_S(s)$ if $|\boldsymbol{A}| \leq k-1$
- W-GS: $\max_{v_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|v_{\boldsymbol{A}})\right] = \max_{s \in \mathcal{S}} P_S(s)$ if $|\boldsymbol{A}| \leq k-1$

- Clearly, [weaker] A-GS $\preceq$ W-GS $\preceq$ PS [stronger]

Our Interest

- Gaps among the security notions

Guessing Secrecy for Secret Sharing in Min-Entropies

Definition (GS for Secret Sharing Schemes in Probabilities)

- A-GS: $\mathbb{E}_{V_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|V_{\boldsymbol{A}})\right] = \max_{s \in \mathcal{S}} P_S(s)$ if $|\boldsymbol{A}| \leq k-1$
- W-GS: $\max_{v_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|v_{\boldsymbol{A}})\right] = \max_{s \in \mathcal{S}} P_S(s)$ if $|\boldsymbol{A}| \leq k-1$

Definition (GS for Secret Sharing Schemes in Min-Entropies)

- A-GS: $R_\infty^{\mathsf{avg}}(S|V_{\boldsymbol{A}}) = R_\infty(S)$ if $|\boldsymbol{A}| \leq k-1$
- W-GS: $R_\infty^{\mathsf{wst}}(S|V_{\boldsymbol{A}}) = R_\infty(S)$ if $|\boldsymbol{A}| \leq k-1$

Note

- This talk: we mainly focus on constructions of $(2,2)$–SSS under GS

Overview of This Talk

Obvious Relation

[weaker] A-GS $\preceq$ W-GS $\preceq$ PS [stronger]

Part I: A-GS vs. PS

- SKE & SSS: A-GS $\prec$ PS ("$\prec$" means that explicit gap exists)
- A-GS attains shorter share size than PS for ideal SSS

Part II: Security level of W-GS

- SKE: (A-GS $\prec$) W-GS $=$ PS
- SSS: A-GS $\prec$ W-GS $\prec$ PS

Key Point

- GS does not require statistical independence
- Non-uniformity of the secret: M and S

Part I

Average Guessing Secrecy

in Secret Sharing Schemes

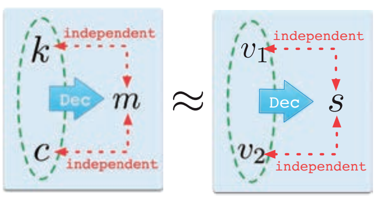
Naïve Idea: $(2,2)$-SSS as SKE

- We show how to construct SSS under A-GS
  - ☞ We concentrate on construction of $(2,2)$–SSS under A-GS
  - ☞ Easy to extend to (k,n)–threshold and general access structures

Naïve Idea:

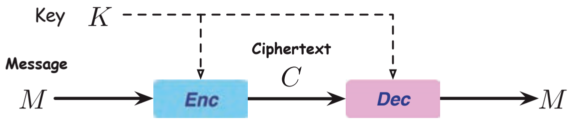
- SKE $\approx$ $(2,2)$–SSS under PS

SKE　　　　(2,2)-SSS



OTP-like SKE under A-GS

SKE: $\Sigma := (P_K, \mathsf{Enc}, \mathsf{Dec})$



OTP-like SKE [I–Shikata, ICITS2013]

- $\mathcal{M} = \mathcal{C} = \mathcal{K} = \{0, 1\}$
- $P_M(0) = P_K(0) = p$, $1/2 \le p \le 1 \Leftarrow$ PS iff $p = 1/2$
- One-time pad for 1-bit encryption:

  Encryption: $\pi_{enc}(k, m) = k \oplus m$

  Decryption: $\pi_{dec}(k, c) = k \oplus c$

Analysis on OTP-like Construction

☞ $q := 1 - p < 1/2$

M	K	C	P_{MKC}	$P_{M\|C}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

For $c \in \{0, 1\}$, $\max_m P_{M|C}(m|c)$ is attained by $m = 0$, hence,

$$\mathbb{E}_C\left[\max_m P_{M|C}(m|C)\right] = P_M(0)\ (= \max_m P_M(m))$$

Theorem [I–Shikata, ICITS2013]

- Security: $R_\infty(M) = R_\infty(M|C) = -\log p$, but $M \not\perp C$!
- Efficiency (in key-size): $R_\infty(K) = R_\infty(M) = -\log p$ (optimal)

Regarding SKE as $(2, 2)$–SSS

One Time Pad (OTP)

M	K	C	P_{MKC}	$P_{M\|C}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

$\Rightarrow$

OTP-like SSS

S	V_1	V_2	$P_{SV_1V_2}$	$P_{S\|V_2}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

🙂 Can be extended to (n, n)–threshold and general access structures

Question

- How about the share size ?
- Can it be ideal secret sharing?

Efficiency in Share Size: Ideal GS under PS

Proposition (Lower Bound) [Karnin–Greene–Hellman, 1983]

$$\forall P_S \in \mathscr{P}(\mathcal{S}), \quad \text{PS-SSS} \Rightarrow H(V_i) \geq H(S), i \in [n]$$

Definition (Ideal SSS with perfect secrecy)

$$\text{Ideal (i.e., efficient) PS-SSS} \overset{\text{def}}{\Longleftrightarrow} H(V_i) = H(S), i \in [n]$$

Proposition [Blundo et al., 1998]

$$\forall P_S \in \mathscr{P}(\mathcal{S}), \quad \text{PS-SSS} \Rightarrow H(V_i) \geq \log |\mathcal{S}|, i \in [n]$$

where the equalities hold only when S is uniform

Corollary

PS-SSS can be ideal iff S is uniform

Ideal SSS under A-GS

Theorem [Dodis ICITS2012, I–Shikata ICITS2013]

$$\text{A-GS/W-GS} \Rightarrow R_\infty(V_i) \geq R_\infty(S)$$

Pf) Lower bounding via Rényi entropies of order α and $\alpha \to \infty$ (omitted)

Question

Does ideal (k, n)-threshold GS-SSS exist for non-uniform S?

$$R_\infty(V_i) = R_\infty(S), \quad i \in [n]$$

c.f.) (k, n)-threshold PS-SSS can be ideal iff S is uniform

Theorem [I–Shikata, ISIT2014]

$\exists S$ (non-uniform), $\exists$ ideal (k, n)–SSS under A-GS

OTP-like SSS Cannot Be "Non-trivial" SSS under A-GS

One Time Pad (OTP)

M	K	C	P_{MKC}	$P_{M\|C}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

$\Rightarrow$

OTP-like SSS

S	V_1	V_2	$P_{SV_1V_2}$	$P_{S\|V_2}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

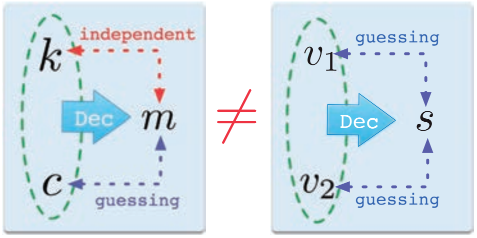
- If OTP-like GS-SSS is ideal: $R_\infty(S) = R_\infty(V_1) = R_\infty(V_2)$
  - ☞ $R_\infty(S) = R_\infty(V_1) = -\log p$ but $R_\infty(V_2) = -\log(p^2 + q^2)$,
  - ☞ OTP-like Ideal GS-SSS $\Rightarrow p = 0, 1/2$

In this case GS-SSS = PS-SSS $\Rightarrow$ trivial and not interesting

Towards "Non-trivial" Ideal $(2,2)$–SSS under A-GS

- OTP-like $(2,2)$-SSS cannot be "non-trivial" SSS under A-GS!
- More efficient ideal SSS is possible under A-GS !

SKE (2,2)-SSS



"Non-trivial" Ideal $(2,2)$–SSS under A-GS

Example [I–Shikata, ISIT2014]

OTP-like SSS under A-GS

$(q := 1-p < 1/2)$

S	V_1	V_2	$P_{SV_1V_2}$	$P_{S\|V_2}$
0	0	0	p^2	$\frac{p^2}{p^2+q^2}$
1	1	0	q^2	$\frac{q^2}{p^2+q^2}$
0	1	1	pq	$1/2$
1	0	1	pq	$1/2$

Ideal $\Rightarrow p = 0, 1/2$

Ideal SSS under A-GS

$(p \geq 1/4)$

S	V_1	V_2	$P_{SV_1V_2}$	$P_{S\|V_2}$
0	0	0	p	$\frac{3p}{1+2p}$
1	1	0	$\frac{1-p}{3}$	$\frac{1-p}{1+2p}$
0	1	1	$\frac{1-p}{3}$	$1/2$
1	0	1	$\frac{1-p}{3}$	$1/2$

$P_S(0) = P_{V_i}(0) = p + \frac{1-p}{3}$

- For each $v_i \in \{0,1\}$, $\max_s P_{S|V_i}(s|v_i)$ is attained by $s = 0$, hence,

$$\mathbb{E}_{V_i}\left[\max_s P_{S|V_i}(s|V_i)\right] = P_S(0) \Leftrightarrow R_\infty(S|V_i) = R_\infty(S)$$

Efficiency of Ideal SSS Under A-GS

Analysis [I–Shikata, ISIT2014]

The proposed construction satisfies

$$R_\infty(V_1) = R_\infty(V_2) = R_\infty(S) = -\log \tfrac{1+2p}{3}$$

Since S is binary,

$$H(V_1) = H(V_2) = H(S) = h(\tfrac{1+2p}{3}) < 1 \text{ if } p > 1/4$$

☞ PS-SSS cannot attain $H(V_i) < 1$ due to the following result:

Proposition [Blundo et al., IPL1998]

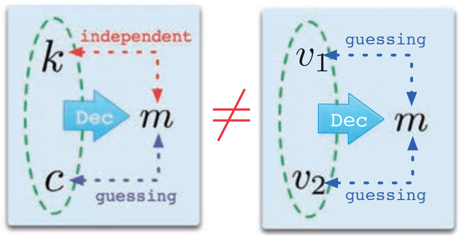
$$\forall P_S \in \mathscr{P}(\{0,1\}), \quad \text{PS-SSS} \Rightarrow H(V_i) \geq 1 \ (= \log|\mathcal{S}|)$$

where the equalities hold only when S is uniform

Summary of Part I

- SKE & SSS: A-GS $\prec$ PS
- A-GS attains shorter share size than PS for ideal SSS
  - ☞ Non-trivial Ideal SSS cannot be obtained from SKE under A-GS
- Observation

GS-Encryption $\neq$ GS-Secret Sharing



Part II

Worst-case Guessing Secrecy

in Secret Sharing Schemes

Guessing Secrecy in Secret Sharing Schemes

Definition (GS for Secret Sharing)

- A-GS: $\max_{s \in \mathcal{S}} P_S(s) = \mathbb{E}_{V_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|V_{\boldsymbol{A}})\right]$ if $|\boldsymbol{A}| \leq k-1$
- W-GS: $\max_{s \in \mathcal{S}} P_S(s) = \max_{v_{\boldsymbol{A}}}\left[\max_{s \in \mathcal{S}} P_{S|V_{\boldsymbol{A}}}(s|v_{\boldsymbol{A}})\right]$ if $|\boldsymbol{A}| \leq k-1$

- Clearly, [weaker] A-GS $\preceq$ W-GS $\preceq$ PS [stronger]

Claim of Part II

- SKE: (A-GS $\prec$) W-GS $=$ PS [I–Shikata, ISIT2015]
- SSS: A-GS $\prec$ W-GS $\prec$ PS

"Weak" Independence between S and V_i under W-GS

Theorem (Necessary Condition for W-GS-SSS)

- $s^* := \arg\max_m P_S(s)$, $i \in \{1,2\}$

$$\forall v_i, \qquad P_{SV_i}(s^*, v_i) - P_S(s^*)P_{V_i}(v_i) = 0 \qquad \text{(w-ind)}$$

Pf) Easy to derive from the definition (omitted)

Remark

- If $S \perp V_i$ (i.e., PS),

$$\forall s, \forall v_i, \quad P_{SV_i}(s, v_i) - P_S(s)P_{V_i}(v_i) = 0$$

then (w-ind) is obviously satisfied

Encryption by Latin Square

- We require $|\mathcal{S}| = |\mathcal{V}|$

  $\because$) A-GS, W-GS $\Rightarrow |\mathcal{S}| \le |\mathcal{V}|$ (proof: omitted)

Definition (SSS based on Latin square)

For a fixed $s \in \mathcal{S}$, the map $f_s : v_1 \mapsto v_2$ is bijective

Example (Value of s when v_1 and v_2 are given)

$v_1 \backslash v_2$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

- Regarding (s, v_1, v_2) as (m, k, c), $(2,2)$–SSS becomes SKE
- In the following, assume SKE & SSS are based on Latin square

Distributions of Shares Are Equivalent via Permutation

Weak Independence

$$i \in \{1,2\}, \quad \forall v_i, \quad P_{SV_i}(s^*, v_i) - P_S(s^*)P_{V_i}(v_i) = 0 \qquad \text{(w-ind)}$$

Theorem (Equivalence via permutation)

Probability vector $[P_{V_1}(v_1)]_{v_1 \in \mathcal{V}}$ is obtained by permuting $[P_{V_2}(v_2)]_{v_2 \in \mathcal{V}}$

Pf) Immediately follows from def. of Latin square (L) and (w-ind):

$$\begin{aligned} 0 &\overset{\text{(w-ind)}}{=} P_{SV_1}(s^*, v_1) - P_S(s^*)P_{V_1}(v_1) \\ &\overset{(L)}{=} P_{SV_2}(s^*, f_{s^*}(v_1)) - P_S(s^*)P_{V_i}(v_i) \\ &\overset{\text{(w-ind)}}{=} P_S(s^*)P_{V_2}(f_{s^*}(v_1)) - P_S(s^*)P_{V_i}(v_i) \end{aligned}$$

- This result does not hold in A-GS if S is not uniform

SKE: W-GS = PS

- Regarding (s, v_1, v_2) as (m, k, c), $(2,2)$–SSS becomes SKE

Theorem
If W-GS SSS is based on Latin square

$$V_1 \perp S \Longrightarrow V_1 \text{ is uniform over } \mathcal{V}$$

Corollary [I–Shikata, ISIT2015]
If W-GS SKE is based on Latin square

$$K \perp M \Longrightarrow K \text{ is uniform over } \mathcal{K}$$
$$\Longrightarrow \text{SKE satisfies PS}$$

Proof of W-GS = PS on SKE

Theorem
If W-GS SSS is based on Latin square

$$V_1 \perp S \Longrightarrow V_1 \text{ is uniform over } \mathcal{V}$$

Pf) $v_i^* := \arg\max_{v_i} P_{V_i}(v_i) \Longrightarrow P_{V_1}(v_1^*) = P_{V_2}(v_2^*)$ $\quad (\sharp)$

$$\begin{aligned}
0 &= \sum_{s\in\mathcal{S}} \left(P_{SV_2}(s, v_2^*) - P_S(s)P_{V_2}(v_2^*)\right) \\
&= \sum_{s\in\mathcal{S}} \left(P_{SV_1}(s, f_s^{-1}(v_2^*)) - P_S(s)P_{V_1}(v_1^*)\right) && \because)\ (L)\ \&\ (\sharp) \\
&= \sum_{s\in\mathcal{S}} P_S(s)\left(P_{V_1}(f_s^{-1}(v_2^*)) - P_{V_1}(v_1^*)\right) && \because)\ S \perp V_1 \\
&\Longrightarrow \forall v_1, P_{V_1}(v_1) = P_{V_1}(v_1^*) \ \text{ i.e., } V_1 \text{ is uniform}
\end{aligned}$$

SSS: W-GS $\prec$ PS ?

Theorem (Necessary Condition for W-GS-SSS)
- $s^* := \arg\max_m P_M(m)$, $i \in \{1,2\}$

$$\forall v_i, \qquad P_{SV_i}(s^*, v_i) - P_S(s^*)P_{V_i}(v_i) = 0 \qquad \text{(w-ind)}$$

Question
- Can S and V_i be correlated while satisfying (w-ind)? $\Longrightarrow$ Yes!

Example of $(2,2)$–SSS: W-GS $\prec$ PS

- $\max_s P_S(s) = \max_{s,v_1} P_{S|V_1}(s|v_1) = \max_{s,v_2} P_{S|V_2}(s|v_2) = 1/2$

s	v_1	v_2	$P_S(s)$	$P_{SV_1V_2}(s,v_1,v_2)$	$P_S(s)P_{V_1}(v_1)$	$P_S(s)P_{V_2}(v_2)$
	0	0		$7/40$	$7/40$	$7/40$
0	1	2	$1/2$	$7/40$	$7/40$	$7/40$
	2	1		$6/40$	$6/40$	$6/40$
	0	2		$5/40$	$91/800$	$91/800$
1	1	1	$13/40$	$4/40$	$91/800$	$78/800$
	2	0		$4/40$	$78/800$	$91/800$
	0	1		$2/40$	$49/800$	$42/800$
2	1	0	$7/40$	$3/40$	$49/800$	$49/800$
	2	2		$2/40$	$42/800$	$49/800$

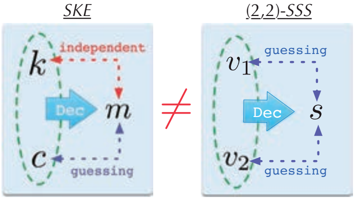
A-GS and W-GS Can Depend on Shares

- $\max_s P_S(s) = \max_{s,v_1} P_{S|V_1}(s|v_1) = \mathbb{E}_{V_2}\left[\max_s P_{S|V_2}(s|V_2)\right] = 4/7$

s	v_1	v_2	$P_S(s)$	$P_{SV_1V_2}(s,v_1,v_2)$	$P_S(s)P_{V_1}(v_1)$	$P_S(s)P_{V_2}(v_2)$
	0	0		$16/49$	$16/49$	$80/343$
0	1	2	$4/7$	$8/49$	$8/49$	$44/343$
	2	1		$4/49$	$4/49$	$72/343$
	0	2		$8/49$	$48/343$	$240/2401$
1	1	1	$12/49$	$3/49$	$24/343$	$132/2401$
	2	0		$1/49$	$12/343$	$216/2401$
	0	1		$4/49$	$36/343$	$180/2401$
2	1	0	$9/49$	$3/49$	$18/343$	$99/2401$
	2	2		$2/49$	$9/343$	$162/2401$

Summary of Part II

- Relation among security notions depends on primitive:
  - SKE: (A-GS $\prec$) W-GS $=$ PS
  - SSS: A-GS $\prec$ W-GS $\prec$ PS
  - "Weak" independence is important
  - Future work: General construction of SSS under W-GS
- Observation: SKE (2,2)-SSS



Function Secret Sharing Using Fourier Basis

Naruhiro KUROKAWA
(Joint work with Takuya OHSAWA and Takeshi KOSHIBA)

Bank of Japan
naruhiro.kurokawa@boj.or.jp

Function secret sharing (FSS) scheme, formally introduced by Boyle et al.[1] at EUROCRYPT2015, is a mechanism that calculates a function $f(x)$ for $x \in \{0,1\}^n$ which is shared among p parties, by using distributed function $f_i : \{0,1\}^n \to \mathbb{G}(1 \le i \le p)$, where $\mathbb{G}$ is an Abelian group, while the function $f : \{0,1\}^n \to \mathbb{G}$ is kept secret to the parties. We observe that any function f can be described as a linear combination of the basis functions by regarding the function space as a vector space of dimension 2^n and give a new framework for FSS schemes based on this observation. Based on the new framework, we introduce a new FSS scheme using the Fourier basis. This method provides efficient computation for a different class of functions (e.g., hard-core predicates of one-way functions), which may be inefficient to compute if we use the standard basis such as point functions. Our FSS scheme based on Fourier basis is quite simple due to the fact that the Fourier basis is closed under the multiplication, while the previous constructions[1, 3] have to incorporate some complex mechanisms to overcome the difficulty.

References

[1] E. Boyle, N. Gilboa and Y. Ishai: Function secret sharing, in: EUROCRYPT 2015, Part II, LNCS 9057, pp.337–367, 2015.

[2] N. Gilboa and Y. Ishai: Distributed point functions and their applications, in: EUROCRYPT 2014, LNCS 8441, pp.640–658, 2014.

[3] T. Ohsawa, N. Kurokawa and T. Koshiba: Function Secret Sharing Using Fourier Basis, in: Proc. the 8th International workshop on Trustworthy Computing and Security, Lecture Notes on Data Engineering and Communications Technologies, to appear, Springer.

Function Secret Sharing Using Fourier Basis

Naruhiro KUROKAWA (Bank of Japan)

Joint work with Takuya OHSAWA[1.] and Takeshi KOSHIBA [2.]

(1. Saitama Univ. 2.Waseda Univ.)

Topics

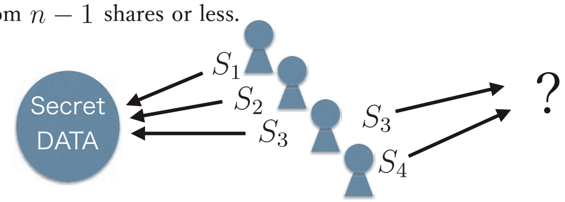
- Threshold Secret Sharing
- Definition Function Secret Sharing(FSS)
- Related work (Distributed Point Function)
- Linear Combination of FSS
- Basis function
- General FSS by using Basis FSS
- Distributed Fourier Basis
- Conclusion

2

Threshold Secret Sharing

In Secret Sharing (SS) scheme, share information $S_i (1 \leq i \leq p)$, generated from the secret information S, are distributed to p parties.

In (n, p)-threshold SS scheme, the secret information S can be recovered from n shares, but no information on S is leaked from $n-1$ shares or less.



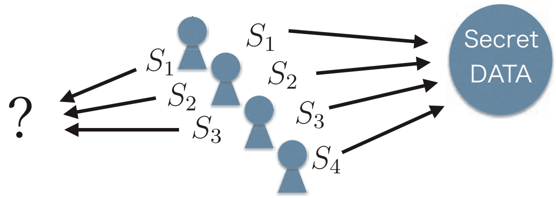
$(4,3)$-Threshold SS

3

(n,n)-Threshold Secret Sharing

simple (n,n)-threshold scheme

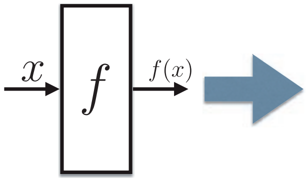
$$S = \sum_{i=1}^{p} S_i \pmod q$$

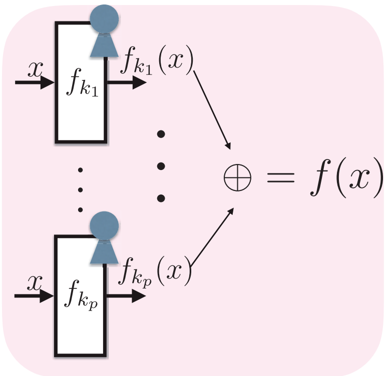


$(4,4)$-Threshold SS

4

p-party Function Secret Sharing





Correctness：

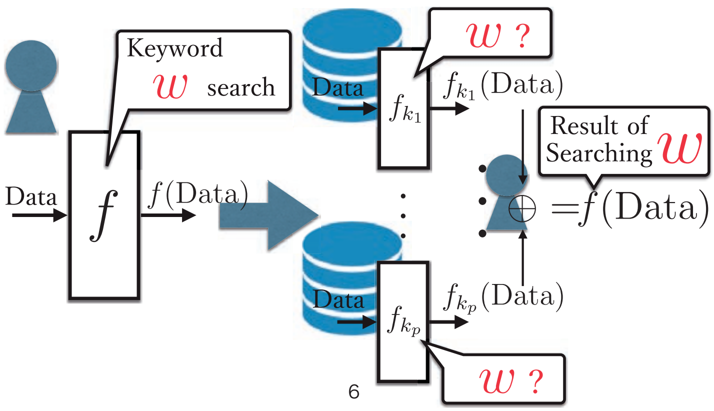
$$f(x) = \sum_{i=1}^{p} f_{k_i}(x)$$

Security：

f is not leaked out from at most $p-1$ distributed functions

5

Application of FSS (Distributed Database)



6

Definition of FSS

A p-party FSS scheme with respect to a function class $\mathcal{F}$ is a pair of PPT algorithms $(Gen, Eval)$.
The functional value $f(x)$ is obtained from all shares $(y_1, y_2, \cdots, y_p)$ of the parties by using a decode function Dec.

$Gen(1^\lambda, f) \rightarrow (k_1, \cdots, k_p)$

$f \in \mathcal{F}$: Secret Target function λ : Security parameter

$Eval(i, k_i, x) \rightarrow y_i$

y_i : i-th party's evaluated share

$Dec(y_1, \cdots, y_p) \rightarrow f(x)$

7

Related work

Initial 2-party DPF
[Gilboa et al, 2014]

Improved 2-party DPF
p-party DPF
[Boyle et al, 2015]

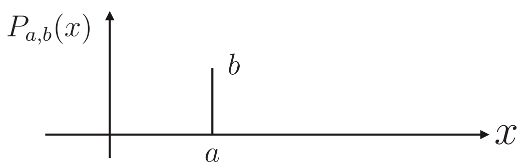
FSS for Fourier Basis
[Ohsawa, K & Koshiba, 2017]

8

Point function

For $a \in \{0,1\}^n, \ b \in \{0,1\}^m$,
the point function $P_{a,b} : \{0,1\}^n \rightarrow \{0,1\}^m$

$$\begin{cases} P_{a,b}(a) = b, \\ P_{a,b}(a') = 0^m \ \ for\ all\ a' \neq a \end{cases}$$



9

Distributed Point function(DPF)

[Gilboa et al, 2014]

$Gen(a,b) \to k_0, k_1 \in (\mathbb{F}_{2^m})^{2^n}$

$$\begin{array}{rl} k_0 = & r_1,\ r_2,\ \cdots,\ r_a^0,\ \cdots,\ r_{2^n} \\ k_1 = & r_1,\ r_2,\ \cdots,\ r_a^1,\ \cdots,\ r_{2^n} \\ \hline k_0 \oplus k_1 = & 0\quad 0\qquad\quad b\qquad\quad 0 \end{array}$$

$Eval(i, k_i, x) \to k_i[x']$ which is the x'-th element of k_i

Dec

$Eval(0, k_0, x) \oplus Eval(1, k_1, x) = P_{a,b}(x)$

Key size

$O(2^n)$

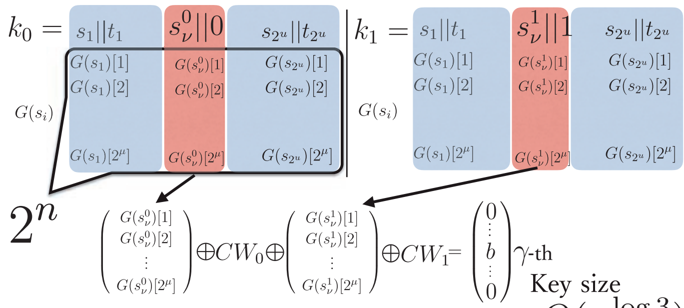
10

Distributed Point function(DPF)

[Gilboa et al, 2014]

x is viewed as a pair$(i,j) \in \{0,1\}^u \times \{0,1\}^\mu$ $\quad a = (\nu, \gamma)$

$u = \lceil \log((\frac{m \cdot 2^n}{\kappa+1})^{1/2}) \rceil$ $\quad \mu = \lceil \log((\frac{2^n \cdot (\kappa+1)}{m})^{1/2}) \rceil$ $\quad G: \{0,1\}^\kappa \to \{0,1\}^{\kappa \cdot 2^{n/2}}$



Key size

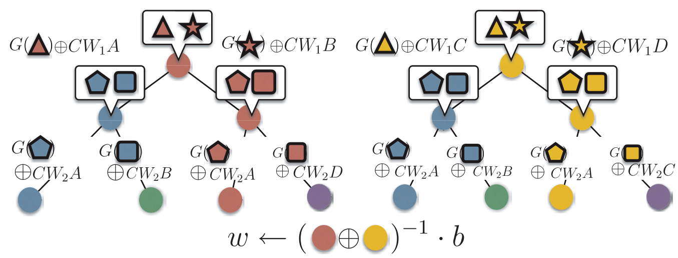
$O(n^{\log 3})$

11

Improved DPF

[Boyle et al, 2015]

$P_{a,b}(x) \quad x = (x_1, x_2, x_3, \cdots, x_n) \in \{0,1\}^n$



$Eval(k_i, x) \to \cdot w$

$Dec: \cdot w \oplus \cdot w \to b$

$\cancel{\cdot w} \oplus \cancel{\cdot w}$

Key size

$O(n)$

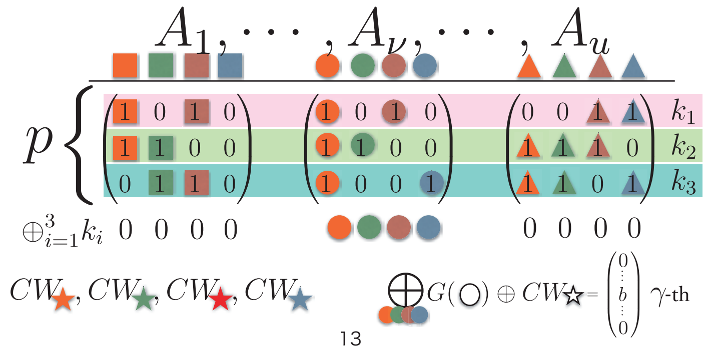
12

p-party DPF

[Boyle et al, 2015]

In case of $p = 3$ $\quad a = (\nu, \gamma)$

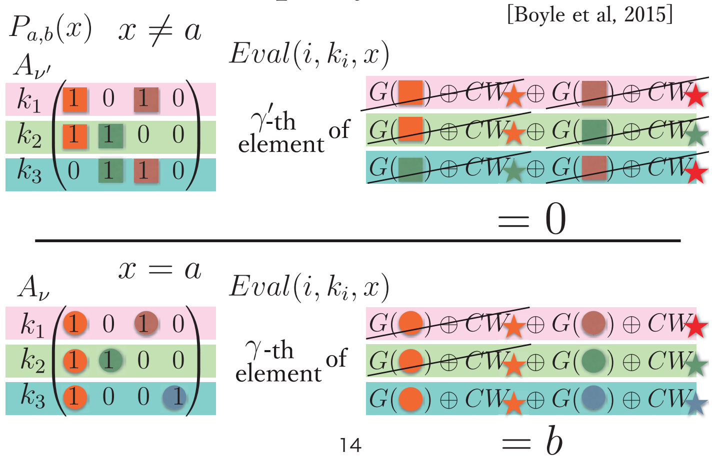
x is viewed as a pair $(i, j) \in \{0,1\}^u \times \{0,1\}^\mu$



13

p-party DPF

[Boyle et al, 2015]



14

Linear Combination of FSS

[Boyle et al, 2015]

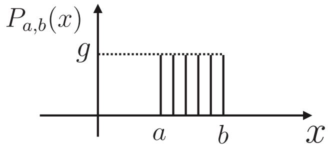
Given FSS schemes for function families $\mathcal{F}$, $\mathcal{G}$ taking $\mathbb{G}_1 \to \mathbb{G}$, there exists an FSS scheme for class $\mathcal{F} + \mathcal{G} := \{f \oplus g \mid f \in \mathcal{F}, g \in \mathcal{G}\}$, with key size equal to $size(\mathcal{F} + \mathcal{G}) = size(\mathcal{F}) + size(\mathcal{G})$, and evaluation time $time(\mathcal{F} + \mathcal{G}) = time(\mathcal{F}) + time(\mathcal{G})$.

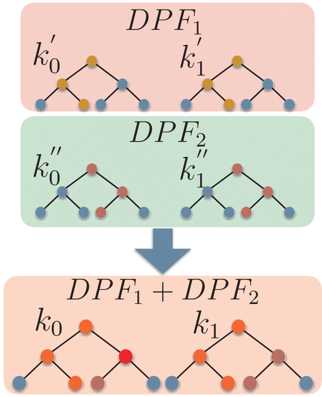
15

Interval function

[Boyle et al, 2015]

$$f_{(a,b)} = \begin{cases} g & a < x < b \\ 0 & \text{else} \end{cases}$$





16

Basis function

Function $f : \{0,1\}^n \to \{0,1\}^m$ can be regarded as a vector space of 2^n

$x \in \{0,1\}^3$	$f(x)$
000	1
001	0
010	1
011	0
100	0
101	1
110	0
111	1

$$f : \{0,1\}^3 \to \{0,1\}$$

$$f : (1,0,1,0,0,1,0,1) \in (\mathbb{F}_2)^{2^3}$$

Vector space has basis vectors.
So function space also has be basis.

$$f(x) = \sum_{i \in \{0,1\}^n} \beta_i h_i$$

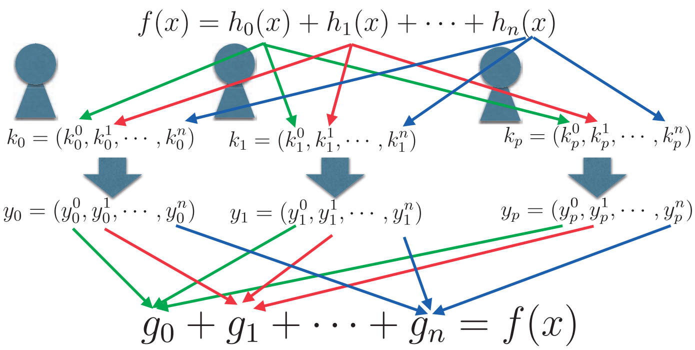
B_i : Coefficients
h_i : Basis functions

17

General FSS by using Basis FSS

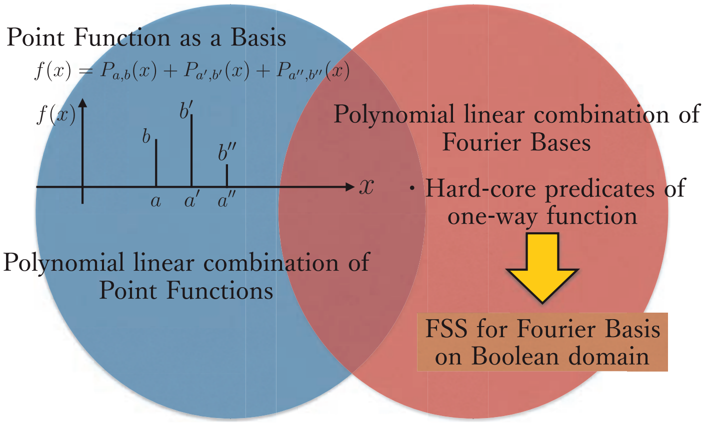
[Ohsawa et al, 2017]

If there exists an FSS scheme for Basis function $h_i(x)$



18

Merit of using other Basis



19

Fourier Translation

$X = \{0, \cdots, T-1\}^n \qquad f : X \to C$

$$\boxed{f(x) = \sum_{a \in \{0,\cdots,T-1\}^n} \hat{f}(a)\chi_a(x)}$$

Fourier Coefficient

$$\hat{f}(a) = \frac{1}{T^n}\sum_{x \in X} f(x)e^{-2\pi i\langle a \cdot x\rangle/T}$$

Fourier Basis

$$\chi_a(x) = e^{2\pi i\langle a \cdot x\rangle/T}$$

20

Fourier Translation on Boolean domain

$X = \{0,1\}^n \qquad f : X \to C$

$$\boxed{f(x) = \sum_{a \in \{0,1\}^n} \hat{f}(a)\chi_a(x)}$$

Fourier Coefficient

$$\hat{f}(a) = \frac{1}{2^n}\sum_{x \in X} f(x)e^{-\pi i\langle a \cdot x\rangle}$$

Fourier Basis

Euler's formula $e^{\pi i} = -1$

$$\chi_a(x) = e^{\pi i\langle a \cdot x\rangle} = (-1)^{\langle a \cdot x\rangle}$$

21

Fourier Basis

$$\chi_a(x) = (-1)^{\langle a \cdot x \rangle} \quad a \in \{0,1\}^2$$

$$(-1)^{\langle (a_0 \oplus a_1)\cdot x \rangle} = (-1)^{\langle a_0 \cdot x \rangle} \cdot (-1)^{\langle a_1 \cdot x \rangle}$$

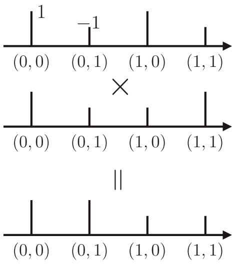
$k_0 = a_0 = (0,1)$

$\oplus$

$k_1 = a_1 = (1,1)$

$\|$

$a = a_0 \oplus a_1 = (1,0)$



22

2-party FSS for the Fourier Basis

[Ohsawa et al, 2017]

$$\chi_a(x) = (-1)^{\langle a \cdot x \rangle}$$

$Gen_2^F(1^\lambda, a)$

a

$k_0 = a_0 \quad k_1 = a_1$

(n,n)-threshold scheme $\bigoplus_{i=0}^{1} a_i = a$

$Eval_2^F(0, k_0, x) = \langle k_0 \cdot x \rangle \quad Eval_2^F(1, k_1, x) = \langle k_1 \cdot x \rangle$

Dec_2^F

$ans = Eval_2^F(0, k_0, x) \oplus Eval_2^F(1, k_1, x)$

$(-1)^{ans} = \chi_a(x)$

$\langle k_0 \cdot x \rangle \oplus \langle k_1 \cdot x \rangle = \langle (k_0 \oplus k_1) \cdot x \rangle$

23

p-party FSS for the Fourier Basis

[Ohsawa et al, 2017]

$$\chi_a(x) = (-1)^{\langle a \cdot x \rangle}$$

$Gen_p^F(1^\lambda, a)$

a

$k_0 = a_0 \quad k_1 = a_1 \cdots k_p = a_p$

(n,n)-threshold scheme $\bigoplus_{i=0}^{p} a_i = a$

$Eval_p^F(0, k_0, x) = \langle k_0 \cdot x \rangle \quad Eval_p^F(1, k_1, x) = \langle k_1 \cdot x \rangle \quad Eval_p^F(p, k_p, x) = \langle k_p \cdot x \rangle$

Dec_p^F

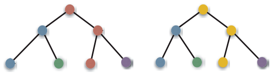
$ans = \bigoplus_{i=0}^{p} Eval_2^F(i, k_i, x)$

$(-1)^{ans} = \chi_a(x)$

24

Conclusion

- Introduction of Function Secret Sharing(FSS)
- Distributed Point Function



$$\begin{array}{c} k_1 \\ k_2 \\ k_3 \end{array} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

- Linear Combination of Basis FSS

$$\text{Secret}\rightarrow f(x) = \sum_{i\in\{0,1\}^n} \beta_i h_i$$

$\uparrow$ Secret

- Distributed Fourier Basis

Easy to Constract

$$\chi_a(x) = (-1)^{\langle a\cdot x\rangle}$$

a

(n,n) -threshold

$$k_0 = a_0 \quad k_1 = a_1 \cdots k_p = a_p \quad \bigoplus_{i=0}^{p} a_i = a$$

25

Ad Hoc PSM Protocols: Secure Computation Without Coordination

Eyal Kushilevitz, Technion
(Joint work with Amos Beimel and Yuval Ishai)

eyalk@cs.technion.ac.il

We study the notion of *ad hoc secure computation*, recently introduced by Beimel et al. (ITCS 2016), in the context of the *Private Simultaneous Messages* (PSM) model of Feige et al. (STOC 2004). In ad hoc secure computation we have n parties that may potentially participate in a protocol but, at the actual time of execution, only k of them, whose identity is *not* known in advance, actually participate. This situation is particularly challenging in the PSM setting, where protocols are non-interactive (a single message from each participating party to a special output party) and where the parties rely on pre-distributed, correlated randomness (that in the ad-hoc setting will have to take into account all possible sets of participants).

We present several different constructions of ad hoc PSM protocols from standard PSM protocols. These constructions imply, in particular, that efficient information-theoretic ad hoc PSM protocols exist for NC^1 and different classes of log-space computation, and efficient computationally-secure ad hoc PSM protocols for polynomial-time computable functions can be based on a one-way function. As an application, we obtain an information-theoretic implementation of *order-revealing encryption* whose security holds for two messages.

We also consider the case where the actual number of participating parties t may be larger than the minimal k for which the protocol is designed to work. In this case, it is unavoidable that the output party learns the output corresponding to each subset of k out of the t participants. Therefore, a "best possible security" notion, requiring that this will be the *only* information that the output party learns, is needed. We present connections between this notion and the previously studied notion of *t-robust PSM* (also known as "non-interactive MPC"). We show that constructions in this setting for even simple functions (like AND or threshold) can be translated into non-trivial instances of program obfuscation (such as *point function obfuscation* and *fuzzy point function obfuscation*, respectively). We view these results as a negative indication that protocols with "best possible security" are impossible to realize efficiently in the information-theoretic setting or require strong assumptions in the computational setting.

Ad Hoc PSM Protocols: Secure Computation without Coordination

Amos Beimel (BGU)

Yuval Ishai (Technion, UCLA)

Eyal Kushilevitz (Technion)

(Appeared in EuroCrypt 2017)

Ad-Hoc MPC [BGIK16]

The (basic) problem:

- Universe of n (honest but curious) parties
- Set of k parties S, not known in advance, participate in the actual computation of some f (say, symmetric).

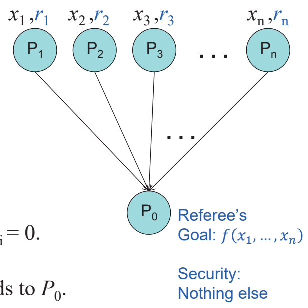
Examples:

- Voting$_k$: output majority vote of k participants.
- Dating: 2 out of n players want to know if they match.

Easy in "standard" MPC model where parties can interact

Private Simultaneous Messages (PSM) model [FKN94,IK97]

- Simple communication pattern
- Shared/Correlated Randomness



Example: SUM

Input: Each P_i is given $x_i \in G$.

Output: P_0 gets Σx_i.

Randomness: $r_1,\ldots,r_{n-1},r_n \in_R G$ s.t. $\Sigma r_i = 0$.

Protocol:

1. Each P_i computes $m_i = x_i + r_i$ and sends to P_0.
2. P_0 computes $\Sigma m_i = \Sigma x_i + \Sigma r_i = \Sigma x_i$.

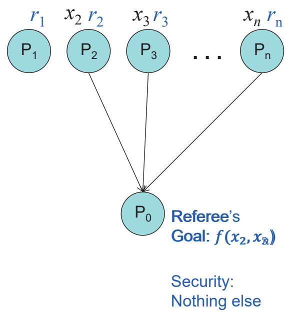
Security: by choice of r_i's.

Why PSM?

- Minimal model – potentially easier to analyze
- Building-block for low-round MPC in the plain model
- A special type of randomized encoding [IK00,IK02]
- Implies Conditional Disclosure of Secrets (CDS)
- …

Ad-Hoc PSM model

- n parties
- Correlated Randomness
- Exactly k parties show up
- Participants not known in advance



Ad-Hoc PSM: assumptions + variants

- Exactly k parties show up.
  If allow $|S| > k$ "best possible security" definition gives Ref f's value on all size-k subsets.
- f symmetric; else can sort by id's or specific f_S, for any S.
- S not known to the parties but will be known to Ref.
  If require anonymity, need anonymous channels.
- Information-Theoretic or Computational security

Rest of the talk

- IT Constructions
  - Warm-up: Ad-hoc PSM protocols for specific functions f
  - Ad-hoc PSM for f from standard PSM for f
  - Ad-hoc PSM for f from standard PSM for a related g

- Connections of other primitives to (variants of) ad-hoc PSM:
  - Order revealing encryption from (IT) ad-hoc PSM
  - NIMPC (t-robust PSM) iff ad-hoc PSM w/best possible security
  - iO exists iff computational ad-hoc PSM w/best possible security
  - (fuzzy) point function obfuscation from ad-hoc PSM for simple f's w/best possible security

Basic Example #1: difference (k=2)

For $S=\{P_i,P_j\}$, $i<j$, output $x_i - x_j$.
(common) Randomness: $r \in_R G$

Protocol:
1. P_i: $m_i=x_i+r$
2. P_0: given m_i,m_j, where $i<j$, outputs $m_i-m_j = x_i-x_j$.

Correctness: √
Security: √

Basic Example #2: SUM_k

Recall PSM protocol for SUM_n:
Randomness: $r_1,\ldots,r_n \in_R G$ s.t. $\Sigma r_i = 0$.
Messages: $m_i=x_i+r_i$.

Ad-hoc PSM for SUM_k:
Randomness: $r_1,\ldots,r_n \in_R G$ s.t. $\Sigma r_i = 0$, as above.
k-of-n secret sharing of each r_j into $\{r_{j,i}\}_{i\in[n]}$
P_i receives r_i and $\{r_{j,i}\}_{j\neq i}$
Messages: P_i sends $m_i=x_i+r_i$ and all its shares $\{r_{j,i}\}_{j\neq i}$
Output of P_0 (on S of size k): for $i\in S$ knows x_i+r_i, for $i\notin S$ can reconstruct r_i (knows k shares) $\Rightarrow$ output $\Sigma_{i\in S}\, x_i+r_i + \Sigma_{i\notin S} r_i = \Sigma_{i\in S}\, x_i$.

Security: for $i\in S$, value of r_i hidden; view of P_0 can be generated from its view in SUM_n protocol where each $P_j\notin S$ has $x_j=0$.

Generic Protocols – 1^{st} attempt

For all T of size k, distribute randomness for PSM_T for f.
Each P_i sends its messages for all T s.t. $i\in T$.

Correctness: for actual set S, referee has all messages of PSM_S.

Problems:

- Complexity overhead of $\binom{n}{k}$ compared to standard PSM for f.
- What if for $T\neq S$ the messages of PSM_T (sent by parties $P_i\in S\cap T$) reveal information?
  - Can be fixed…

Generic Protocols – The case $k=2$

Assume Π_f (standard) PSM for f with players Q_0,Q_1.
Goal: Turn Π_f into ad-hoc PSM Π' that works for any $S=\{P_i,P_j\}$.

Idea: Let one of P_i,P_j simulate Q_0, and the other Q_1.

Problem: Which of Q_0,Q_1 to simulate? (Parties do not know S.)
Solution: Use binary representation $i=(i_1,\ldots,i_{\log n})$. P_i applies Π_f $\log n$ times. In t^{th} iteration simulates Q_{i_t}. For $i\neq j$ exists t s.t. $i_t\neq j_t$.

Problem: When $i_t=j_t$ both simulate same $Q_{i_t}\Rightarrow$ correlated msgs.
Solution: Each P_i sends message of Π_f masked using "key" k_{i_t} and discloses $k_{1-i_t}\Rightarrow$ messages can be un-masked iff $i_t\neq j_t$.

The case $k=2$ (cont.)

Randomness:
For $t=1,\ldots,\log n$: generate randomness $r_{t,0}$, $r_{t,1}$, for PSM Π_f for 2 parties Q_0,Q_1, + random $a_{t,0},b_{t,0}$, $a_{t,1},b_{t,1}\in_R \mathbb{F}_p$.
Give $a_{t,0},b_{t,0}$, $a_{t,1},b_{t,1}$ and r_{t,i_t} to P_i.

Messages of P_i:
For $t=1,\ldots,\log n$: P_i simulates Q_{i_t} message $m_{t,i}$ in Π_f on (x_i,r_{t,i_t}).
It sends masked message $m_{t,i}+a_{t,i_t}*i+b_{t,i_t}$ and also $a_{t,1-i_t}$, $b_{t,1-i_t}$.

Correctness: For t s.t. $i_t\neq j_t$ P_0 has $a_{t,0}$, $b_{t,0}$, $a_{t,1},b_{t,1}$ and can un-mask $m_{t,0}$, $m_{t,1}$ to compute $f(x_i,x_j)$.
Security: Since $i\neq j$ then messages hidden (2-wise ind.).
Complexity: O($\log n$) overhead in randomness and communication.

Generic Protocols – General k

Idea: Use perfect hash family to select which P_i simulates each Q_j.
(A family $H=\{h: [n]\to[k]\}$ s.t $\forall S$ of size k, $\exists$ 1-1 func. $h\in H$.)

Perfect Hash facts:

- For k=2, the log n bit functions form such H.
- Explicit and probabilistic constructions.
  E.g., probabilistically $|H| \approx e^k k\cdot\log n$ suffices.

Idea (cont.): Run original PSM Π_f for each $h\in H$. Mask messages with k-wise independent keys ($A_{h,j}$, $j\in[k]$) + shares of (k-1)-of-n sharing of other keys. P_0 can remove mask iff h is 1-1 on S.

Complexity: overhead of $\approx |H|$ (good for "small" k)

Generic Protocols from a PSM for a related func.

Given f: $X^k \to Y$, define g: $(X\cup\{\perp\})^n \to Y\cup\{\perp\}$:
if #non-$\perp$ inputs is k, then output f on those inputs; otherwise $\perp$.

Assume Π_g (standard) PSM for g. Construct ad-hoc PSM Π_f for f.
Randomness: $r_1,\ldots,r_n$ for Π_g.
Let $m_{\perp,j}$ = message of P_j in Π_g on $(\perp,r_j)$.
Let $\{m_{\perp,j,i}\}_i$ = shares in a k-out-of-n sharing of $m_{\perp,j}$.
Give P_i randomness r_i and shares $\{m_{\perp,j,i}\}_j$.
Message of P_i: its Π_g message $m_{x_i,i}$ on (x_i,r_i) + its shares $\{m_{\perp,j,i}\}_{j\neq i}$.

Correctness: For S of size k, P_0 has $m_{x_i,i}$ for $i\in S$ + can reconstruct all $m_{\perp,j}$ for $j\notin S$ $\Rightarrow$ Output of Π_g is the correct answer.
Security: cannot reconstruct $m_{\perp,j}$ for $j\in S$.
Complexity: $O(n)$ overhead due to secret-sharing.

Corollaries

- Every function g has a PSM (with complexity $|X|^n$)

Cor: Every function f has an ad-hoc PSM

- If g has a poly. size (modular) branching program, then it has an efficient PSM
- If f has poly. size (modular) branching program, then so does the corresponding g

Cor: If f has a poly. size (modular) branching program, then f has an *efficient* ad-hoc PSM

Order Revealing Encryption (ORE)

ORE [AKSX04, BCLO09, BCO11]:

- A private-key encryption equipped with a comparison
  - A public procedure Comp
    - $c_1 = \mathrm{Enc}(x_1, k), c_2 = \mathrm{Enc}(x_2, k)$
    - $\mathrm{Comp}(c_1, c_2) = 1$ iff $x_1 \leq x_2$
  - Encryption does not leak additional information

Ad-Hoc PSM $\Rightarrow$ ORE

- Use ad-hoc PSM for the Greater-Than (GT) function with $n = 2^\lambda$ parties and $k = 2$
  - λ – security parameter
  - GT has a small branching program $\Rightarrow$ (IT) PSM
- Key generation: pick randomness for the ad-hoc PSM
- Encryption of $x \in \{0,1\}^\ell$:
  - Choose a random party P_i, generate r_i
  - Encryption $c = (i$, message of P_i on $(x, r_i))$
- Comparing c_1, c_2: use $(2,n)$ ad-hoc computation of GT
- IT-Security for two messages: if c_1, c_2 use different parties
- Complexity: $\log n \cdot \mathrm{poly}(\ell) = \lambda \cdot \mathrm{poly}(\ell)$

Best-possible secure ad-hoc PSM vs. NIMPC

NIMPC [BGIKMP14] = t-robust-PSM = A PSM that can tolerate a coalition of P_0 with $\leq t$ parties.

NIMPC also uses best possible security notion.

Def: (k,t,n)-ad hoc PSM = best possible security $\forall T$ s.t. $k \leq |T| \leq t$.

We prove:

- $(n/2, n/2+t, n)$ ad-hoc PSM for f $\Rightarrow$ t-robust PSM for f with same complexity.
- t-robust PSM for some related $3n$-argument g' $\Rightarrow$ (k,t,n) ad-hoc PSM for f with O(n) overhead.

Computational Ad-Hoc PSM: Remarks

- [BGIK16]: Multi-Input Functional Encryption (MIFE) $\Rightarrow$ Distribution Design $\Rightarrow$ Computational best-possible-security ad-hoc PSM (w/indistinguishability def.)
- Best-possible-security ad-hoc PSM $\Rightarrow$ NIMPC $\Rightarrow$ iO [BGIKMP14]
- Best-possible-security ad-hoc $(n,2n,2n)$ PSM for AND
  $\Rightarrow$ point function obfuscation
- Best-possible-security ad-hoc $(n,2n,2n)$ PSM for Threshold func.
  $\Rightarrow$ fuzzy point function obfuscation

Ad-hoc PSM for AND $\Rightarrow$ Point Function Obfuscation

- For a point $x = (x_1, \ldots, x_n)$, define $I_x(y) = 1$ iff $y = x$.
- Π – $(n,2n,2n)$ ad-hoc PSM for AND
- Obfuscating point function I_x:
  - Generate randomness $r_1, \ldots, r_n$ for Π
  - Let $m_{i,b}$ = message of P_i on (b, r_i)
  - $\forall_i$ let $a_{i,x_i} = m_{i,1}$ and $a_{i,\bar{x}_i} = m_{i,0}$
  - Obfuscation: $a_{1,0}, a_{1,1}, \ldots, a_{n,0}, a_{n,1}$
  - Computing $I_x(y)$: ad-hoc decoding from $a_{1,y_1}, \ldots, a_{n,y_n}$

Summary

We present concrete and generic constructions of Ad-Hoc PSM protocols.

- Every function has an ad-hoc PSM
- All functions that are known to have an efficient PSM have an efficient ad-hoc PSM

- Connections to ORE, NIMPC, iO, point function obfuscation

Obvious open problems: more protocols, improved complexity and parameters, more connections with other primitives.

- Best possible security

Thank you!

Secure Message Transmission against Rational Adversaries

Takeshi KOSHIBA (Joint work with Maiki Fujita)

Waseda University

tkoshiba@waseda.jp

Secure Message Transmission (SMT) is a two-party cryptographic scheme by which a sender securely and reliably sends messages to a receiver using n channels. Suppose that an adversary corrupts at most t out of n channels and makes eavesdropping or tampering over the corrupted channels. It is known that if $t < n/2$ then the perfect SMT (PSMT) in the information-theoretic sense is achievable and if $t \geq n/2$ then no PSMT scheme is possible to construct. If we are allowed to use a public channel in addition to the normal channels, we can achieve the almost reliable SMT (ARSMT), which admits transmission failures of small probability, against $t < n$ corruptions. In the standard setting in cryptography, the participants are classified into honest ones and corrupted ones: every honest participant follows the protocol but corrupted ones are controlled by the adversary and behave maliciously. As a real setting, the notion of rationality in the game theory is often incorporated into cryptography. In this paper, we first consider "rational adversary" who behaves according to his own preference in SMT. We show that it is possible to achieve PSMT even against any $t < n$ corruptions under some reasonable settings for rational adversaries.

In the above, we consider settings where the rational entity is a single adversary. It means that the adversary's behavior is determined by his own preference (utility). We also consider the case where there are two independent rational adversaries. We show some cases where the Nash equlibria plays an important role to design SMT protocols secure against two independent ratinal adversaries.

References

[1] D. Dolev, C. Dwork, O. Waarts, and M. Yung, Perfectly secure message transmission, *J. ACM* 40(1):17–47, 1993.

[2] M. Franklin and R. N. Wright, Secure communication in minimal connectivity models, *J. Cryptology* 13(1):9–30, 2000.

[3] H. Shi, S. Jiang, R. Safavi-Naini, and M. A. Tuhin, On optimal secure message transmission by public discussion, *IEEE Transactions on Information Theory* 57(1):572–585, 2011.

[4] J. Halpern and V. Teague, Rational secret sharing and multiparty computation, in *Proc. 36th Annual ACM Symposium on Theory of Computing*, pp.623–632, ACM, 2004.

[5] A. Groce, J. Katz, A. Thiruvengadam, and V. Zikas, Byzantine agreement with a rational adversary, in *Proc. ICALP 2012*, Vol.2, LNCS 7392, pp.561–572, Springer, 2012.

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling
13th June 2017
AirIMaQ, Momochi, Fukuoka, Japan

Secure Message Transmission against Rational Adversaries

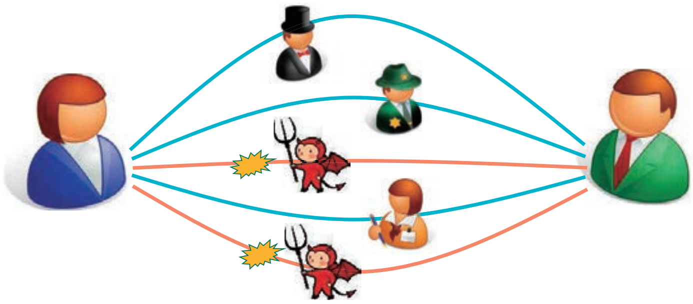
Takeshi Koshiba (Waseda Univ., Japan)

Joint work with Maiki Fujita

WASEDA University

Secure Message Transmission

Dolev, Dwork, Waarts & Yung (FOCS 1990 & JACM 1993)



Secure Message Transmission

Setting

- Several channels are available between Alice and Bob
- Some of them are corrupted by (adversarial) Eve
  - Eve can alter messages over the corrupted channels

Goal

- Alice sends her messages to Bob
  - ***Correctness***: Messages Bob receives are the same as ones Alice sends
  - ***Privacy***: Eve cannot get any information on the messages

Secure Message Transmission

Why SMT ?

- In the standard setting of multi-party secure computation,
  - Each player is a node of a complete graph
  - Between any two players, there is a secure channel represented as an edge
- In an incomplete graph (i.e., network),
  - Alice (on a node) and Bob (on another node) want to exchange messages
  - If Alice and Bob execute SMT, a virtual secure channel can be assumed

Possibilities and Limitations of SMT

Eve corrupts t out of n channels

- Perfect Case (Perfect SMT (PSMT))

$n > 2t$: efficient PSMT protocol

e.g., Kurosawa & Suzuki (EuroCrypt 2008 & IEEEIT 2009)

$n \leq 2t$: impossible (Dolev, Dwork, Waarts & Yung 1993)

- Almost Reliable Case (Bob receives a wrong message with small prob.)

$n \leq 2t$: still impossible (Franklin & Wright, EuroCrypt 1998 & JoC 2000)

Public Channel

Public channel is an authenticated one

- No secrecy
- Cannot be tampered

- Almost Reliable SMT (ARSMT) with public channel

$n > t$: 3-round protocol

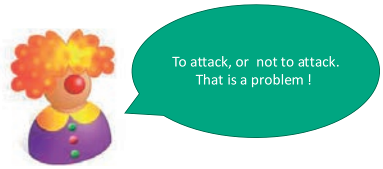
(Shi, Jiang, Safavi-Naini & Tuhin, ISIT 2009 & IEEEIT 2011)

Rational Adversaries

Cryptographic adversaries attack on protocols
without considering any risk

Rational adversaries attack on protocols
if the attack is economically reasonable

Rational Adversaries



If I succeed in the attack, I will get $1,000,000
But if I fail, I must pay a fine of $500,000
Hmm…

Game Theory in Cryptography

Halpern & Teague (STOC 2004)

- In Shamir's (n, n)-threshold secret sharing,
  - After n-1 participants submit their shares, the n^{th} participant might stop to submit his share to monopolize the secret
- To prevent this kind of malicious behavior, which may be a consequence of his preference, the notion of *Nash equilibrium* was introduced to design secure protocols
  - Design a protocol so that choosing "obeying the protocol" for all the participants is Nash equilibrium

Game Theory

- Mathematical models of conflict and cooperation among rational decision-makers

	Player 2 Chooses C	Player 2 Chooses D
Player 1 Chooses A	5, 2	-1, -1
Player 1 Chooses B	0, 0	2, 5

Payoff matrix of a (2-player, 2-strategy) game

Nash Equilibrium

- The Prisoner's Dilemma

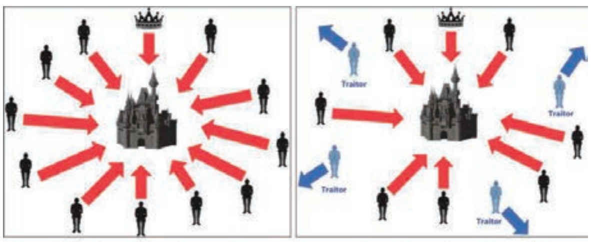
	Prisoner 2 Silent	Prisoner 2 Betray
Prisoner 1 Silent	-1, -1	-10, 0
Prisoner 1 Betray	0, -10	-5, -5

If each prisoner betray the other, each of them will serve 5 years in prison

If both prisoners remain silent, both of them will only serve 1 year in prison

If P1 betrays P2 but P2 remains silent, P1 will be set free and B will serve 10 years in prison

Byzantine Agreement



Coordinated Attack Leading to Victory | Uncoordinated Attack Leading to Defeat

By Debraj Ghosh : How the Byzantine General Sacked the Castle : A Look Into Blockchain

Byzantine Agreement

n Generals want to agree "attack" or "withdraw"
even if there exist t of n faulty Generals

$n > 3t$: protocols for solving the Byzantine Agreement (BA)
$n \leq 3t$: impossible
$n \leq 2t$: impossible in any setting (e.g., a PKI setting)

Rationality in Byzantine Agreement

$n > t$: a perfectly secure protocol against rational adversaries
(Groce, Katz, Thiruvengadam & Zikas, ICALP 2012)

Eve can corrupts t out of n Generals
- Whether Eve corrupts or not depends on *expected* payoff value
- The simplest setting in Game Theory

Rationality in Secure Message Transmission

- Case 1
  - Eve can corrupt t ouf of n channels
  - Whether Eve corrupts or not depends on the expected payoff
    (as in Rational Byzantine Agreement)
- Case 2
  - Two independent rational adversaries : Eve & Eva

Rationality Models (for Case 1)

- **Timid Model**

Eve is afraid of loss of the reliability or being exposed her dishonesty

For example, she owns a channel and gains the usage fee from users. If she loses the reliability of the channel, then her gain may be decreased or she may be accused of her behavior.

- **Conservative Model**

Eve is afraid of the environmental degradation

The environmental degradation means that the traffic environment could be difficult to maintain because of the detection of some dishonesty. Thus, Eve is afraid of being specified corrupted channels or the protocol abortion.

Results

- Case 1 (Single Adversary)

PSMT with public channel in Timid Model, if $n > t$

PSMT in Conservative Model, if $n > t$

- Case 2 (Independent Two Adversaries)

PSMT if $n > t$ and some condition holds

c.f.

In the standard setting, PSMT only if $n > 2t$ even with public channel

Strategies of Rational Eve

- Eve can tamper (T) a channel or eavesdrop (E) on the channel
- Her possible actions are T&E, T only, E only, and nothing

- Assume that passive attack (i.e. eavesdropping) is not exposed
  - No reason why Eve stops eavesdropping !

- Thus, she chooses "T&E" (σ_a : active) or "E only" (σ_p : passive) for her action

Utilities of Rational Eve

- Several viewpoints
  - The result of message transmission
    - The same message is delivered (u_s)
    - A different message is delivered (u_f)
    - Aborted (u_a)
  - Eve's points
    - Acquisition of the secret message (u_q)
    - Detection of corrupted channels (u_d)

Rationality Models and Utilities

- **Timid Model**

Eve is afraid of loss of the reliability or being exposed her dishonesty

$$\min\{ u_a, u_f \} > u_s, \quad u_q > 0, \quad u_d < 0$$

- **Conservative Model**

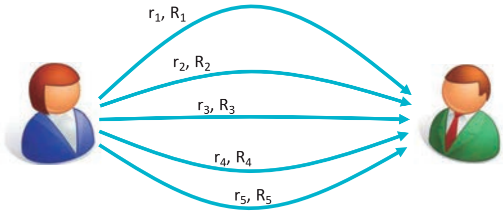
Eve is afraid of the environmental degradation

$$u_f, > u_s > u_a, \quad u_q > 0, \quad u_d = 0$$

Protocol 1 (against Timid Eve)

- Shi et al's 3-round ARSMT protocol with public channel works as PSMT protocol against Timid rational adversaries
- It uses 2^{1-2L}-almost strongly universal hash functions
  - L : length of hash values
  - Pr[$h(x_1) = y_1$ & $h(x_2) = y_2$] $\leq 2^{1-2L}$

Protocol 1 : 1^{st} Round



Bob receives r_i' & R_i' over i^{th} channel

h_i : random universal hash

Bob computes $T_i'=R_i'\oplus h_i(r_i')$ if he can and sets $b_i=0$; Otherwise set $b_i=1$

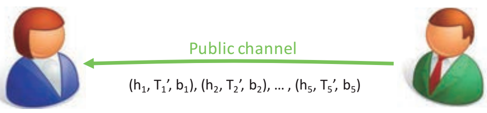
Protocol 1 : 2^{nd} Round

Alice ignores i^{th} channel if $b_i = 1$

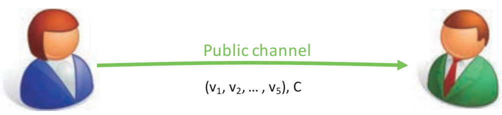
If $b_i = 0$, Alice computes $T_i=R_i \oplus h_i(r_i)$

If $T_i = T_i'$ then set $v_i=1$ else set $v_i=0$

Alice computes $C=m\oplus(\oplus_{v_i=1} R_i)$



Protocol 1 : 3^{rd} Round



Bob computes $m=C\oplus(\oplus_{v_i=1} R_i)$

Protocol 1 : Properties

- Secrecy
  - Protocol 1 is perfect
- Correctness
  - Protocol 1 delivers a different message with prob. $(n-1)2^{1-L}$

Expected Payoff of Timid Eve

- If Eve takes σ_a as her action

$$u(\sigma_a) = (n-1)2^{1-L}u_f + (1-(n-1)2^{1-L})(u_s + u_d)$$

- If Eve takes σ_p

$$u(\sigma_p) = u_s$$

Thm 1

Suppose *n* > *t*

If

$L > 1 + \log ((n - 1)(u_f - u_s - u_d)/(-u_d))$

then

Protocol 1 is PSMT (with public channel) against Timid rational adversary

Protocol 2 (against Conservative Eve)

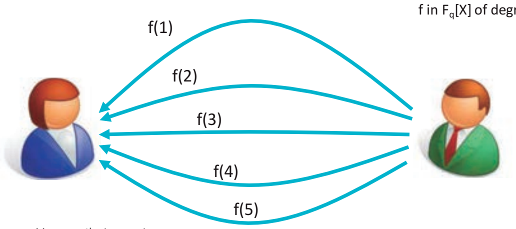
- It does not use public channel
- The basic part consists of 2 rounds and Protocol 2 repeats it

- Protocol 2 is a probabilistic mixture of 2 sub-procedures
  - With probability p, it executes message transmission
  - With probability 1-p, it checks whether some channels are tampered

- The expected number of repetitions is 1/p

Protocol 2 : Round 1

Bob chooses a random polynomial f in $F_q[X]$ of degree n-1



Alice receives g(i) over i^{th} channel

If possible, Alice recovers g by interpolation and computes s = g(0) + m

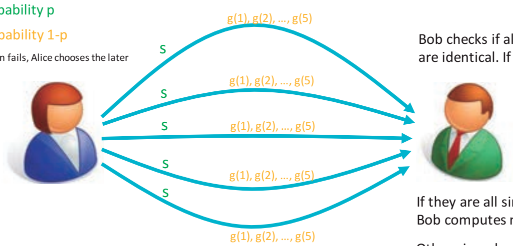
Protocol 2 : Round 2

With probability p

With probability 1-p

If interpolation fails, Alice chooses the later

Bob checks if all the received data are identical. If "no", abort.



If they are all singleton s', Bob computes m' = f(0) – s' and halt

Otherwise, check if the received data coincide with f(1), f(2), …, f(5)

Restart if "yes" ; Abort if "no"

Expected Payoff of Conservative Eve

- If Eve takes σ_a as her action

$$u(\sigma_a) = p\, u_f + (1 - p)\, u_a$$

- If Eve takes σ_p

$$u(\sigma_p) = u_s$$

Thm 2

Suppose $n > t$

If

$p > (u_a - u_s)/(u_a – u_f)$

then

Protocol 2 is PSMT against Conservative rational adversary

Rational Eve & Eva

- In case of two independent adversaries, there are many possible models
- We take a case where Eve and Eva are hostile to each other

Utilities of Eve and Eva

- The result of message transmission
  - The same message is delivered ($u_{i,s}$)
  - A different message is delivered ($u_{i,f}$)
  - Aborted ($u_{i,a}$)
- Adv i's points
  - Detection of channels corrupted by Adv i ($u_{i,d}$)
  - Adv i's acquisition of the secret message ($u_{i,q}$)
  - Detection of channels corrupted by the opponent ($u_{i,od}$)
  - The opponent's acquisition of the secret message ($u_{i,oq}$)

Hostile Model

- $u_{i,s} < u_{i,f}$
  - Transmission of a different message is better than that of the same message
- $u_{i,q} + u_{i,oq} > 0$
  - By Eve or Eva, the acquisition of the message is nice
- $u_{i,d} < 0$
  - They hate the detection of channels corrupted by them
- $u_{i,oq} < 0$
  - They hate the acquisition of the message by the opponent
- $u_{i,q} > 0$
  - The acquisition of the message is good
- $u_{i,od} > 0$
  - The detection of channels corrupted by the opponent is a kind of windfall profit

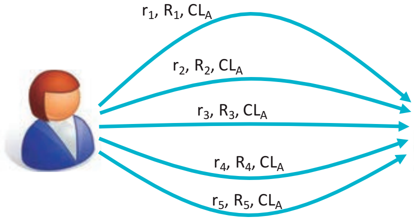
Protocol 3 in Hostile Model

- Use a slightly modified version of Protocol 1 iteratively
- Alice and Bob have their own CLs (corruption lists) and update them if necessary
  - Initial CLs are empty
  - If a channel is added to CL, the channel is not used any more. Thus the number of available channels decreases
- If CLs are updated, Protocol 3 continues the iteration

- There exists an iterated dominant strategy which leads to an equilibrium

Protocol 3: Initial Round (or Alice→Bob in Phase A)

Case : CL_A is updated



Bob checks if all the received data are of equal length and received CLs are identical with CL_B

If "yes", go to Phase B

Classify channels based on the length. Add all the channels not in the major group into CL_B

Bob searches the received CLs for channels not in CL_B. If he finds them, add them into CL_B

Protocol 3: Alice←Bob in Phase A

Case : CL_B is updated

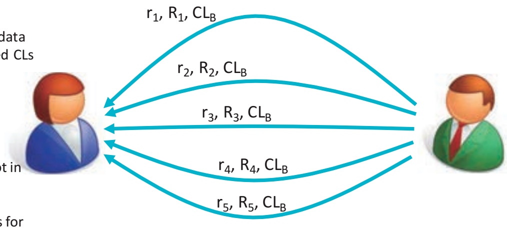
Alice checks if all the received data are of equal length and received CLs are identical with CL_A

If "no", execute below

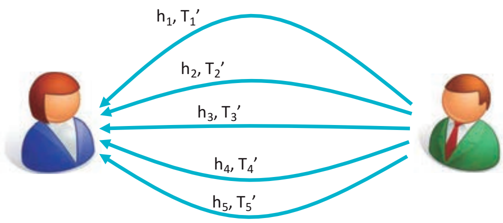
Classify channels based on the length. Add all the channels not in the major group into CL_A

Alice searches the received CLs for channels not in CL_A. If he finds them, add them into CL_A

Go to Alice→Bob in Phase A



Protocol 3: Alice←Bob in Phase B



Bob receives r_i' & R_i' over alive i^{th} channel

h_i : random universal hash

Bob computes $T_i'=R_i' \oplus h_i(r_i')$

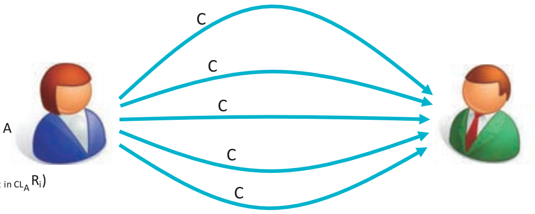
Protocol 3: Alice→Bob in Phase B

Alice computes $T_i=R_i \oplus h_i(r_i)$

If $T_i \neq T_i'$ then add i into CL_A

If CL_A is updated, go to Phase A

Otherwise,
Alice computes $C=m\oplus(\oplus_{i \text{ not in } CL_A} R_i)$



Consistency check as in Initial Round

If inconsistent go to Phase A else
Bob computes $m'=C'\oplus(\oplus_{i \text{ not in } CL_B} R_i')$

Protocol 3 : Properties

- Secrecy
  - Protocol 3 has perfect secrecy (in the standard crypto setting)
    - if $n > 2t$ and Adv can tamper and eavesdrop, or
    - if $n > t$ and Adv can eavesdrop only
- Reliability
  - Protocol 3 fails in the message transmission w.p. $(n-1)2^{-L}$
    - if $n > 2t$ and Adv can tamper and eavesdrop
  - Protocol 3 always succeeds in the message transmission
    - if $n > t$ and Adv can eavesdrop only

Protocol 3 is PSMT if $n > t$ and Adv can eavesdrop only

Iterated Dominance

- σ_p : a strategy
- σ_{-p} : other strategies other than σ_p

σ_p is iterated dominant if

- $u_A(\sigma_{-p}, \sigma_p) < u_A(\sigma_p, \sigma_p)$,
- $u_B(\sigma_p, \sigma_{-p}) < u_B(\sigma_p, \sigma_p)$, and
- $u_A(\sigma_{-p}, \sigma_{-p}) < u_A(\sigma_p, \sigma_{-p})$ or $u_B(\sigma_{-p}, \sigma_{-p}) < u_B(\sigma_{-p}, \sigma_p)$

Thm 3

There exists a setting in Hostile Model where "eavesdropping only" is the iterated dominant strategies for Eve and Eva in Protocol 3

That is, Protocol 3 is PSMT in Hostile Model

Conclusion

- We have introduced "rationality" in Secure Message Transmission
- Since rational adversaries are weaker than cryptographic adversaries, the bound on the number of corrupted channels can be better than the standard cryptographic setting

See ia.cr/2017/309 for the first half; the second half in preparation

Optimized Honest-Majority MPC for Malicious Adversaries - Breaking the 1 Billion-Gate Per Second Barrier

Kazuma OHARA (Joint work with Toshinori ARAKI, Assi BARAK, Jun FURUKAWA, Yehuda LINDELL, Ariel NOF, Adi WATZMAN, Or WEINSTEIN.)

NEC Corporation

k-ohara@ax.jp.nec.com

Secure multiparty computation enables a set of parties to securely carry out a joint computation of their private inputs without revealing anything but the output. In the past few years, the efficiency of secure computation protocols has increased in leaps and bounds. However, when considering the case of *security in the presence of malicious adversaries* (who may arbitrarily deviate from the protocol specification), we are still very far from achieving high efficiency.

In this talk, we consider the specific case of three parties and an honest majority. We provide general techniques for improving efficiency of cut-and-choose protocols on multiplication triples and utilize them to significantly improve the recently published protocol of Furukawa et al. (at Eurocrypt'17). We reduce the bandwidth of their protocol down from 10 bits per AND gate to 7 bits per AND gate, and show how to improve some computationally expensive parts of their protocol. Most notably, we design cache-efficient shuffling techniques for implementing cut-and-choose without randomly permuting large arrays (which is very slow due to continual cache misses). We provide a combinatorial analysis of our techniques, bounding the cheating probability of the adversary.

Our implementation achieves a rate of approximately *1.15 billion AND gates* per second on a cluster of three 20-core machines with a 10Gbps network. Thus, we can securely compute 212,000 AES encryptions per second (which is hundreds of times faster than previous work for this setting). Our results demonstrate that high-throughput secure computation for *malicious adversaries* is possible.

\Orchestrating a brighter world NEC

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

Optimized Honest Majority MPC for Malicious Adversaries
– Breaking the Billion-Gate Per-Second Barrier

2017/06/13
Kazuma Ohara (NEC)

Joint work with
Toshinori Araki, Jun Furukawa (NEC)
Assi Barak, Yehuda Lindell, Ariel Nof ,
Adi Watzman, Or Weinstan (Bar-Ilan University)

Secure Multiparty Computation (MPC)

- Compute on private inputs without revealing anything but the output
- Applications
  - Protect credentials and biometrics
  - Run learning algorithm on distributed databases (e.g., health)
  - Secure SQL
  - Compare DNA samples without revealing them
- Two models
  - **Semi-honest**: protection against inadvertent leakage and more
  - **Malicious**: protection against arbitrary attacks (required in many cases)

Secure Multiparty Computation

- Powerful in theory:
  Any functionality can be computed
- Secure multiparty computation holds **great promise**, but can we fulfill that promise?
- Can we achieve speeds of MPC that is fast enough for applications in practice?

CYBERNETICA

PARTISIA

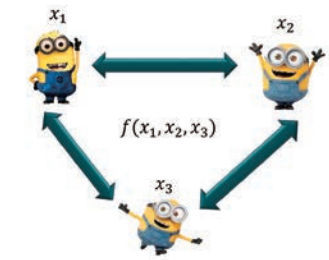
DYADIC

- We can solve some problems of interest today, but medium to large scale secure computation seems beyond reach
  - This is especially true for **malicious adversaries**

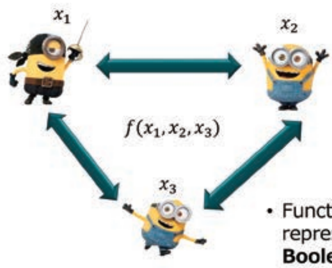
This Work

- Large scale secure three-party computation is practical, even in the presence of a malicious adversary.
  - Carried out highly optimized implementation of MPC and obtained **over 1-billion AND-gate/sec**
    - it's close to limit of physical network bandwidth on the framework we deployed.

Secure 3-Party Computation with an Honest Majority



Secure 3-Party Computation with an Honest Majority



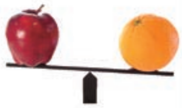
- Functionality is represented by a **Boolean circuit**
- Security with **abort**

Low Latency VS. High-Throughput

Low Latency

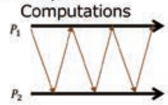
- Constant rounds of communication

"the garbled-circuit approach"



High-Throughput

- Low bandwidth
- Simple Computations



"the secret-sharing approach"

Semi-honest 3-party MPC [Araki et al., CCS'16]

- "High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority"
  Toshinori Araki, Jun Furukawa (NEC), Yehuda Lindell, Ariel Nof (Bar-Ilan University) and Kazuma Ohara (NEC)

- Evaluates Boolean circuit
  - can extend for arithmetic circuit
- Based on (a kind of) additive secret sharing
  - No cryptographic protocol without PRG
  - No Communication for XOR-gate/NOT-gate
  - Only 1-bit Communication for AND-gate

- From next page:
  - Secret sharing
  - MPC for XOR-gate
  - MPC for AND-gate

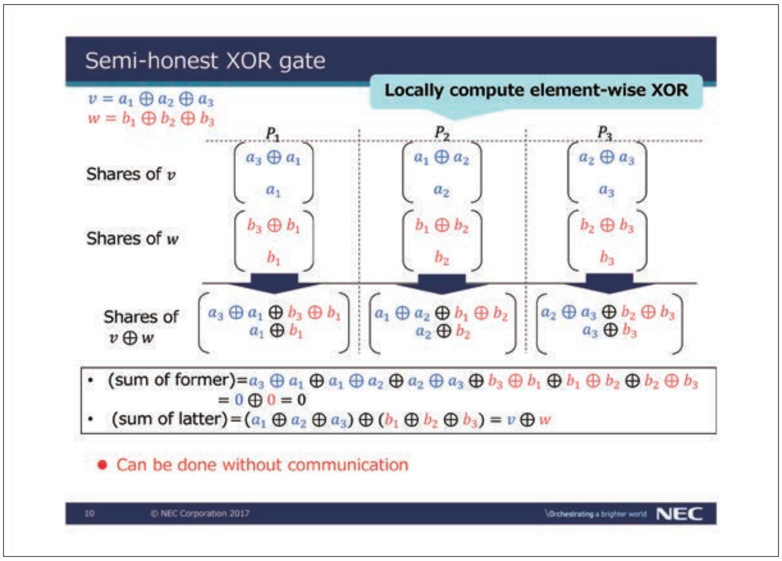
Secret Sharing

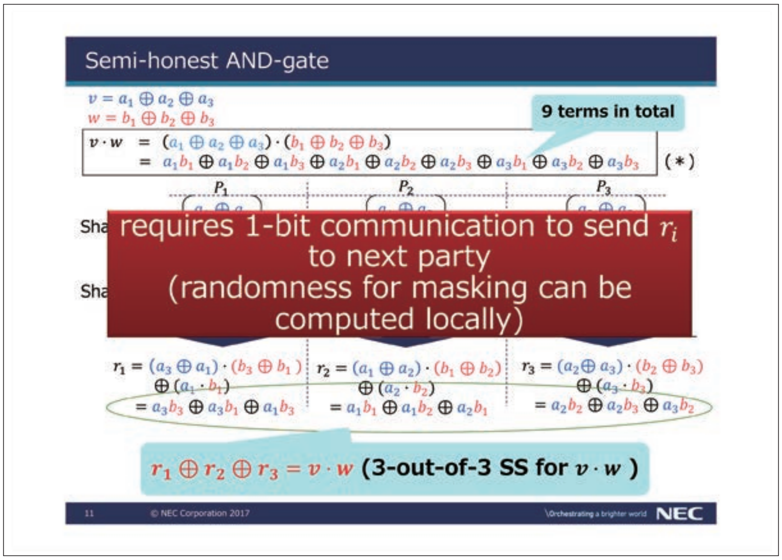
- Share generation : secret $v \in \{0,1\}$
  - a_1, a_2, a_3 such that $a_1 \oplus a_2 \oplus a_3 = v$
  -

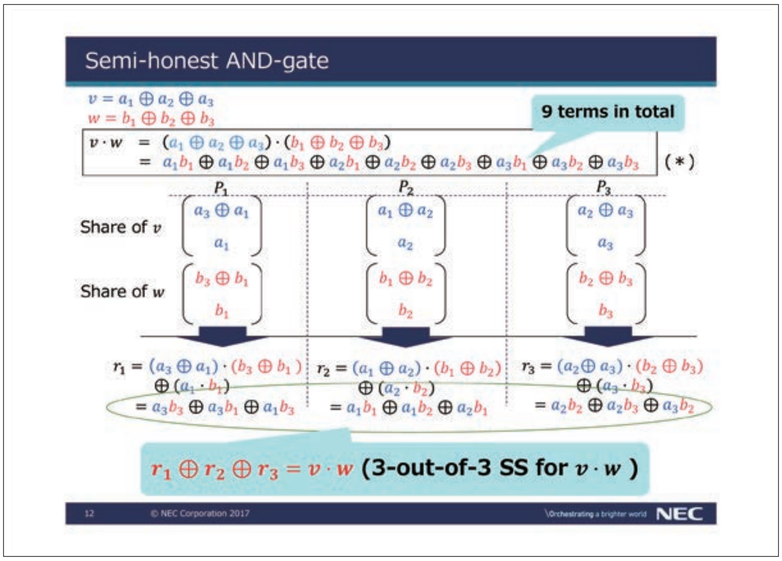
	Share for P_1	Share for P_2	Share for P_3
①	$a_3 \oplus a_1$	$a_1 \oplus a_2$	$a_2 \oplus a_3$
②	a_1	a_2	a_3

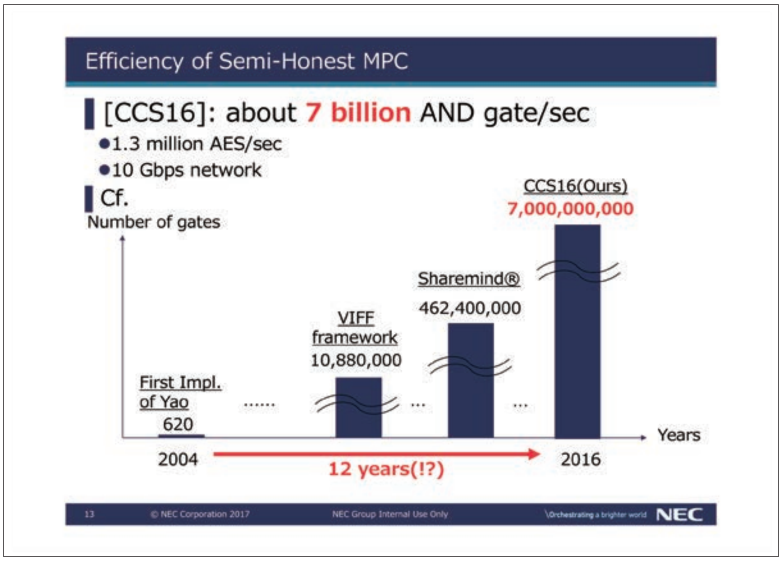
Each party has two elements of (a_1, a_2, a_3)

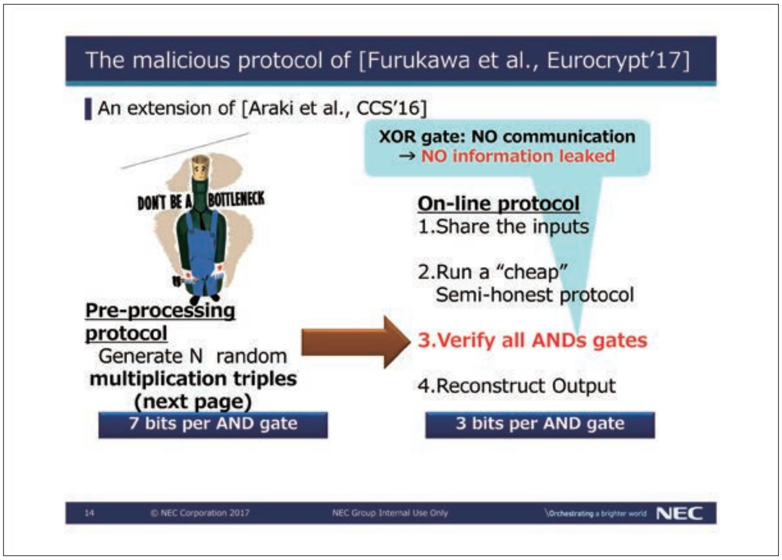
- Reconstruction
  - From any combination of two share, (a_1, a_2, a_3) can get.
- Properties
  - The sum of former part is equal to 0.
    - $x_1 \oplus x_2 \oplus x_3 = a_3 \oplus a_1 \oplus a_1 \oplus a_2 \oplus a_2 \oplus a_3$
  - The sum of latter part is equal to v (secret).
    - $a_1 \oplus a_2 \oplus a_3 = v$

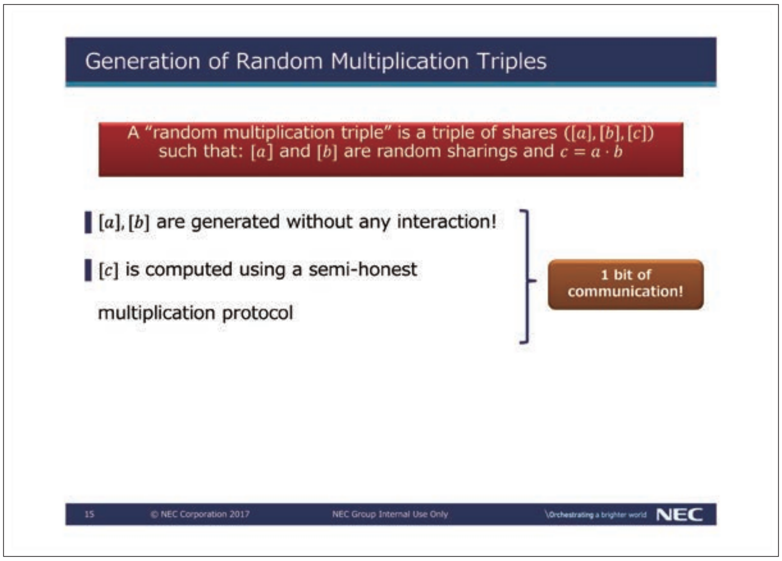

Semi-honest XOR gate
Locally compute element-wise XOR
$v = a_1 \oplus a_2 \oplus a_3$
$w = b_1 \oplus b_2 \oplus b_3$
P_1 P_2 P_3
Shares of v
$a_3 \oplus a_1$, a_1 | $a_1 \oplus a_2$, a_2 | $a_2 \oplus a_3$, a_3
Shares of w
$b_3 \oplus b_1$, b_1 | $b_1 \oplus b_2$, b_2 | $b_2 \oplus b_3$, b_3
Shares of $v \oplus w$
$a_3 \oplus a_1 \oplus b_3 \oplus b_1$, $a_1 \oplus b_1$ | $a_1 \oplus a_2 \oplus b_1 \oplus b_2$, $a_2 \oplus b_2$ | $a_2 \oplus a_3 \oplus b_2 \oplus b_3$, $a_3 \oplus b_3$
• (sum of former)$= a_3 \oplus a_1 \oplus a_1 \oplus a_2 \oplus a_2 \oplus a_3 \oplus b_3 \oplus b_1 \oplus b_1 \oplus b_2 \oplus b_2 \oplus b_3$
$= 0 \oplus 0 = 0$
• (sum of latter)$= (a_1 \oplus a_2 \oplus a_3) \oplus (b_1 \oplus b_2 \oplus b_3) = v \oplus w$
● Can be done without communication
10 © NEC Corporation 2017 \Orchestrating a brighter world NEC


Semi-honest AND-gate
$v = a_1 \oplus a_2 \oplus a_3$
$w = b_1 \oplus b_2 \oplus b_3$
9 terms in total
$v \cdot w = (a_1 \oplus a_2 \oplus a_3) \cdot (b_1 \oplus b_2 \oplus b_3)$
$= a_1b_1 \oplus a_1b_2 \oplus a_1b_3 \oplus a_2b_1 \oplus a_2b_2 \oplus a_2b_3 \oplus a_3b_1 \oplus a_3b_2 \oplus a_3b_3$ (*)
P_1 P_2 P_3
requires 1-bit communication to send r_i to next party
(randomness for masking can be computed locally)
$r_1 = (a_3 \oplus a_1) \cdot (b_3 \oplus b_1) \oplus (a_1 \cdot b_1) = a_3b_3 \oplus a_3b_1 \oplus a_1b_3$
$r_2 = (a_1 \oplus a_2) \cdot (b_1 \oplus b_2) \oplus (a_2 \cdot b_2) = a_1b_1 \oplus a_1b_2 \oplus a_2b_1$
$r_3 = (a_2 \oplus a_3) \cdot (b_2 \oplus b_3) \oplus (a_3 \cdot b_3) = a_2b_2 \oplus a_2b_3 \oplus a_3b_2$
$r_1 \oplus r_2 \oplus r_3 = v \cdot w$ (3-out-of-3 SS for $v \cdot w$)
11 © NEC Corporation 2017 \Orchestrating a brighter world NEC


Semi-honest AND-gate
$v = a_1 \oplus a_2 \oplus a_3$
$w = b_1 \oplus b_2 \oplus b_3$
9 terms in total
$v \cdot w = (a_1 \oplus a_2 \oplus a_3) \cdot (b_1 \oplus b_2 \oplus b_3)$
$= a_1b_1 \oplus a_1b_2 \oplus a_1b_3 \oplus a_2b_1 \oplus a_2b_2 \oplus a_2b_3 \oplus a_3b_1 \oplus a_3b_2 \oplus a_3b_3$ (*)
P_1 P_2 P_3
Share of v
$a_3 \oplus a_1$, a_1 | $a_1 \oplus a_2$, a_2 | $a_2 \oplus a_3$, a_3
Share of w
$b_3 \oplus b_1$, b_1 | $b_1 \oplus b_2$, b_2 | $b_2 \oplus b_3$, b_3
$r_1 = (a_3 \oplus a_1) \cdot (b_3 \oplus b_1) \oplus (a_1 \cdot b_1) = a_3b_3 \oplus a_3b_1 \oplus a_1b_3$
$r_2 = (a_1 \oplus a_2) \cdot (b_1 \oplus b_2) \oplus (a_2 \cdot b_2) = a_1b_1 \oplus a_1b_2 \oplus a_2b_1$
$r_3 = (a_2 \oplus a_3) \cdot (b_2 \oplus b_3) \oplus (a_3 \cdot b_3) = a_2b_2 \oplus a_2b_3 \oplus a_3b_2$
$r_1 \oplus r_2 \oplus r_3 = v \cdot w$ (3-out-of-3 SS for $v \cdot w$)
12 © NEC Corporation 2017 \Orchestrating a brighter world NEC


Efficiency of Semi-Honest MPC
[CCS16]: about 7 billion AND gate/sec
●1.3 million AES/sec
●10 Gbps network
Cf.
Number of gates
CCS16(Ours)
7,000,000,000
Sharemind®
462,400,000
VIFF framework
10,880,000
First Impl. of Yao
620
Years
2004
12 years(!?)
2016
13
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world
NEC


The malicious protocol of [Furukawa et al., Eurocrypt'17]
An extension of [Araki et al., CCS'16]
XOR gate: NO communication
→ NO information leaked
DONT BE A BOTTLENECK
Pre-processing protocol
Generate N random multiplication triples (next page)
7 bits per AND gate
On-line protocol
1.Share the inputs
2.Run a "cheap" Semi-honest protocol
3.Verify all ANDs gates
4.Reconstruct Output
3 bits per AND gate
14
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world
NEC


Generation of Random Multiplication Triples
A "random multiplication triple" is a triple of shares ([a], [b], [c]) such that: [a] and [b] are random sharings and c = a · b
[a], [b] are generated without any interaction!
[c] is computed using a semi-honest multiplication protocol
1 bit of communication!
15
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world
NEC

Generation of Random Multiplication Triples

A "random multiplication triple" is a triple of shares $([a],[b],[c])$ such that: $[a]$ and $[b]$ are random sharings and $c = a \cdot b$

- $[a],[b]$ are generated without any interaction!
- $[c]$ is com[puted using a semi-honest] multiplication protocol

Input/output tuple of AND gate can be seen as "triple"

1 bit of communication!

Verifying Gate(/Triple) by Triple

$[a]$, $[b]$, $[c]$ triples ⟷ $[x]$, $[y]$, $[z]$ AND-gate

$[z]$

$[x]$ $[y]$

accept iff both triples are correct or both are incorrect

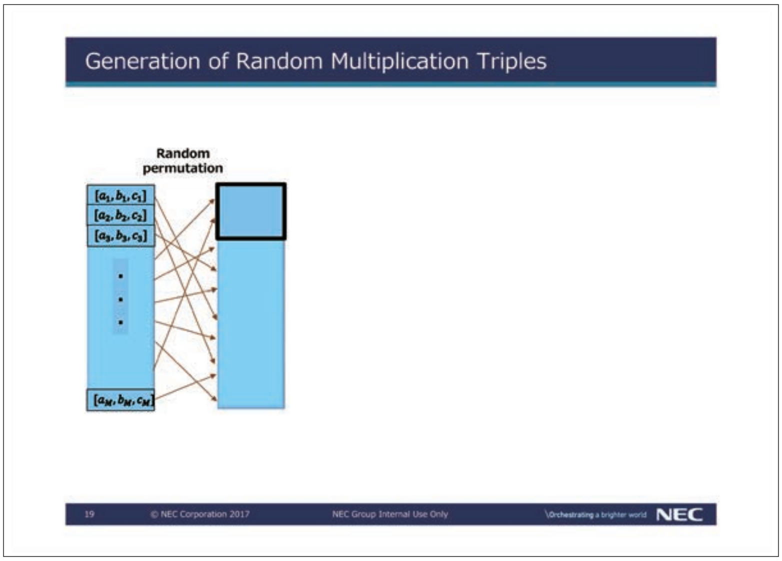
Generation of Random Multiplication Triples

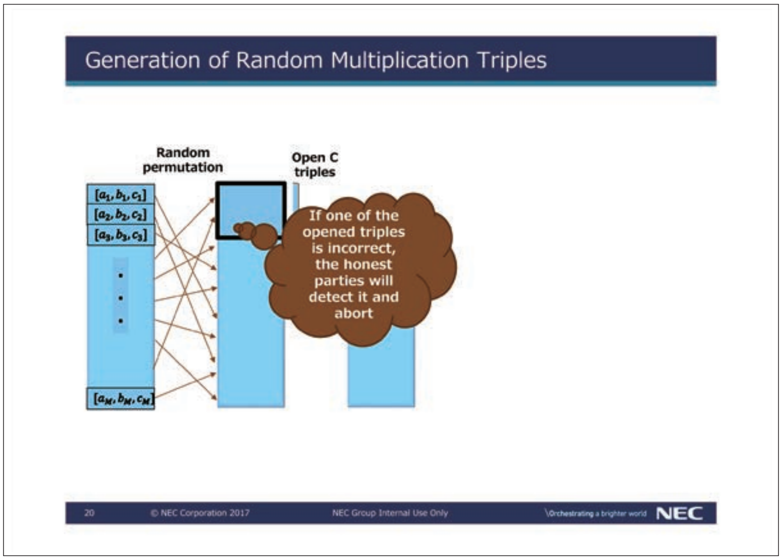
A "random multiplication triple" is a triple of shares $([a],[b],[c])$ such that: $[a]$ and $[b]$ are random sharings and $c = a \cdot b$

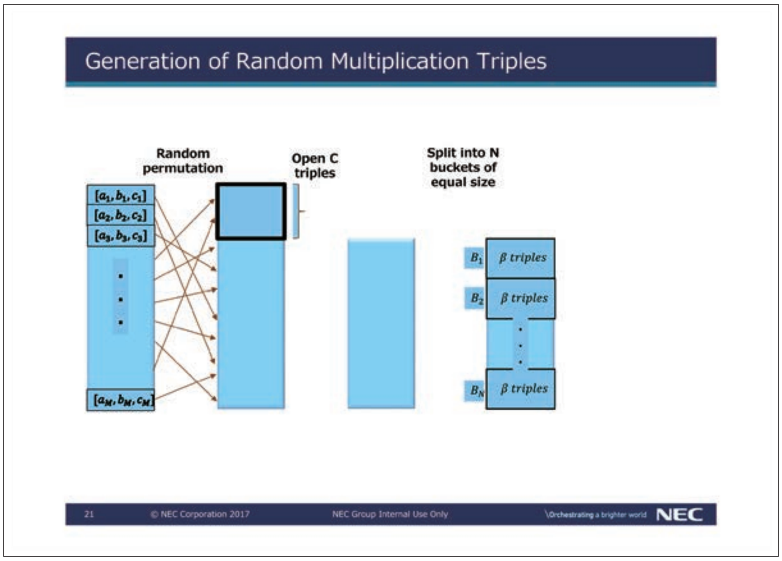
- $[a],[b]$ are generated without any interaction!
- $[c]$ is computed using a semi-honest multiplication protocol

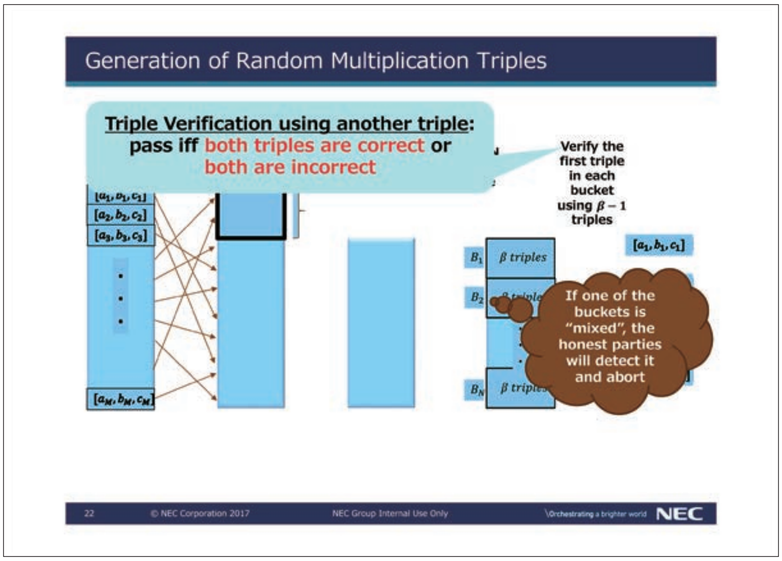
1 bit of communication!

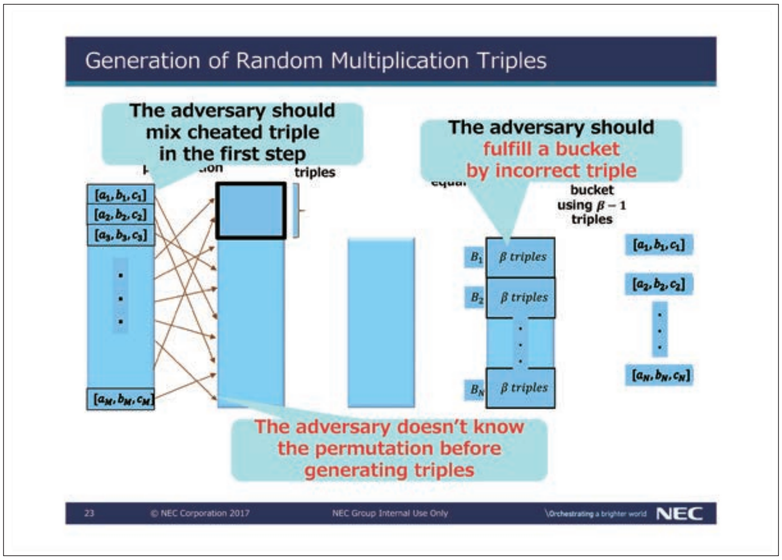
How can we verify that the triples generated are valid?

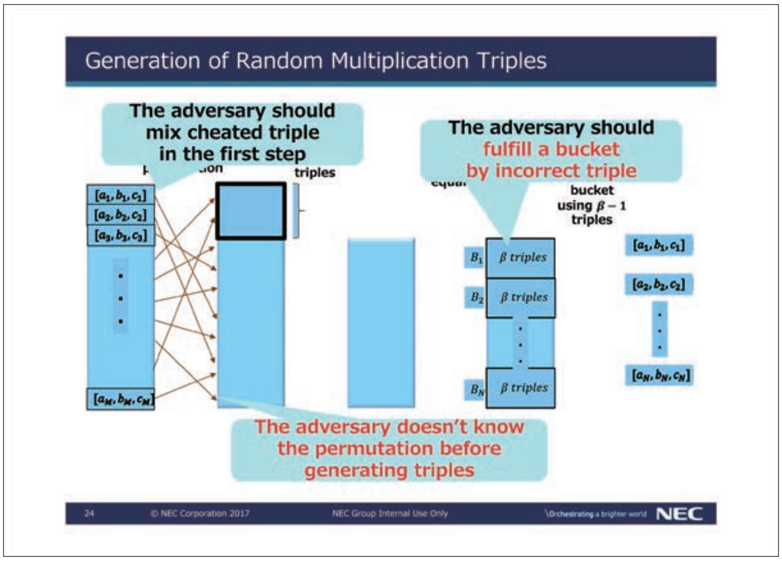

Generation of Random Multiplication Triples
Random permutation
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
19 © NEC Corporation 2017 NEC Group Internal Use Only \Orchestrating a brighter world NEC


Generation of Random Multiplication Triples
Random permutation
Open C triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
If one of the opened triples is incorrect, the honest parties will detect it and abort
20 © NEC Corporation 2017 NEC Group Internal Use Only \Orchestrating a brighter world NEC


Generation of Random Multiplication Triples
Random permutation
Open C triples
Split into N buckets of equal size
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
B_1 β triples
B_2 β triples
B_N β triples
21 © NEC Corporation 2017 NEC Group Internal Use Only \Orchestrating a brighter world NEC


Generation of Random Multiplication Triples
Triple Verification using another triple:
pass iff both triples are correct or both are incorrect
Verify the first triple in each bucket using $\beta - 1$ triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
B_1 β triples
B_2
B_N
$[a_1, b_1, c_1]$
If one of the buckets is "mixed", the honest parties will detect it and abort
22
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world NEC


Generation of Random Multiplication Triples
The adversary should mix cheated triple in the first step
triples
The adversary should fulfill a bucket by incorrect triple
bucket
using $\beta - 1$
triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
B_1 β triples
B_2 β triples
B_N β triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_N, b_N, c_N]$
The adversary doesn't know the permutation before generating triples
23
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world NEC


Generation of Random Multiplication Triples
The adversary should mix cheated triple in the first step
triples
The adversary should fulfill a bucket by incorrect triple
bucket
using $\beta - 1$
triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_3, b_3, c_3]$
$[a_M, b_M, c_M]$
B_1 β triples
B_2 β triples
B_N β triples
$[a_1, b_1, c_1]$
$[a_2, b_2, c_2]$
$[a_N, b_N, c_N]$
The adversary doesn't know the permutation before generating triples
24
© NEC Corporation 2017
NEC Group Internal Use Only
\Orchestrating a brighter world NEC

β: the bucket size
C: the number of opened triples

Increase β and C
Lower cheating probability

THE TRADE OFF
YOU CAN'T HAVE IT BOTH WAYS

Reduce β and C
Better efficiency

Achieving 1-Billion AND Gates per Second

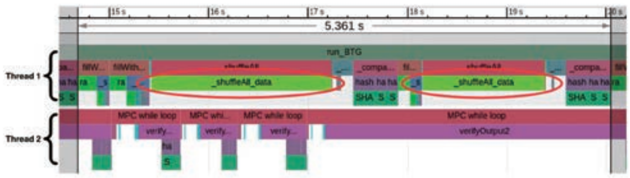
503,766,615 → gate/sec

The baseline protocol
- 10 bits per AND gate
- $\beta = 3$
- $C = 3$

1-billion AND gates per second

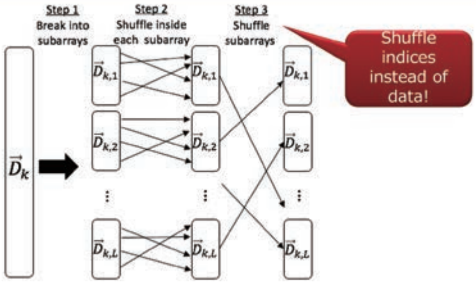
* Statistical error = 2^{-40}
**Cluster of three mid-level servers (2.3GHz CPUs with 20 cores), with a 10Gbps network

Benchmarking



Shuffling of the array is expensive due to cache misses
The dilemma: a large array is needed for the combinatorics, but results in slowdown

Optimization 1: cache-efficient shuffling



This is not a random permutation!
But: We prove that the cheating probability is almost the same

Achieving 1-Billion AND Gates per Second

503,766,615 765,448,459 gate/sec

The baseline protocol
- 10 bits per AND gate
- $\beta = 3$
- $C = 3$

Using a "cache-efficient" shuffle
- 10 bits per AND gate
- $\beta = 3$
- $C = 2^{10}$

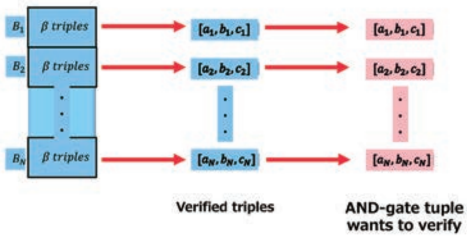
1-billion AND gates per second

* Statistical error = 2^{-40}
**Cluster of three mid-level servers (2.3GHz CPUs with 20 cores), with a 10Gbps network

Optimization 2: Reduce the size of the bucket

Intuition

- "Verifying triple by triple" and "Verifying the gate by triple" is essentially same procedure.
- Can we use the online multiplication triple generated in each AND gate computation, as <u>one of the triples in the bucket</u>?

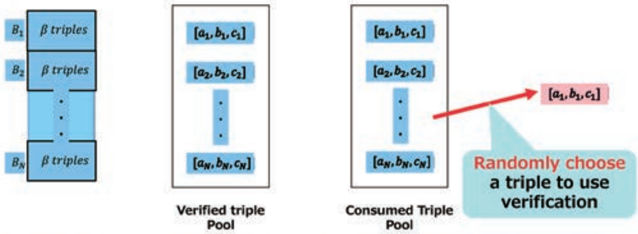


Generation of Random Multiplication Triples

Intuition

- "Verifying triple by triple" and "Verifying the circuit by triple" is essentially same procedure.
- Can we use the online multiplication triple generated in each AND gate computation, as one of the triples in the bucket? → **Yes**

Modification: On-demand shuffling

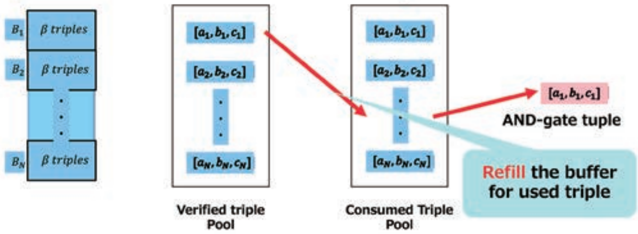


Generation of Random Multiplication Triples

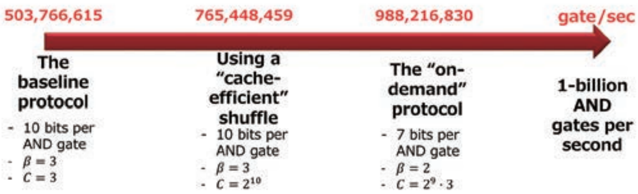
Intuition

- "Verifying triple by triple" and "Verifying the circuit by triple" is essentially same procedure.
- Can we use the online multiplication triple generated in each AND gate computation, as one of the triples in the bucket? → **Yes**

Modification: On-demand shuffling

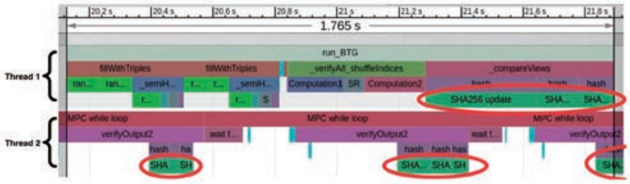


Achieving 1-Billion AND Gates per Second



* Statistical error = 2^{-40}
**Cluster of three mid-level servers (2.3GHz CPUs with 20 cores), with a 10Gbps network

Securely Comparing Views



- SHA-256 is used to compare the views of the parties in the verification stage (instead of sending openings)
- Can replace it with GMAC – implemented using PCLMULQDQ

Achieving 1-Billion AND Gates per Second

gate/sec

503,766,615	765,448,459	988,216,830	1,152,751,967
The baseline protocol - 10 bits per AND gate - $\beta = 3$ - $C = 3$	**Using a "cache-efficient" shuffle** - 10 bits per AND gate - $\beta = 3$ - $C = 2^{10}$	**The "on-demand" protocol** - 7 bits per AND gate - $\beta = 2$ - $C = 2^9 \cdot 3$	**1-billion AND gates per second**

* Statistical error = 2^{-40}
**Cluster of three mid-level servers (2.3GHz CPUs with 20 cores), with a 10Gbps network

Summary

- It is possible to achieve very fast rates even for malicious adversaries
  - This holds for 3-party with honest majority (e.g., service model, auxiliary server, internal protection)
- We achieve rates of
  - *Semi-honest*:
    - **1 bit comm/AND gates**
    - over **7 billion AND gates/second** (over 1.3 million AES operations per second)
  - *Malicious*:
    - **7 bit comm./AND gates** (Statistical error = 2^{-40})
    - over **1 billion AND gates/second** (about 215,000 AES operations per second)
- Can significantly extend the applications that MPC can utilize
- For more detail, please see our paper at IEEE S&P2017.

Orchestrating a brighter world

NEC

Key Components in MEVAL

Ryo KIKUCHI
(Joint work with Dai IKARASHI, Koki HAMADA, Koji CHIDA, Naoto KIRIBUCHI, Gembu MOROHASHI)

NTT Corporation
kikuchi.ryo@lab.ntt.co.jp

We have developed a novel system MEVAL: Multiparty EVALuator, which performs secret-sharing-based secure computation with an honest majority. In the system, a user can choose either two security levels: passive (a.k.a. semi-honest) or active (a.k.a. malicious) security with abort. One of features of MEVAL is efficiency. As an example, we experimented with secure AES computation and MEVAL achieved 517 Mbps (involving 4 million AES per second) in passive security, and 131 Mbps (involving 1 million AES per second) in active security with abort. These are faster than 169 Mbps [2] in passive security and 27 Mbps [1] in active security with abort.

For practical use of secure computation, not only basic functions, such as multiplication, are *not* enough and high-level functions, such as comparison and sort, are required [4]. We have developed MEVAL for practical use and it therefore supports many high-level functions.

In this talk, we introduce three key components of high-level functions in MEVAL: bit decomposition, sort, and join. These components use novel techniques and improve efficiency drastically. Table 1 shows an experimental result of the components in three-party setting with a gigabit network.

	function	passive security	active security with abort
[4]	bit decomposition (10^7 elements)	200 sec	-
MEVAL		0.90 sec	14.81 sec
[3]	sort (10^5 elements)	150 sec	-
MEVAL		0.54 sec	1.43 sec
[5]	join (10^3 records)	25 sec	-
MEVAL		0.02 sec	0.06 sec

Table 1. Efficiency comparison in a gigabit network

References

[1] T. Araki, A. Barak, J. Furukawa, Y. Lindell, A. Nof, K. Ohara, A. Watzman, and O. Weinstein. Optimized honest-majority MPC for malicious adversaries - breaking the 1 billion-gate per second barrier. S&P 2017.

[2] T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. High-throughput semi-honest secure three-party computation with an honest majority. ACM CCS 2016.

[3] D. Bogdanov, S. Laur, and R. Talviste. A practical analysis of oblivious sorting algorithms for secure multi-party computation. NordSec 2014.

[4] D. Bogdanov, M. Niitsoo, T. Toft, and J. Willemson. High-performance secure multi-party computation for data mining applications. Int. J. Inf. Sec., 2012.

[5] S. Laur, R. Talviste, and J. Willemson. From oblivious AES to efficient and secure database join in the multiparty setting. ACNS 2013.

Key components in MEVAL

Ryo Kikuchi @ NTT Corporation

Protocols by Dai Ikarashi, Koki Hamada, and Ryo Kikuchi

Icons are designed by Freepik from Flaticon

Today's talk

- MEVAL: Multiparty EVALuator
  - Web page coming soon
- Key components
  - Bit-decomposition
  - Oblivious sort
  - Oblivious join

2

Multiparty computation

3

Two security models

passive security

An adversary follows the protocol

active security with abort

An adversary can do arbitrary behavior

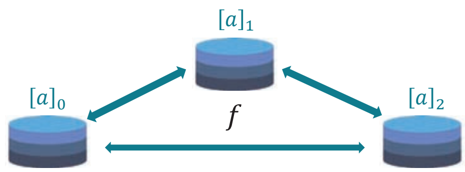
Honest parties output ⊥ if an adversary cheats

4

MEVAL: Multiparty EVALuator

MEVAL is an MPC system we have developed

secret sharing + three-party + honest majority



User can choose either passive or active w/ abort

5

Feature of MEVAL

- Fulfilling functions
  - logical/arithmetic circuit
  - high-level operations: join, sort, map, comparison, etc.
  - various SS conversion and field conversion
- Efficiency
  - original (optimized) protocols
  - implemented by expert programmer **Dai Ikarashi**

openstack

ex.) His erasure code library for OpenStack Swift storage is 10% faster than Intel's and 2 times faster than Swift-embedded "j-erasure" (USENIX FAST'13, a storage top conference)!

http://www.ntt.co.jp/news2015/1505e/150518a.html

6

Efficiency of MEVAL (basic function)

Benchmark by AES computation
(consists of local comp. and mult. prot.)

	Measure	Passive	Active with abort
[AFL+16]	AES/sec	1,324,117	-
[ABF+17]		-	212,000
MEVAL		**4,041,655**	**1,025,303**

	Measure	Passive	Active with abort
[AFL+16]	bit/sec	7,150,231,800	-
[ABF+17]		-	1,152,751,967
MEVAL		16,554,617,600	4,199,641,600

[AFL+16] T. Araki, J. Furukawa, Y. Lindell, A. Nof, K. Ohara. High-throughput semi-honest secure three-party computation with an honest majority. ACM CCS 2016.
[ABF+17] T. Araki, A. Barak, J. Furukawa, T. Lichter, Y. Lindell, A. Nof, K. Ohara, A. Watzman, O. Weinstein.: Optimized Honest-Majority MPC for Malicious Adversaries - Breaking the 1 Billion-Gate Per Second Barrier, S&P 2017.

Efficiency of MEVAL (high-level function)

- Basic function is not enough for practical use
- Many applications require high-level operations
- MEVAL have developed high-level functions

Key components

- Bit-decomposition
  Convert a into $(a^{(0)}, \ldots, a^{(\ell-1)})$, where $a = a^{(\ell-1)} a^{(\ell-2)} \cdots a^{(0)}$
- Oblivious sort
  Sort $(2,4,1,3)$ into $(1,2,3,4)$ and output the permutation π,
  where $\pi(1) = 3, \pi(2) = 1, \pi(3) = 4, \pi(4) = 2$
- Oblivious join
  - Input:

Key	height	weight
3	200	99
9	160	85

Key	purchase
3	water
7	egg

  - Output:

Key	height	weight	purchase
3	200	99	water

Bit-decomposition

Bit decomposition: $\mathbb{F}_p$ to $\mathbb{F}_2$

Motivation: computation in better suited field

	Secret-shared in $\mathbb{F}_p$	Secret-shared in $\mathbb{F}_2$
Sum	☺ Local computation (computing addition)	☹ Communication required (computing adding circuit)
Comparison	☹ Difficult to compute (except [NO07])	☺ Easy to compute (If $a < 2^\ell$, $[a > b]$ is ℓ-th bit of $2^\ell + [a] - [b]$)

Known protocols cost $O(|p|^2)$ bit communication [DFK+06, NO07]

regarding # of parties as constant

MEVAL uses an original bit-decomposition protocol

$O(\ell)$ bit communication

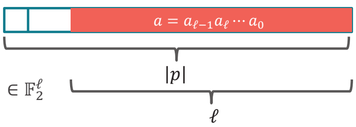
[NO07] T. Nishide and K. Ohta: Multiparty Computation for Interval, Equality, and Comparison without Bit-Decomposition Protocol, PKC 2007
[DFK+06] I. Damgard, M. Fitzi, E. Kiltz, J.B. Nielsen, and T. Toft.: Unconditionally secure constant-rounds multi-party computation for equality, comparison, bits and exponentiation, TCC 2006.

Setting and notation

- Consider $\mathbb{F}_p$ and $\mathbb{F}_2$
  - $p = 2^m - 1$, i.e., a Merrsenne prime
  - $|p| := \lceil \log p \rceil$
- a is $0 < a < 2^\ell$, $\ell < |p|$

  $a \in \mathbb{F}_p$ is represented as $\left(a^{(0)}, \ldots, a^{(\ell-1)}\right) \in \mathbb{F}_2^\ell$
- (2,3)-linear secret sharing
  - $[\![a]\!]$: sharing of a in $\mathbb{F}_p$, $[\![a]\!]$: share of
  - $a = \lambda_{i_0}[\![a]\!]_{i_0} + \lambda_{i_1}[\![a]\!]_{i_1}$ for any i_0, i_1
- Passive security

  Active security can be obtained by using known techniques [IKHC14,ikhc13]



[IKHC14] D. Ikarashi, R. Kikuchi, K. Hamada, K. Chida: Actively Private and Correct MPC Scheme in t < n/2 from Passively Secure Schemes with Small Overhead, ePrint 2014.
[ikhc13] D. Ikarashi, R. Kikuchi, K. Hamada, K. Chida: An Efficient SIMD Protocol against Malicious Adversaries for Secure Computation Schemes Based on (k,n) Secret Sharing Schemes with Small Party Sets, CSS 2013 (in Japanese)

Basic technique of bit-decomposition

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = \left([\![a^{(0)}]\!], \dots, [\![a^{(|p|-1)}]\!]\right)$, where $a^{(i)} \in \mathbb{F}_2$

1. Generate randomness: $\langle r \rangle = \left([\![r^{(0)}]\!], \dots, [\![r^{(|p|-1)}]\!]\right)$, where $r^{(i)} \in \mathbb{F}_2$
2. $[\![r]\!] \coloneqq \sum_{i<|p|} 2^i [\![r^{(i)}]\!]$
3. Reveal $[\![a]\!] - [\![r]\!]$ and obtain $c = a - r$
4. $\langle a' \rangle \coloneqq c + \langle r \rangle$ (adding circuit)
5. $\langle q \rangle \coloneqq \mathrm{Compare}(a' \geq_? p)$ (comparison circuit)
6. $\langle a \rangle \coloneqq \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

13

Where is the bottleneck?

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = \left([\![a^{(0)}]\!], \dots, [\![a^{(|p|-1)}]\!]\right)$, where $a^{(i)} \in \mathbb{F}_2$

1. The output size is $|p|^2$ bits

1. Generate randomness: $\langle r \rangle = \left([\![r^{(0)}]\!], \dots, [\![r^{(|p|-1)}]\!]\right)$, where $r^{(i)} \in \mathbb{F}_2$
2. $[\![r]\!] \coloneqq \sum_{i<|p|} 2^i [\![r^{(i)}]\!]$
3. Reveal $[\![a]\!] - [\![r]\!]$ and obtain $c = a - r$
4. $\langle a' \rangle \coloneqq c + \langle r \rangle$ (adding circuit)
5. $\langle q \rangle \coloneqq \mathrm{Compare}(a' \geq_? p)$ (comparison circuit)
6. $\langle a \rangle \coloneqq \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

2. Randomness is $|p|^2$ bits

3. Circuit size is $O(|p|)$

14

Optimization 1: modifying output

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = \left([\![a^{(0)}]\!], \dots, [\![a^{(|p|-1)}]\!]\right)$, where $a^{(i)} \in \mathbb{F}_2$

1. The output size is $|p|^2$ bits

ℓ elements in $\mathbb{F}_2$ are sufficient
($[\cdot]$ denotes share in $\mathbb{F}_2$)
1bit $\times$ $\ell = \ell$ bit

1. Generate ... $\left([\![r^{(0)}]\!], \dots, [\![r^{(|p|-1)}]\!]\right)$, where $r^{(i)} \in \mathbb{F}_2$
2. $[\![r]\!] \coloneqq \sum$ ...
3. Reveal $[\![a]\!] - [\![r]\!]$ and obtain $c = a - r$
4. $\langle a' \rangle \coloneqq c + \langle r \rangle$ (adding circuit)
5. $\langle q \rangle \coloneqq \mathrm{Compare}(a' \geq_? p)$ (comparison circuit)
6. $\langle a \rangle \coloneqq \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

2. Randomness is $|p|^2$ bits

3. Circuit size is $O(|p|)$

15

Optimization 1: modifying output

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = \boxed{([a^{(0)}], \ldots, [a^{(\ell-1)}])}$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. Generate randomness: $\langle r \rangle = ([\![r^{(0)}]\!], \ldots, [\![r^{(|p|-1)}]\!])$, where $r^{(i)} \in \mathbb{F}_2$
2. $[\![r]\!] \coloneqq \sum_{i<|p|} 2^i [\![r^{(i)}]\!]$
3. Reveal $[\![a]\!] - [\![r]\!]$ and obtain $c = a - r$
4. $\langle a' \rangle \coloneqq c + \langle r \rangle$ (adding circuit)
5. $\langle q \rangle \coloneqq \mathrm{Compare}(a' \geq_? p)$ (comparison circuit)
6. $\langle a \rangle \coloneqq \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

2. Randomness is $|p|^2$ bits

3. Circuit size is $O(|p|)$

16

Optimization 2: Generate (2, 2)-sharing

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = ([a^{(0)}], \ldots, [a^{(\ell-1)}])$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. Generate randomness: $\langle r \rangle = ([\![r^{(0)}]\!], \ldots, [\![r^{(|p|-1)}]\!])$, where $r^{(i)} \in \mathbb{F}_2$
2. $[\![r]\!] \coloneqq \sum_{i<|p|} 2^i [\![r^{(i)}]\!]$
3. Reveal $[\![a]\!] - [\![r]\!]$ and obtain $c = a - r$
4. $\langle a' \rangle \coloneqq c + \langle r \rangle$ (adding circuit)
5. $\langle q \rangle$ ... n circuit)
6. $\langle a \rangle$

Generating (2, 2) sharing of a
i.e., $a = c + r$

2. Randomness is $|p|^2$ bits

3. Circuit size is $O(|p|)$

17

Optimization 2: Generate (2, 2)-sharing

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = ([a^{(0)}], \ldots, [a^{(\ell-1)}])$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. 2 parties
   1. compute $\lambda_i [\![a]\!]_i$,
   2. secret-share each bit of $\lambda_i [\![a]\!]_i$ in $\mathbb{F}_2$
2. Let the above shares be $\langle a_0 \rangle$ and $\langle a_1 \rangle$
3. $\langle a' \rangle \coloneqq \langle a_0 \rangle + \langle a_1 \rangle$ (adding circuit)
4. $\langle q \rangle \coloneqq \mathrm{Compare}(a' \geq_? p)$ (comparison circuit)
5. $\langle a \rangle \coloneqq \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

$O(\ell)$ bits

3. Circuit size is $O(|p|)$

18

Optimization 3: Excluding comparison circuit

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = ([a^{(0)}], \dots, [a^{(\ell-1)}])$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. 2 parties
   1. compute $\lambda_i [\![a]\!]_i$,
   2. secret-share each bit of $\lambda_i [\![a]\!]_i$ in $\mathbb{F}_2$
2. Let the above shares be $\langle a_0 \rangle$ and $\langle a_1 \rangle$
3. $\langle a' \rangle := \langle a_0 \rangle + \langle a_1 \rangle$ (adding circuit)
4. $\langle q \rangle := \boxed{\mathrm{Compare}(a' \geq_? p)}$ (comparison circuit)
5. $\langle a \rangle := \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

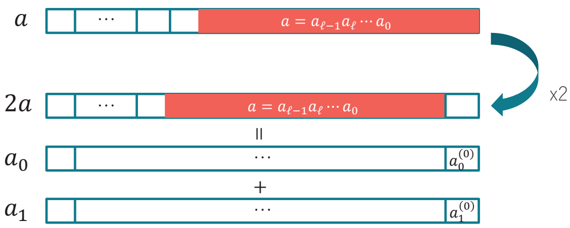
$O(\ell)$ bits

3. Circuit size is $O(|p|)$

19

Quotient appears at LSB

Assume



If $2a$ is shared as $2a = a_0 + a_1$, $\mathrm{Compare}(a \geq_? p) = a_0^{(0)} \oplus a_1^{(0)}$

Proof is appeared in [ihkc13]

[ihkc13] D. Ikarashi, K. Hamada, R. Kikuchi, and K. Chida: $O(\ell)$ Bits Communication Bit Decomposition and $O(|p'|)$ Bits Communication Modulus Conversion for Small k Secret-sharing-based Secure Computation, CSS 2013 (in Japanese)

20

Optimization 3: Excluding comparison circuit

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = ([a^{(0)}], \dots, [a^{(\ell-1)}])$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. 2 parties
   1. compute $\lambda_i [\![a]\!]_i$,
   2. secret-share each bit of $\lambda_i [\![a]\!]_i$ in $\mathbb{F}_2$
2. Let the above shares be $\langle a_0 \rangle$ and $\langle a_1 \rangle$
3. $\langle a' \rangle := \langle a_0 \rangle + \langle a_1 \rangle$ (adding circuit)
4. $\langle q \rangle := \boxed{\mathrm{Compare}(a' \geq_? p)}$ (comparison circuit)
5. $\langle a \rangle := \langle a' \rangle - p\langle q \rangle$ (subtracting circuit)

$O(\ell)$ bits

3. Circuit size is $O(|p|)$

21

Optimization 3: Excluding comparison circuit

- Input: $[\![a]\!] \in \mathbb{F}_p$
- Output: $\langle a \rangle = ([a^{(0)}], \ldots, [a^{(\ell-1)}])$, where $a^{(i)} \in \mathbb{F}_2$

$O(\ell)$ bits

1. 2 parties
   1. compute $\boxed{2\lambda_i [\![a]\!]_i}$
   2. secret-share each bit of $\boxed{2\lambda_i [\![a]\!]_i}$ in $\mathbb{F}_2$
2. Let the above shares be $\langle a_0 \rangle$ and $\langle a_1 \rangle$
3. $\langle a' \rangle := \langle a_0 \rangle + \langle a_1 \rangle$ (adding circuit)
4. $[q] := \boxed{[a_0^{(0)}] \oplus [a_1^{(0)}]}$
5. $\langle a \rangle := \langle a' \rangle - p[q]$ (subtracting circuit)
6. Shift $\langle a \rangle$ a single bit: $\boxed{[a^{(i-1)}] := [a^{(i)}]}$

$O(\ell)$ bits

Circuit size is $O(\ell)$

22

Result (bit-decomposition)

- Bit-decomposition protocol with $O(\ell)$ bit communication
  Existing protocols cost $O(|p|^2)$
- Experimental result on 10^7 records, 1G LAN, $p = 2^{61} - 1$, $\ell = 20$

	Passive	Active w/ abort
[BNTW12]	200 sec	-
MEVAL	0.90 sec	14.81 sec

[BNTW12] D. Bogdanov, M. Niitsoo, T. Toft, J. Willemson.: High-performance secure multi-party computation for data mining applications. Int. J. Inf. Sec. 2012.

23

Oblivious sort

24

What is oblivious sort?

- Input: $[\![2]\!], [\![4]\!], [\![1]\!], [\![3]\!]$
- Output: $[\![1]\!], [\![2]\!], [\![3]\!], [\![4]\!], [\![\pi]\!]$,
  where $\pi(1) = 3, \pi(2) = 1, \pi(3) = 4, \pi(4) = 2$

An important component for

- computing median and percentile,
- other high-level functions, such as join

But, difficult to explain ☹ so we skip the detail

25

Experimental result (oblivious sort)

Experiment on 10^5 records, 1G LAN, $p = 2^{61} - 1$, $\ell = 20$

	Passive	Active w/ abort
[BLT14]	150 sec	-
MEVAL	0.54 sec	1.43 sec

[BLT14] D. Bogdanov, S. Laur, and R. Talviste.: A Practical Analysis of Oblivious Sorting Algorithms for Secure Multi-party Computation. NordSec 2014.

26

Oblivious join

27

Oblivious join

Joining secret-shared two tables

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water

Key	height	weight	purchase
3	200	99	water
9	160	85	medicine
9	160	85	water

Application

Cross analysis of different companies

Key	height	weight

Attribute data of company A

Key	purchase

History data of company B

Key	height	weight	purchase

Cross table of company A and B

Setting

Key of history data may duplicate

No duplication

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water

Duplication

Key	height	weight	purchase
3	200	99	water
0	0	0	0
9	160	85	medicine
9	160	85	water

0 if no corresponding record in attribute data

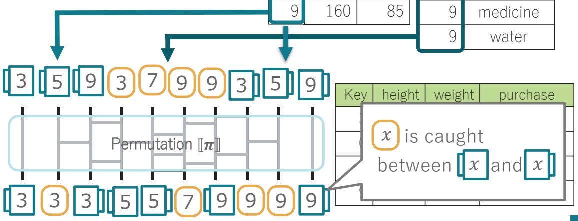
This talk omits how to eliminate 0

Computing “weight” column 1/4

Use oblivious sort and store $[\![\pi]\!]$

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water



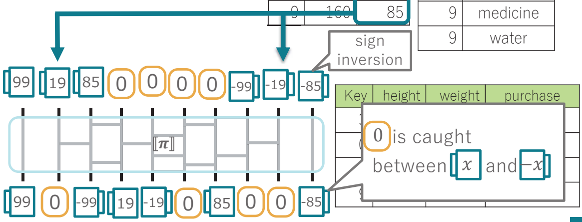
31

Computing “weight” column 2/4

Apply π with “weight” and a tuple of 0

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water



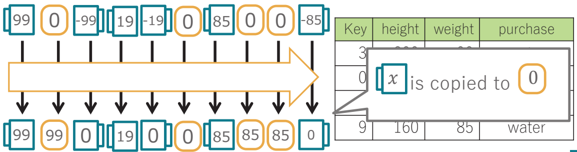
32

Computing “weight” column 3/4

Compute prefix-sum

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water



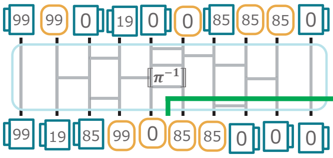
33

Computing “weight” column 4/4

Compute π^{-1}

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water



Key	height	weight	purchase
3	200	99	water
0	0	0	0
9	160	85	medicine
9	160	85	water

34

Computing “height” column

Apply the same thing

Key	height	weight
3	200	99
5	110	19
9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water

Key	height	weight	purchase
3	200	99	water
0	0	0	0
9	160	85	medicine
9	160	85	water

35

Computing “purchase” column

Apply the same thing with a tuple of 1

e	Key	height	weight
1	3	200	99
1	5	110	19
1	9	160	85

Key	purchase
3	water
7	egg
9	medicine
9	water

Key	purchase
3	water
7	egg
9	medicine
9	water

×

e
1
0
1
1

Key	height	weight	purchase
3	200	99	water
0	0	0	0
9	160	85	medicine
9	160	85	water

36

Experimental result (oblivious join)

Experiment on 10^3 records, 1G LAN, $p = 2^{61} - 1$, $\ell = 20$

	Passive	Active
[LTW13]	30 s	-
MEVAL	0.02 s	0.35 s

Experiment on 10^6 records,

	Passive	Active
MEVAL	15.13 s	44.04 s

[LTW13] S. Laur, R. Talviste, J. Willemson.: From oblivious AES to efficient and secure database join in the multiparty setting. ACNS 2013.

37

Summary

- We have developed MEVAL

  secret sharing + three-party + honest majority

  User can choose either security level: passive or active w/ abort

- Support high-level functions

  Three key components: bit-decomposition, oblivious sort, oblivious join

	Function	Passive	Active w/ abort
[BNTW12]	Bit-decomposition (10^7 records)	200 s	-
MEVAL		0.90 s	14.81 s
[BLT14]	Oblivious sort (10^5 records)	150 s	-
MEVAL		0.54 s	1.43 s
[LTW13]	Oblivious join (10^3 records)	25 s	-
MEVAL		0.02 s	0.06s

38

Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol

Bernardo DAVID (Joint work with Aggelos Kiayias, Alexander Russell and Roman Oliynykov)

Tokyo Insitute of Technology

bernardo@bmdavid.com

We present Ouroboros, the first blockchain protocol based on proof of stake with rigorous security guarantees. We establish security properties for the protocol comparable to those achieved by the bitcoin blockchain protocol. As the protocol provides a proof of stake blockchain discipline, it offers qualitative efficiency advantages over blockchains based on proof of physical resources (e.g., proof of work). We showcase the practicality of our protocol in real world settings by providing experimental results on transaction processing time obtained with a prototype implementation in the Amazon cloud. We also present a novel reward mechanism for incentivizing the protocol and we prove that given this mechanism, honest behavior is an approximate Nash equilibrium, thus neutralizing attacks such as selfish mining and block withholding.

Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol

Bernardo David

Tokyo Institute of Technology

Joint work with *Aggelos Kiayias, Alexander Russel, Roman Oliynykov*

Outline

1. **History: e-cash**

I ♥ 80s

2. **Bitcoin and Blockchains**
3. **Ouroboros**

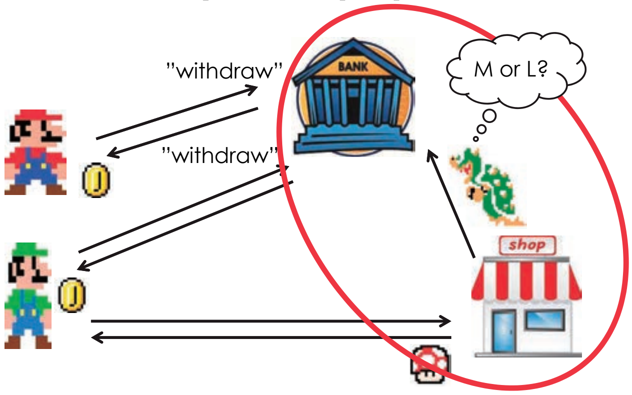
4. **Ouroboros Praos**

The 1980s
David Chaum and anonymous e-cash

"The difference between
a bad electronic cash system
and well-developed digital cash
will determine whether
we will have a dictatorship
or a real democracy"

(attributed to Chaum)

Anonymous payments



Chaum's anonymous e-cash

- **Just like fiat currency:**
  - ***Anonymous***
  - ***Secure*** *(no double spending or faking)*
  - ***Only banks issue money***

- **But…**
  - ***Centralized*** *and* ***bankrupted*** *in 1999*

Outline

1. History: e-cash
2. **Bitcoin and Blockchains**
3. **Ouroboros**
4. **Ouroboros Praos**

A New Era: Bitcoin and Blockchains

A New Era: Bitcoin and Blockchains

- 2009: **Bitcoin announced** by Satoshi Nakamoto
  - *Pseudonym for person or group of people*
- 2009-2011: slow start…
- 2011-2013: Silk Road and Dread Pirate Roberts
- End 2013: **Bitcoin price skyrockets**
  - *and the world notices!*
- Mid-2015: Ethereum and complex Smart Contracts

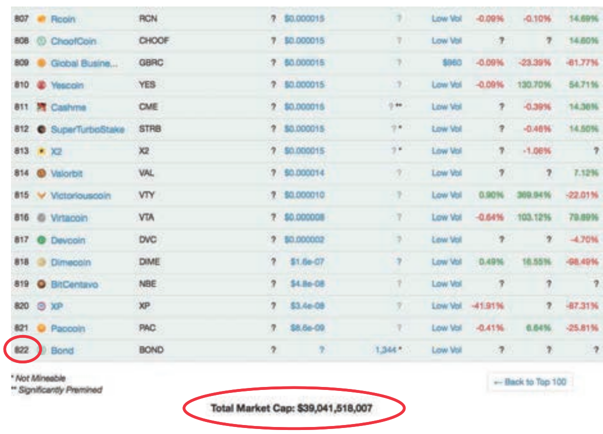
All Currencies

1 BTC = US$1465

All · Currencies · Assets · USD · ← Back to Top 100

#	Name	Symbol	Market Cap	Price	Circulating Supply	Volume (24h)	% 1h	% 24h	% 7d
1	Bitcoin	BTC	$23,901,601,426	$1465.64	16,307,962	$412,628,000	-0.09%	-0.10%	14.70%
2	Ethereum	ETH	$7,108,768,898	$77.88	91,273,569	$144,380,000	0.10%	2.27%	49.96%
3	Ripple	XRP	$2,097,198,310	$0.055357	37,884,902,021 *	$25,107,700	0.75%	5.34%	69.55%
4	Litecoin	LTC	$902,730,748	$17.73	50,922,907	$72,717,000	0.91%	14.07%	14.84%
5	Dash	DASH	$638,909,974	$87.91	7,267,732	$15,765,300	0.35%	3.43%	23.08%
6	Ethereum Classic	ETC	$612,481,006	$6.71	91,266,869	$38,351,400	-0.41%	4.81%	60.85%
7	NEM	XEM	$479,515,500	$0.053280	8,999,999,999 *	$3,550,680	-1.02%	10.33%	22.00%
8	Monero	XMR	$338,821,362	$23.53	14,398,079	$7,976,290	0.17%	4.54%	21.32%
9	Golem	GNT	$188,218,700	$0.229535	820,000,000 *	$11,184,300	0.61%	17.41%	120.76%
10	Augur	REP	$182,014,800	$16.55	11,000,000 *	$3,853,550	0.91%	3.72%	23.65%
11	MaidSafeCoin	MAID	$114,979,086	$0.254068	452,552,412 *	$1,658,020	1.49%	3.56%	7.92%
12	Zcash	ZEC	$113,200,429	$92.76	1,220,344	$8,176,020	-0.27%	3.13%	28.88%
13	Stratis	STRAT	$100,996,406	$1.03	98,369,929 *	$3,584,860	11.95%	39.39%	63.61%
14	PIVX	PIVX	$87,924,895	$1.65	53,228,457 *	$653,889	1.72%	0.99%	29.09%
15	Gnosis	GNO	$86,284,496	$78.11	1,104,590 *	$10,746,800	-4.66%	-10.68%	?
16	Dogecoin	DOGE	$72,133,415	$0.000660	109,217,097,473	$4,602,240	-0.26%	10.36%	21.09%

coinmarketcap.com



807	Rcoin	RCN	?	$0.000015	?	Low Vol	-0.09%	-0.10%	14.69%
808	ChoofCoin	CHOOF	?	$0.000015	?	Low Vol	?	?	14.60%
809	Global Busine...	GBRC	?	$0.000015	?	$960	-0.09%	-23.39%	-61.77%
810	Yescoin	YES	?	$0.000015	?	Low Vol	-0.09%	130.70%	54.71%
811	Cashme	CME	?	$0.000016	? **	Low Vol	?	-0.39%	14.36%
812	SuperTurboStake	STRB	?	$0.000015	? *	Low Vol	?	-0.46%	14.50%
813	X2	X2	?	$0.000015	? *	Low Vol	?	-1.06%	?
814	Valorbit	VAL	?	$0.000014	?	Low Vol	?	?	7.12%
815	Victoriouscoin	VTY	?	$0.000010	?	Low Vol	0.90%	369.94%	-22.01%
816	Virtacoin	VTA	?	$0.000008	?	Low Vol	-0.64%	103.12%	79.89%
817	Devcoin	DVC	?	$0.000002	?	Low Vol	?	?	-4.70%
818	Dimecoin	DIME	?	$1.6e-07	?	Low Vol	0.49%	16.55%	-98.49%
819	BitCentavo	NBE	?	$4.8e-08	?	Low Vol	?	?	?
820	XP	XP	?	$3.4e-08	?	Low Vol	-41.91%	?	-87.31%
821	Paccoin	PAC	?	$8.6e-09	?	Low Vol	-0.41%	6.84%	-25.81%
822	Bond	BOND	?	?	1,344 *	Low Vol	?	?	?

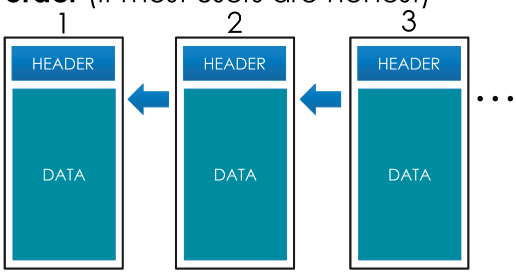
* Not Mineable
** Significantly Premined

← Back to Top 100

Total Market Cap: $39,041,518,007

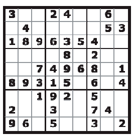
Blockchain:
A Public Ledger

- **Decentralized!**
- You can **write** but **never modify or re-order** (if most users are honest)



Bitcoin's Blockchain:
Creating Blocks (and coins)

Bitcoin **Mining**:

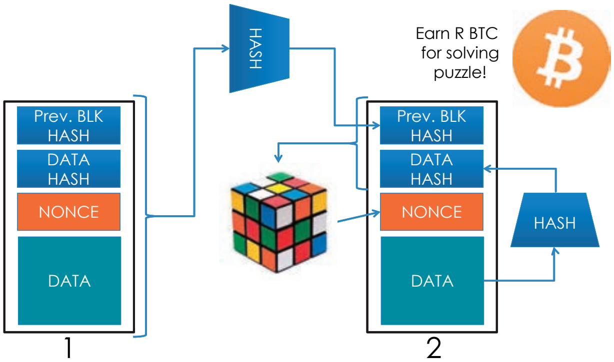


1. Everyone **tries to solve** a puzzle

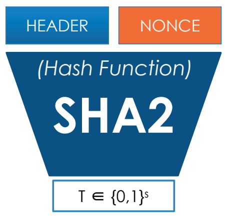
2. The **first one** to solve the puzzle **gets R BTC and generates next block**

3. The solution of **puzzle *i*** **defines puzzle *i+1***

Bitcoin's Blockchain:
Creating Blocks (and coins)



Bitcoin's Blockchain:
Proof-of-Work (PoW)

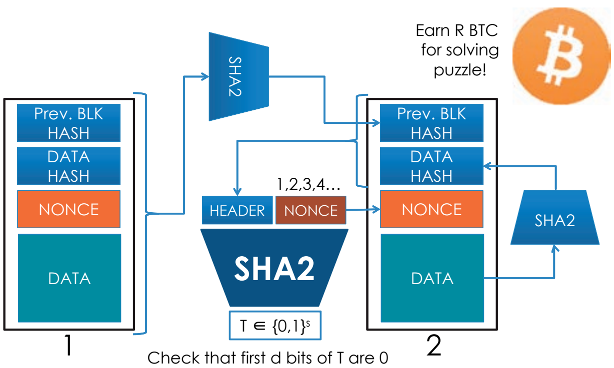


How to solve a PoW?

1. Compute HEADER
2. Set Nonce=0
3. Compute
   T=H(HEADER | NONCE)
4. If first d bits of T are 0
   output NONCE, if not
   NONCE++ and go to 3

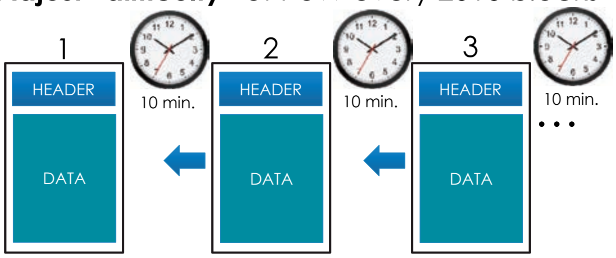
The PoW puzzle:
given Header, find Nonce
such that d first bits of T are 0

Bitcoin's Blockchain:
Creating Blocks (and coins)

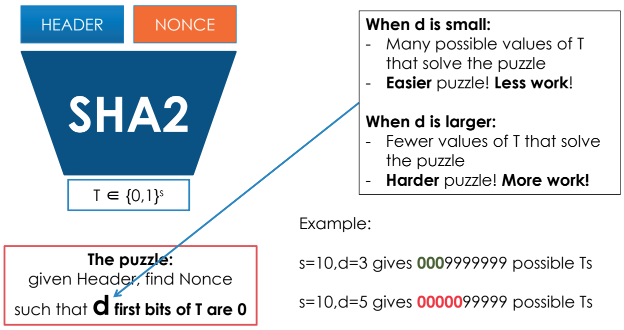


Bitcoin's Blockchain:
Proof-of-Work Difficulty

- Only **1 block** generated **every 10 minutes** on average (**1 PoW** should take **10 min.**)
- **Adjust "difficulty"** of PoW every 2016 blocks



Bitcoin's Blockchain:
Proof-of-Work Difficulty



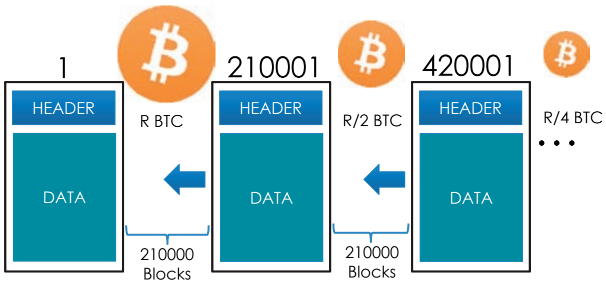
Example:

s=10,d=3 gives 0009999999 possible Ts

s=10,d=5 gives 0000099999 possible Ts

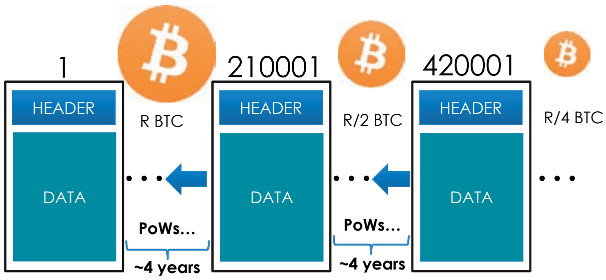
Bitcoins Blockchain:
Inflation and Halving

- Control Inflation
- Rewards halved every 210000 blocks

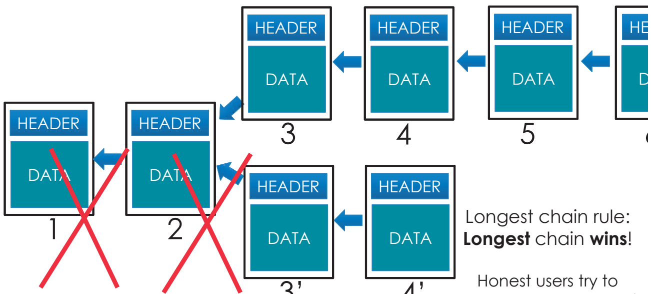


Bitcoin's Blockchain

- One block added every 10 minutes by solving puzzle
- Reward given per block, halved every 210000 blocks
- Total of 21 million BTC to be created in total



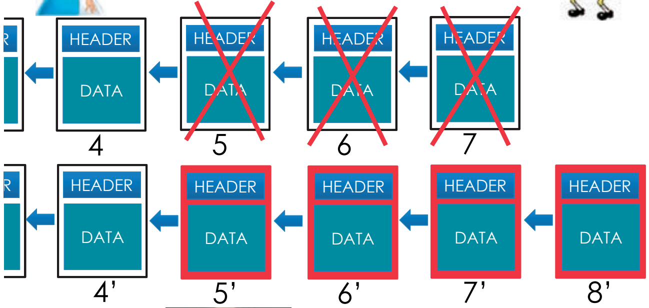
Bitcoin's Blockchain: Forks

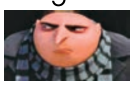


Longest chain rule:
Longest chain **wins**!

Honest users try to extend the longest chain

Bitcoin's Blockchain: Immutability





Malicious users have to **solve more puzzles,** so they hahave to control **more computer power** than honest users.

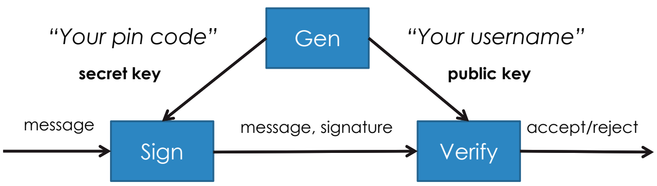
Bitcoin's Blockchain: Recap

- New block every 10 minutes
- Rewards for users who generate blocks
- Forks don't last long: consensus after 6 blocks
- Malicious users have to invest a lot of computer power to change blocks

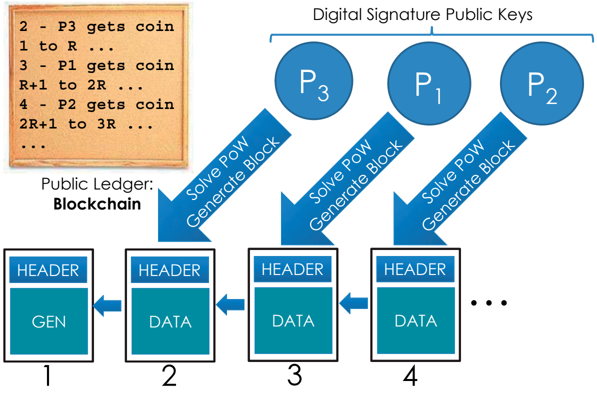
Bitcoin: How to **transfer** money

(Digital) Signatures
- *Only you can sign*
- *Everyone can verify*
- *You cannot deny*

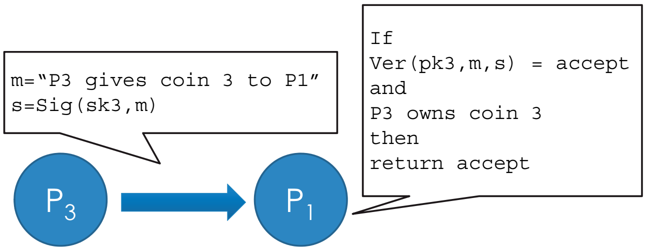
Bitcoin: How to **transfer** money



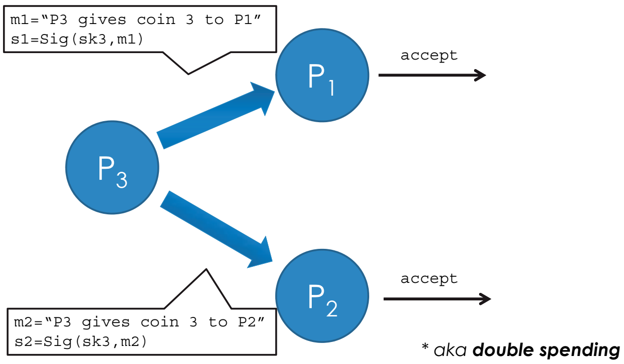
Bitcoin: How to **store** money



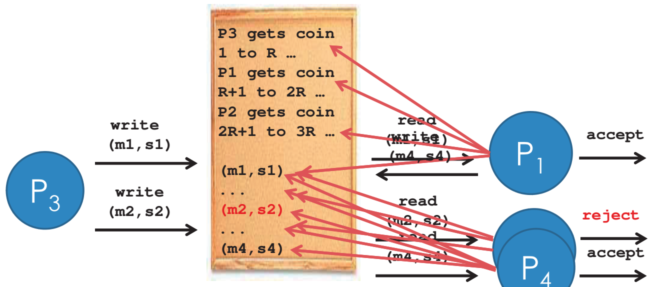
Bitcoin: How to **transfer** money



Bitcoin: How to **transfer** money
Double Spending



Bitcoin: How to **transfer** money



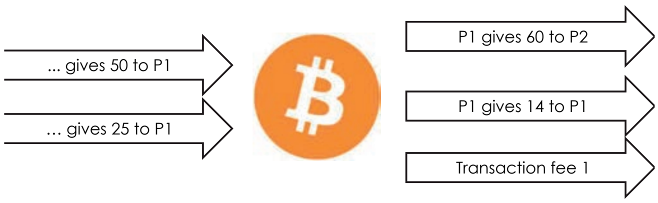

```
m1 = "P3 gives coin 3 to P1"
s1 = Sig(sk3,m1)

m2 = "P3 gives coin 3 to P2"
s2 = Sig(sk3,m2)

m4 = "P1 gives coin 3 to P4"
s4 = Sig(sk1,m4)
```

Bitcoin: How to **transfer** money
Transaction Fees

Example: P1 wants to give 60 to P2



Outline

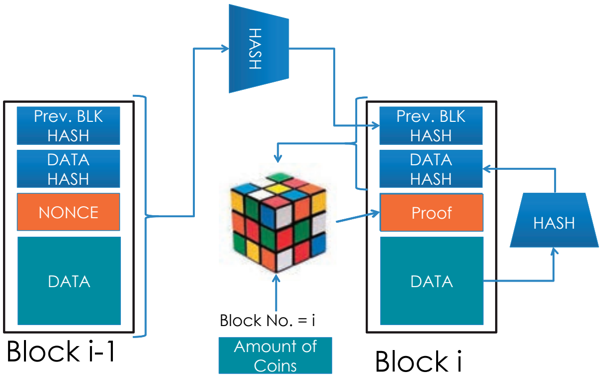
1. History: e-cash
2. Bitcoin and Blockchains
3. **Ouroboros**
4. **Ouroboros Praos**

PoW: Issues and Alternatives

- Distinction between coin holders and mining
- Diminishing rewards for mining
- Control of the network is very centralized

- Alternatives: Proof-of-Space, Proof-of-Stake

An Alternative: Proof-of-Stake



PoW vs. PoS

PoW

- More resources = more control
- Resource waste
- Centralized

PoS

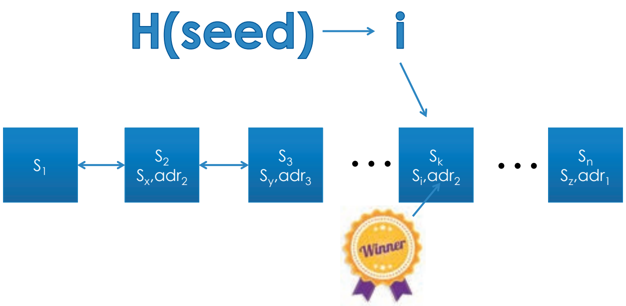
- More to lose = more control
- Less waste
- Democratic

Our Contributions [KRDO17] in Crypto 2017

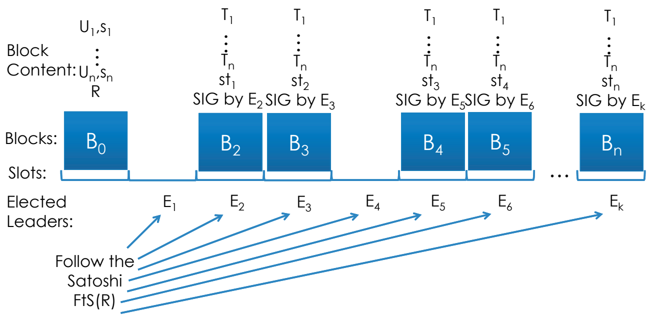
- Formalize PoS
  - *Formal model for PoS based consensus protocols*
- New PoS Based Consensus Protocol
  - *Address attacks to current protocols*
  - *Get better parameters*
  - *Get stronger security guarantees*

Follow-the-Satoshi

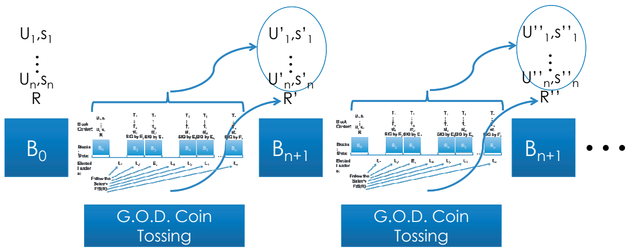
H outputs index $0 < i <$ total number of satoshis $S_1, \ldots, S_n$



The Protocol: One Epoch

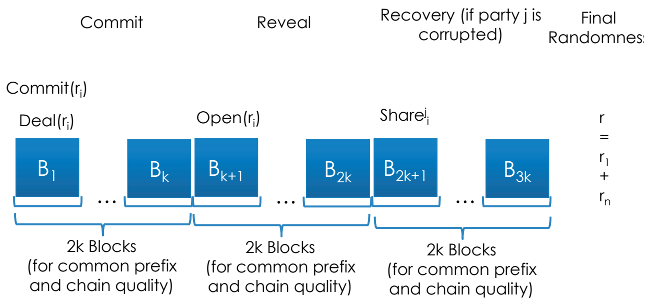


The Protocol: Multiple Epochs



G.O.D. Coin Tossing

- For every stakeholder when each epoch starts:



Building Blocks

- Verifiable secret sharing:
  - *Publicly Verifiable Secret Sharing, e.g. [C**D**17]*
- Commitments, many possibilities:
  - *ROM: H(m|r) where r is random*
  - *DDH (Pedersen) Commitments: $g^m h^r$ where $h=g^t$ and both r and t are random*

Outline

1. History: e-cash
2. Bitcoin and Blockchains
3. Ouroboros
4. **Ouroboros Praos**

Coming Soon: Ouroboros Praos

- Adaptive Security
- Semi-synchronous network: Bounded delay with upper bound unknown to honest parties
- Novel "oblivious leader selection"
- Novel Verifiable Random Functions with "malicious key generation resiliency"

Open Problems

- Prove stronger security guarantees
  - *Asynchronous Networks*
  - *Composition*
- Analyze security in a game theoretic framework
- Determine concrete parameters for Ouroboros Praos (e.g. epoch length)
- Develop a prototype of Ouroboros Praos

http://cs.au.dk/~bernardo bernardo.david@iohk.io

Panel Discussion

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

Panelists: Kazuma Ohara, Ryo Kikuchi, Mitsugu Iwamoto, Bernardo David, Yvo Desmedt, Eyal Kushilevitz and Naruhiro Kurokawa
Moderator: Kirill Morozov

The video of our panel discussion is available at "YouTube":

- https://youtu.be/nPR2f-LHqYM

MI レクチャーノートシリーズ刊行にあたり

　本レクチャーノートシリーズは、文部科学省 21 世紀 COE プログラム「機能数理学の構築と展開」（H.15-19 年度）において作成した COE Lecture Notes の続刊であり、文部科学省大学院教育改革支援プログラム「産業界が求める数学博士と新修士養成」（H19-21 年度）および、同グローバル COE プログラム「マス・フォア・インダストリ教育研究拠点」（H.20-24 年度）において行われた講義の講義録として出版されてきた。平成 23 年 4 月のマス・フォア・インダストリ研究所（IMI）設立と平成 25 年 4 月の IMI の文部科学省共同利用・共同研究拠点として「産業数学の先進的･基礎的共同研究拠点」の認定を受け、今後、レクチャーノートは、マス・フォア・インダストリに関わる国内外の研究者による講義の講義録、会議録等として出版し、マス・フォア・インダストリの本格的な展開に資するものとする。

平成 26 年 10 月
マス・フォア・インダストリ研究所
所長　福本康秀

IMI Workshop of the Joint Research Projects

Cryptographic Technologies for Securing Network Storage and Their Mathematical Modeling

発　行　　2018年３月30日
編　集　　Kirill Morozov, Hiroaki Anada, Yuji Suga
発　行　　九州大学マス・フォア・インダストリ研究所
九州大学大学院数理学府
〒819-0395 福岡市西区元岡744
九州大学数理・IMI 事務室
TEL 092-802-4402　FAX 092-802-4405
URL http://www.imi.kyushu-u.ac.jp/

印　刷　　城島印刷株式会社
〒810-0012 福岡市中央区白金２丁目９番６号
TEL 092-531-7102　FAX 092-524-4411

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note	Mitsuhiro T. NAKAO Kazuhiro YOKOYAMA	Computer Assisted Proofs - Numeric and Symbolic Approaches - 199pages	August 22, 2006
COE Lecture Note	M.J.Shai HARAN	Arithmetical Investigations - Representation theory, Orthogonal polynomials and Quantum interpolations- 174pages	August 22, 2006
COE Lecture Note Vol.3	Michal BENES Masato KIMURA Tatsuyuki NAKAKI	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2005 155pages	October 13, 2006
COE Lecture Note Vol.4	宮田　健治	辺要素有限要素法による磁界解析 - 機能数理学特別講義 21pages	May 15, 2007
COE Lecture Note Vol.5	Francois APERY	Univariate Elimination Subresultants - Bezout formula, Laurent series and vanishing conditions - 89pages	September 25, 2007
COE Lecture Note Vol.6	Michal BENES Masato KIMURA Tatsuyuki NAKAKI	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2006 209pages	October 12, 2007
COE Lecture Note Vol.7	若山　正人 中尾　充宏	九州大学産業技術数理研究センター キックオフミーティング 138pages	October 15, 2007
COE Lecture Note Vol.8	Alberto PARMEGGIANI	Introduction to the Spectral Theory of Non-Commutative Harmonic Oscillators 233pages	January 31, 2008
COE Lecture Note Vol.9	Michael I.TRIBELSKY	Introduction to Mathematical modeling 23pages	February 15, 2008
COE Lecture Note Vol.10	Jacques FARAUT	Infinite Dimensional Spherical Analysis 74pages	March 14, 2008
COE Lecture Note Vol.11	Gerrit van DIJK	Gelfand Pairs And Beyond 60pages	August 25, 2008
COE Lecture Note Vol.12	Faculty of Mathematics, Kyushu University	Consortium “MATH for INDUSTRY” First Forum 87pages	September 16, 2008
COE Lecture Note Vol.13	九州大学大学院 数理学研究院	プロシーディング「損保数理に現れる確率モデル」 — 日新火災・九州大学 共同研究 2008 年 11 月 研究会 — 82pages	February 6, 2009

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.14	Michal Beneš, Tohru Tsujikawa Shigetoshi Yazaki	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2008 77pages	February 12, 2009
COE Lecture Note Vol.15	Faculty of Mathematics, Kyushu University	International Workshop on Verified Computations and Related Topics 129pages	February 23, 2009
COE Lecture Note Vol.16	Alexander Samokhin	Volume Integral Equation Method in Problems of Mathematical Physics 50pages	February 24, 2009
COE Lecture Note Vol.17	矢嶋　　徹 及川　正行 梶原　健司 辻　　英一 福本　康秀	非線形波動の数理と物理　66pages	February 27, 2009
COE Lecture Note Vol.18	Tim Hoffmann	Discrete Differential Geometry of Curves and Surfaces　75pages	April 21, 2009
COE Lecture Note Vol.19	Ichiro Suzuki	The Pattern Formation Problem for Autonomous Mobile Robots ―Special Lecture in Functional Mathematics―　23pages	April 30, 2009
COE Lecture Note Vol.20	Yasuhide Fukumoto Yasunori Maekawa	Math-for-Industry Tutorial: Spectral theories of non-Hermitian operators and their application　184pages	June 19, 2009
COE Lecture Note Vol.21	Faculty of Mathematics, Kyushu University	Forum “Math-for-Industry” Casimir Force, Casimir Operators and the Riemann Hypothesis 95pages	November 9, 2009
COE Lecture Note Vol.22	Masakazu Suzuki Hoon Hong Hirokazu Anai Chee Yap Yousuke Sato Hiroshi Yoshida	The Joint Conference of ASCM 2009 and MACIS 2009: Asian Symposium on Computer Mathematics Mathematical Aspects of Computer and Information Sciences　436pages	December 14, 2009
COE Lecture Note Vol.23	荒川　恒男 金子　昌信	多重ゼータ値入門　111pages	February 15, 2010
COE Lecture Note Vol.24	Fulton B.Gonzalez	Notes on Integral Geometry and Harmonic Analysis　125pages	March 12, 2010
COE Lecture Note Vol.25	Wayne Rossman	Discrete Constant Mean Curvature Surfaces via Conserved Quantities 130pages	May 31, 2010
COE Lecture Note Vol.26	Mihai Ciucu	Perfect Matchings and Applications　66pages	July 2, 2010

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.27	九州大学大学院 数理学研究院	Forum “Math-for-Industry” and Study Group Workshop Information security, visualization, and inverse problems, on the basis of optimization techniques 100pages	October 21, 2010
COE Lecture Note Vol.28	ANDREAS LANGER	MODULAR FORMS, ELLIPTIC AND MODULAR CURVES LECTURES AT KYUSHU UNIVERSITY 2010 62pages	November 26, 2010
COE Lecture Note Vol.29	木田　雅成 原田　昌晃 横山　俊一	Magma で広がる数学の世界 157pages	December 27, 2010
COE Lecture Note Vol.30	原　　　隆 松井　　卓 廣島　文生	Mathematical Quantum Field Theory and Renormalization Theory 201pages	January 31, 2011
COE Lecture Note Vol.31	若山　正人 福本　康秀 高木　　剛 山本　昌宏	Study Group Workshop 2010 Lecture & Report 128pages	February 8, 2011
COE Lecture Note Vol.32	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2011 “TSUNAMI-Mathematical Modelling” Using Mathematics for Natural Disaster Prediction, Recovery and Provision for the Future 90pages	September 30, 2011
COE Lecture Note Vol.33	若山　正人 福本　康秀 高木　　剛 山本　昌宏	Study Group Workshop 2011 Lecture & Report 140pages	October 27, 2011
COE Lecture Note Vol.34	Adrian Muntean Vladimír Chalupecký	Homogenization Method and Multiscale Modeling 72pages	October 28, 2011
COE Lecture Note Vol.35	横山　俊一 夫　　紀恵 林　　卓也	計算機代数システムの進展 210pages	November 30, 2011
COE Lecture Note Vol.36	Michal Beneš Masato Kimura Shigetoshi Yazaki	Proceedings of Czech-Japanese Seminar in Applied Mathematics 2010 107pages	January 27, 2012
COE Lecture Note Vol.37	若山　正人 高木　　剛 Kirill Morozov 平岡　裕章 木村　正人 白井　朋之 西井　龍映 栄　伸一郎 穴井　宏和 福本　康秀	平成 23 年度 数学・数理科学と諸科学・産業との連携研究ワークショップ　拡がっていく数学　～期待される “見えない力” ～ 154pages	February 20, 2012

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.38	Fumio Hiroshima Itaru Sasaki Herbert Spohn Akito Suzuki	Enhanced Binding in Quantum Field Theory　204pages	March 12, 2012
COE Lecture Note Vol.39	Institute of Mathematics for Industry, Kyushu University	Multiscale Mathematics; Hierarchy of collective phenomena and interrelations between hierarchical structures　180pages	March 13, 2012
COE Lecture Note Vol.40	井ノ口順一 太田　泰広 筧　　三郎 梶原　健司 松浦　　望	離散可積分系・離散微分幾何チュートリアル 2012　152pages	March 15, 2012
COE Lecture Note Vol.41	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2012 “Information Recovery and Discovery”　91pages	October 22, 2012
COE Lecture Note Vol.42	佐伯　　修 若山　正人 山本　昌宏	Study Group Workshop 2012 Abstract, Lecture & Report　178pages	November 19, 2012
COE Lecture Note Vol.43	Institute of Mathematics for Industry, Kyushu University	Combinatorics and Numerical Analysis Joint Workshop　103pages	December 27, 2012
COE Lecture Note Vol.44	萩原　　学	モダン符号理論からポストモダン符号理論への展望　107pages	January 30, 2013
COE Lecture Note Vol.45	金山　　寛	Joint Research Workshop of Institute of Mathematics for Industry (IMI), Kyushu University “Propagation of Ultra-large-scale Computation by the Domain-decomposition-method for Industrial Problems (PUCDIP 2012)” 121pages	February 19, 2013
COE Lecture Note Vol.46	西井　龍映 栄　伸一郎 岡田　勘三 落合　啓之 小磯　深幸 斎藤　新悟 白井　朋之	科学・技術の研究課題への数学アプローチ ―数学モデリングの基礎と展開―　325pages	February 28, 2013
COE Lecture Note Vol.47	SOO TECK LEE	BRANCHING RULES AND BRANCHING ALGEBRAS FOR THE COMPLEX CLASSICAL GROUPS　40pages	March 8, 2013
COE Lecture Note Vol.48	溝口　佳寛 脇　　隼人 平坂　　貢 谷口　哲至 島袋　　修	博多ワークショップ「組み合わせとその応用」　124pages	March 28, 2013

シリーズ既刊

Issue	Author／Editor	Title	Published
COE Lecture Note Vol.49	照井　　章 小原　功任 濱田　龍義 横山　俊一 穴井　宏和 横田　博史	マス・フォア・インダストリ研究所　共同利用研究集会 II 数式処理研究と産学連携の新たな発展　137pages	August 9, 2013
MI Lecture Note Vol.50	Ken Anjyo Hiroyuki Ochiai Yoshinori Dobashi Yoshihiro Mizoguchi Shizuo Kaji	Symposium MEIS2013: Mathematical Progress in Expressive Image Synthesis　154pages	October 21, 2013
MI Lecture Note Vol.51	Institute of Mathematics for Industry, Kyushu University	Forum "Math-for-Industry" 2013 "The Impact of Applications on Mathematics"　97pages	October 30, 2013
MI Lecture Note Vol.52	佐伯　　修 岡田　勘三 髙木　　剛 若山　正人 山本　昌宏	Study Group Workshop 2013 Abstract, Lecture ＆ Report 142pages	November 15, 2013
MI Lecture Note Vol.53	四方　義啓 櫻井　幸一 安田　貴徳 Xavier Dahan	平成25年度　九州大学マス・フォア・インダストリ研究所 共同利用研究集会　安全・安心社会基盤構築のための代数構造 ～サイバー社会の信頼性確保のための数理学～　158pages	December 26, 2013
MI Lecture Note Vol.54	Takashi Takiguchi Hiroshi Fujiwara	Inverse problems for practice, the present and the future　93pages	January 30, 2014
MI Lecture Note Vol.55	栄　伸一郎 溝口　佳寛 脇　　隼人 渋田　敬史	Study Group Workshop 2013 数学協働プログラム Lecture & Report 98pages	February 10, 2014
MI Lecture Note Vol.56	Yoshihiro Mizoguchi Hayato Waki Takafumi Shibuta Tetsuji Taniguchi Osamu Shimabukuro Makoto Tagami Hirotake Kurihara Shuya Chiba	Hakata Workshop 2014 ~ Discrete Mathematics and its Applications ~　141pages	March 28, 2014
MI Lecture Note Vol.57	Institute of Mathematics for Industry, Kyushu University	Forum "Math-for-Industry" 2014: "Applications + Practical Conceptualization + Mathematics = fruitful Innovation"　93pages	October 23, 2014
MI Lecture Note Vol.58	安生健一 落合啓之	Symposium MEIS2014: Mathematical Progress in Expressive Image Synthesis　135pages	November 12, 2014

シリーズ既刊

Issue	Author／Editor	Title	Published
MI Lecture Note Vol.59	西井　龍映 岡田　勘三 梶原　健司 高木　　剛 若山　正人 脇　　隼人 山本　昌宏	Study Group Workshop 2014 数学協働プログラム Abstract, Lecture & Report　196pages	November 14, 2014
MI Lecture Note Vol.60	西浦　　博	平成 26 年度九州大学 IMI 共同利用研究・研究集会（I） 感染症数理モデルの実用化と産業及び政策での活用のための新たな展開　120pages	November 28, 2014
MI Lecture Note Vol.61	溝口　佳寛 Jacques Garrigue 萩原　　学 Reynald Affeldt	研究集会 高信頼な理論と実装のための定理証明および定理証明器 Theorem proving and provers for reliable theory and implementations (TPP2014)　138pages	February 26, 2015
MI Lecture Note Vol.62	白井　朋之	Workshop on “β-transformation and related topics”　59pages	March 10, 2015
MI Lecture Note Vol.63	白井　朋之	Workshop on “Probabilistic models with determinantal structure” 107pages	August 20, 2015
MI Lecture Note Vol.64	落合　啓之 土橋　宜典	Symposium MEIS2015: Mathematical Progress in Expressive Image Synthesis　124pages	September 18, 2015
MI Lecture Note Vol.65	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2015 “The Role and Importance of Mathematics in Innovation”　74pages	October 23, 2015
MI Lecture Note Vol.66	岡田　勘三 藤澤　克己 白井　朋之 若山　正人 脇　　隼人 Philip Broadbridge 山本　昌宏	Study Group Workshop 2015 Abstract, Lecture & Report 156pages	November 5, 2015
MI Lecture Note Vol.67	Institute of Mathematics for Industry, Kyushu University	IMI-La Trobe Joint Conference “Mathematics for Materials Science and Processing” 66pages	February 5, 2016
MI Lecture Note Vol.68	古庄　英和 小谷　久寿 新甫　洋史	結び目と Grothendieck-Teichmüller 群 116pages	February 22, 2016
MI Lecture Note Vol.69	土橋　宜典 鍛治　静雄	Symposium MEIS2016: Mathematical Progress in Expressive Image Synthesis　82pages	October 24, 2016
MI Lecture Note Vol.70	Institute of Mathematics for Industry, Kyushu University	Forum “Math-for-Industry” 2016 “Agriculture as a metaphor for creativity in all human endeavors” 98pages	November 2, 2016
MI Lecture Note Vol.71	小磯　深幸 二宮　嘉行 山本　昌宏	Study Group Workshop 2016 Abstract, Lecture & Report　143pages	November 21, 2016

シリーズ既刊

Issue	Author／Editor	Title	Published
MI Lecture Note Vol.72	新井　朝雄 小嶋　　泉 廣島　文生	Mathematical quantum field theory and related topics　133pages	January 27, 2017
MI Lecture Note Vol.73	穴田　啓晃 Kirill Morozov 須賀　祐治 奥村　伸也 櫻井　幸一	Secret Sharing for Dependability, Usability and Security of Network Storage and Its Mathematical Modeling　211pages	March 15, 2017
MI Lecture Note Vol.74	QUISPEL, G. Reinout W. BADER, Philipp MCLAREN, David I. TAGAMI, Daisuke	IMI-La Trobe Joint Conference Geometric Numerical Integration and its Applications　71pages	March 31, 2017
MI Lecture Note Vol.75	手塚　　集 田上　大助 山本　昌宏	Study Group Workshop 2017 Abstract, Lecture & Report　118pages	October 20, 2017
MI Lecture Note Vol.76	宇田川誠一	Tzitzéica 方程式の有限間隙解に付随した極小曲面の構成理論 ―Tzitzéica 方程式の楕円関数解を出発点として―　68pages	August 4, 2017
MI Lecture Note Vol.77	松谷　茂樹 佐伯　　修 中川　淳一 田上　大助 上坂　正晃 Pierluigi Cesana 濵田　裕康	平成 29 年度　九州大学マス・フォア・インダストリ研究所 共同利用研究集会（I） 結晶の界面，転位，構造の数理　148pages	December 20, 2017
MI Lecture Note Vol.78	瀧澤　重志 小林　和博 佐藤憲一郎 斎藤　　努 清水　正明 間瀬　正啓 藤澤　克樹 神山　直之	平成 29 年度　九州大学マス・フォア・インダストリ研究所 プロジェクト研究　研究集会（I） 防災・避難計画の数理モデルの高度化と社会実装へ向けて 136pages	February 26, 2018
MI Lecture Note Vol.79	神山　直之 畔上　秀幸	平成 29 年度　AIMaP チュートリアル 最適化理論の基礎と応用　96pages	February 28, 2018

Institute of Mathematics for Industry
Kyushu University

九州大学マス・フォア・インダストリ研究所
九州大学大学院 数理学府

〒819-0395 福岡市西区元岡744　TEL 092-802-4402　FAX 092-802-4405
URL http://www.imi.kyushu-u.ac.jp/