Privacy-Aware Data Analytics

Problems, Capabilities and Challenges

Shantanu Rane, PARC.

Security & Privacy @ PARC

Ersin Uzun

Shantanu Rane

Alex Brito

Vanishree Rao

What will you get from this course?

1. Understand the relevance of privacy-preserving analytics in today's world.
2. Understand the stakeholders in the analytics setting and their motivations.
3. An overview of the main application scenarios for privacy-preserving analytics.
4. An overview of the main tools at our disposal, their advantages and limitations.
5. What challenges remain, and why they are difficult. A sampling of new research directions.

Overall Organization

- Motivation
- Stakeholders in the analytics setting
- Background
- Problem settings and Tools
  - Computationally private approaches
  - Information-theoretically private approaches
  - Statistical anonymization methods
- Use Cases
- Major Challenges & Research Directions

1

Motivation

Indiscriminate data collection

Bigger & bigger data breaches

Gap between theory & practice

Some motivating applications

http://posttypography.com

Social network data

Smartphone app data

Online shopping

Car navigation data

Biometrics

Healthcare data

Internet of things telemetry

Smart grid pricing & usage

Intellectual property

Industrial diagnostics data

Demographic data

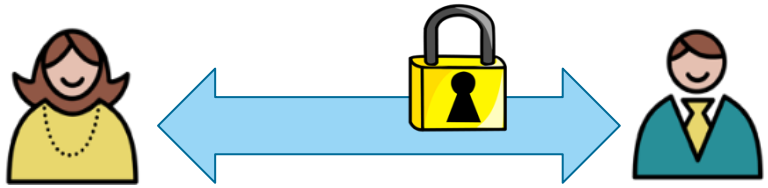
National security data

⋮

Today

Privacy in Storage & Transmission

E.g., Full disk encryption



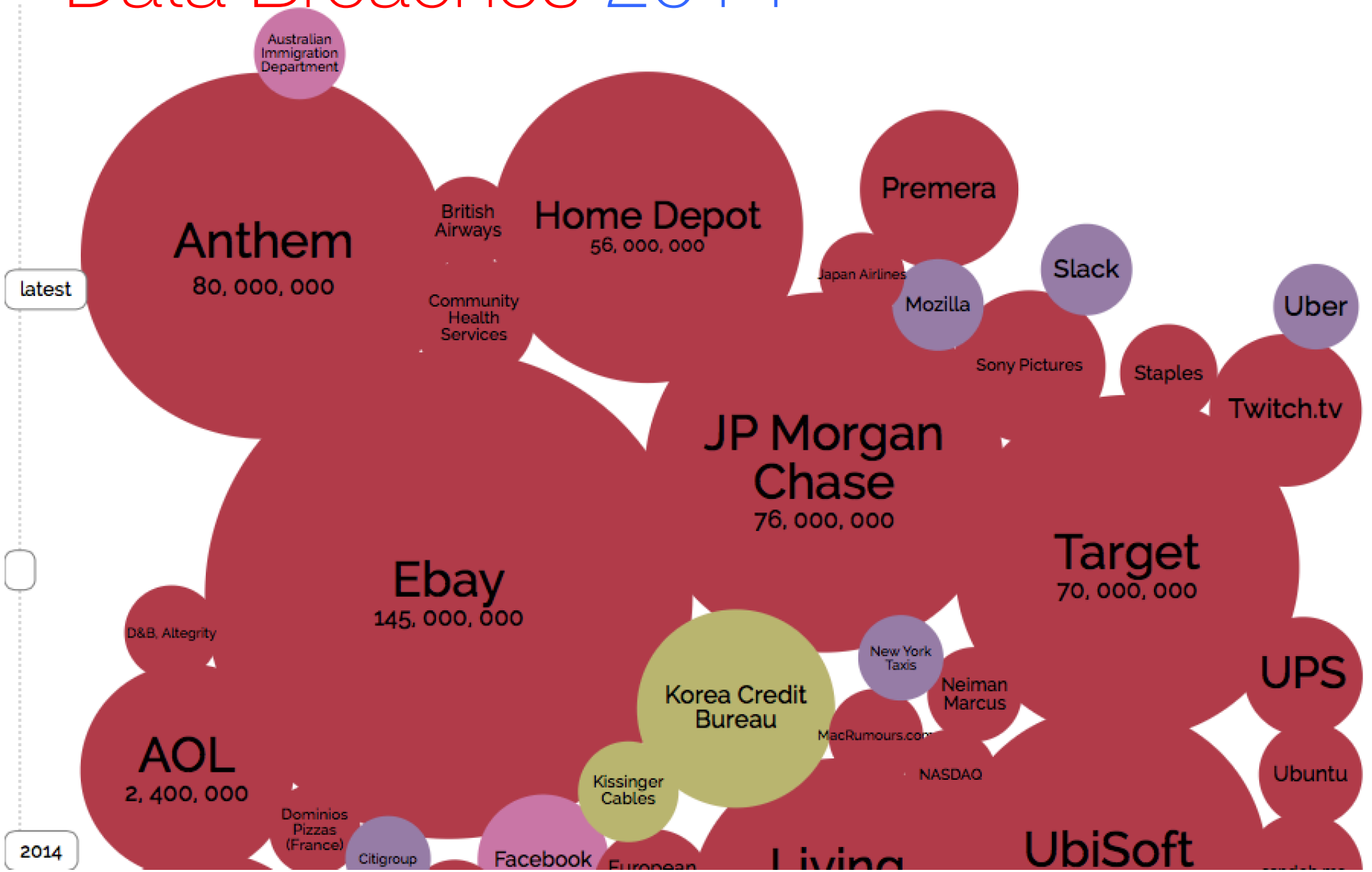
E.g., SSL / TLS

http://informationisbeautiful.net

Data Breaches 2011 2012 2013

http://informationisbeautiful.net

Data Breaches 2014



Facebook Privacy Settings Example

People are sharing increasing amounts of personal information on social networks, without realizing it.

http://mattmckeon.com/facebook-privacy/

Can you find the logout button on Facebook?

"**Recommendation 3**: … the NITRD agencies, should strengthen U.S. research in privacy-related technologies and in the relevant areas of social science that inform the successful application of those technologies."

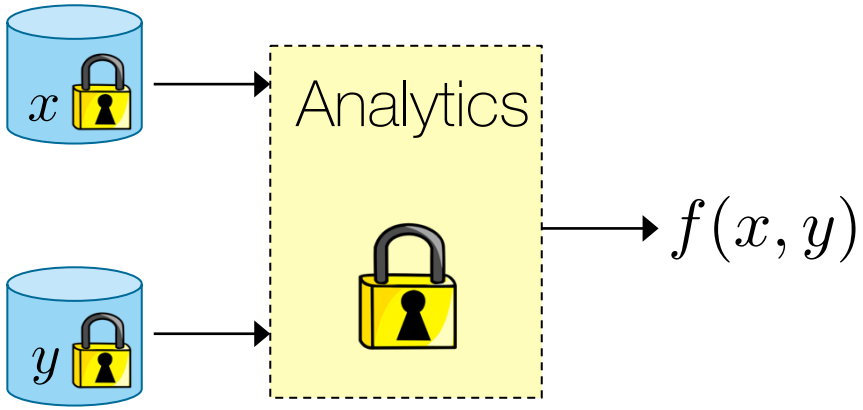
"…. create appropriate balance among economic opportunity, national priorities, and privacy protection."

[PCAST Report, May 2014]

Breaches happen at the interfaces where plaintext data is exposed.

E.g., Full disk encryption

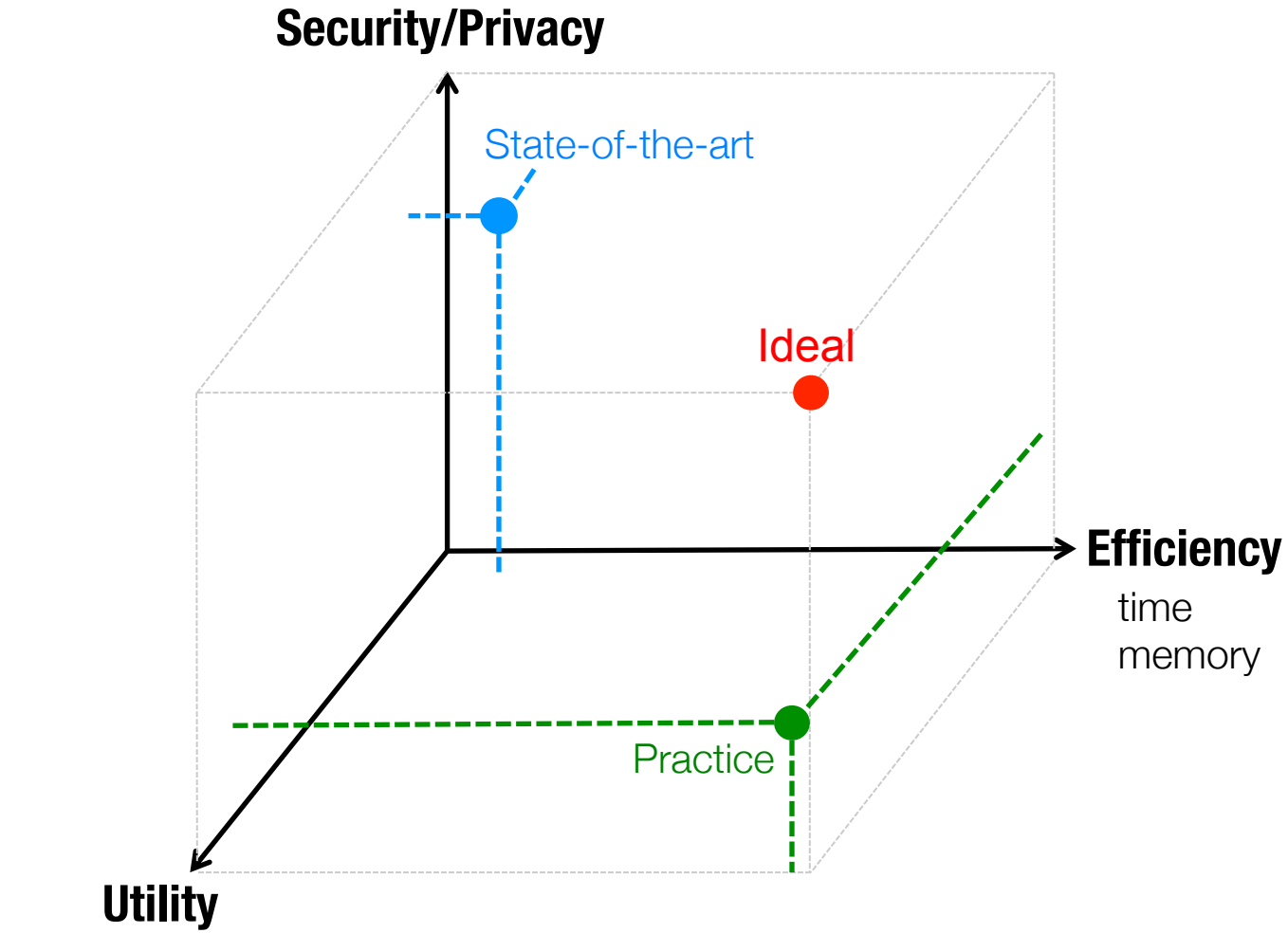
E.g., SSL



Future

Privacy in Storage, Transmission, & Computation

Privacy Research vs Deployment



2

Data Analytics Setting

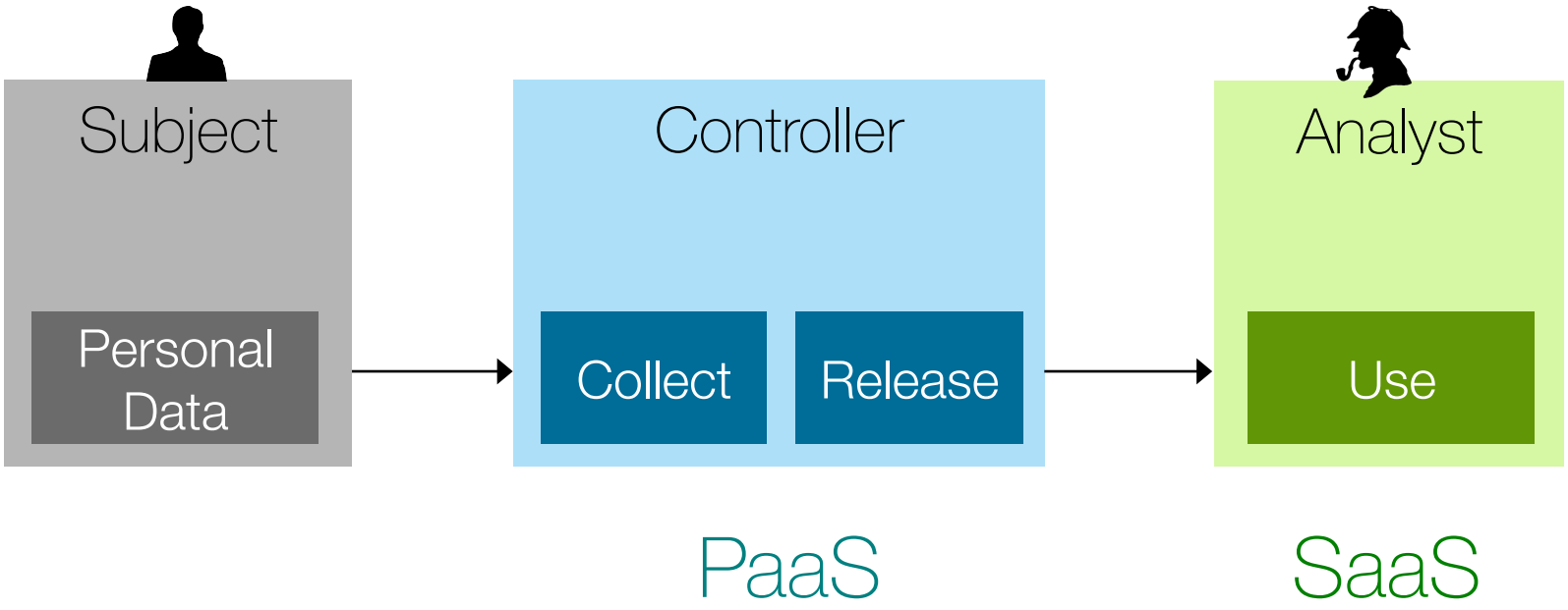
Stakeholders

Personal & Enterprise settings

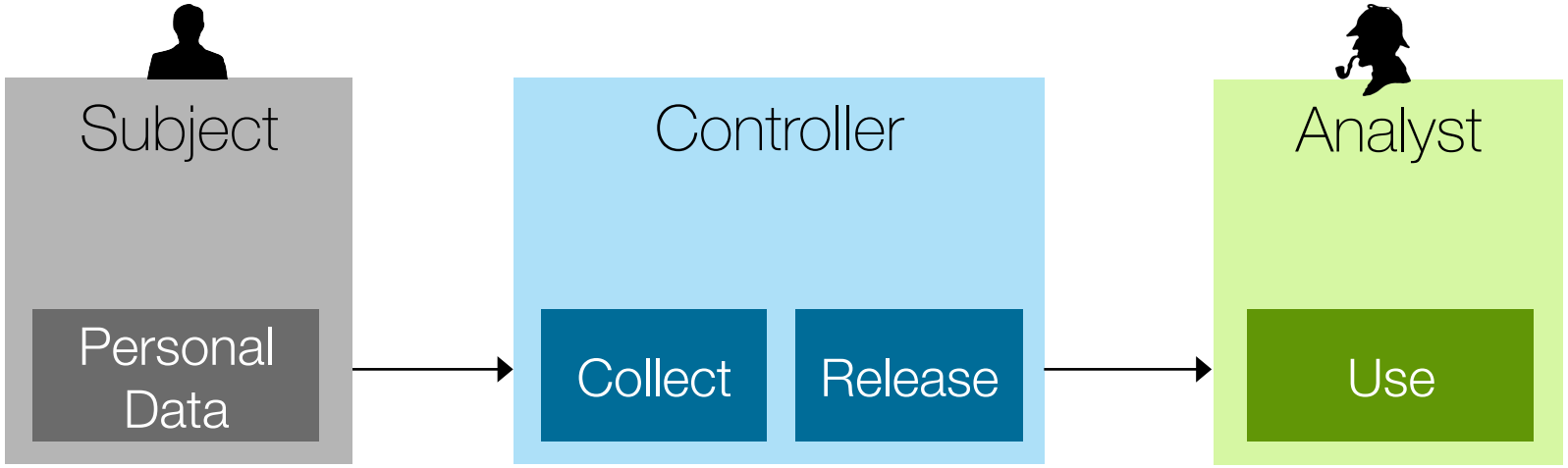
Technical Areas

- Sharing
- Mining
- Anonymization

Data Analytics Setting

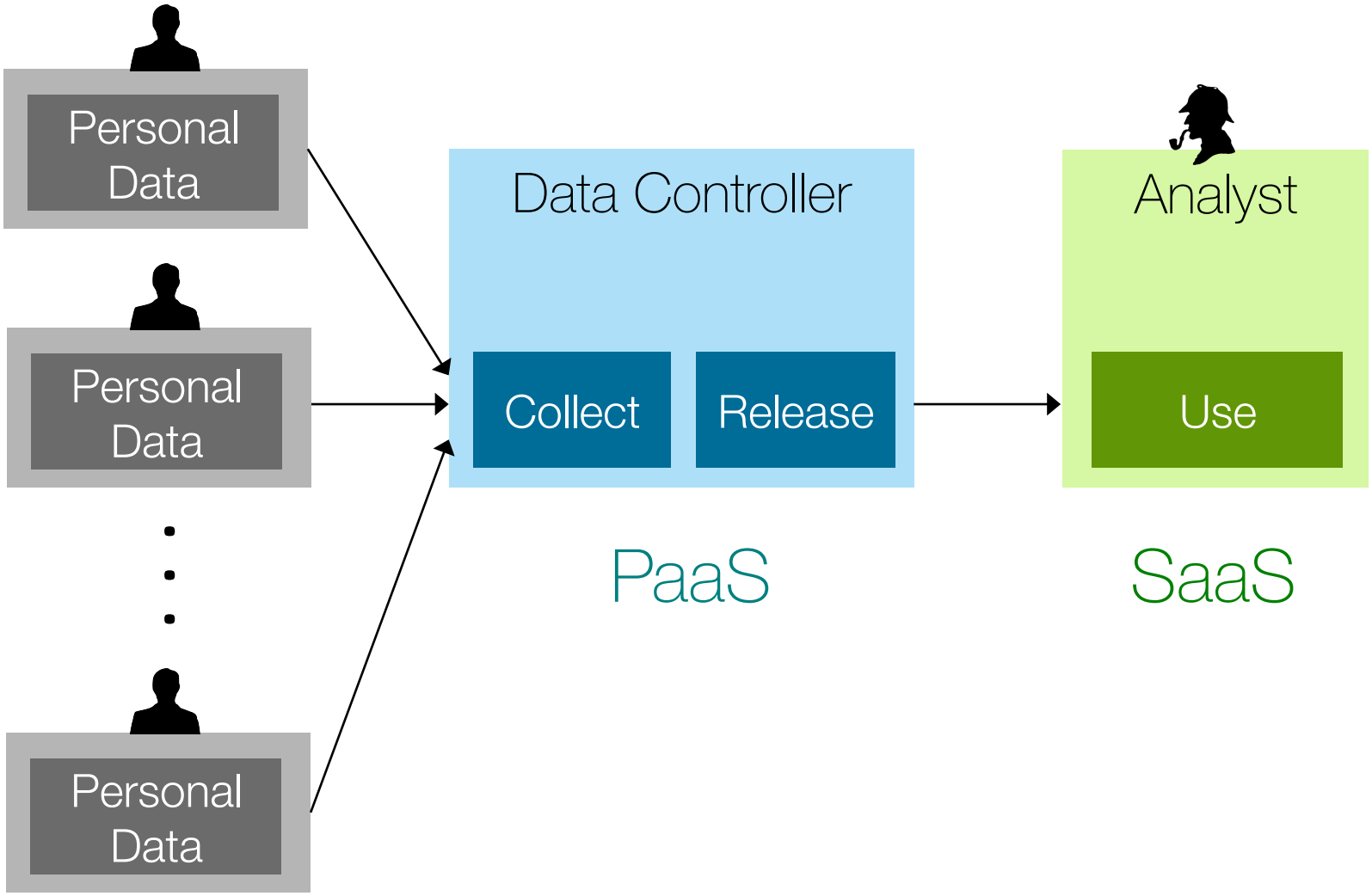


Privacy & Security Requirements



Subject	Controller	Analyst
Prevent Disclosure	Prevent Disclosure	Protect Expertise
Control Use	Control Use	
	Control Liability	

Personal Privacy: Setting



Personal Privacy: Applications

Social Networks

Fitness and Wearables

Location Services

Shopping & Recommendations

Payments

Driving & Commuting

Smart-meter Privacy

E-voting

Internet-of-Things

Private Behavioral Targeting

Browsing History
Installed Apps
Search History
Social Network

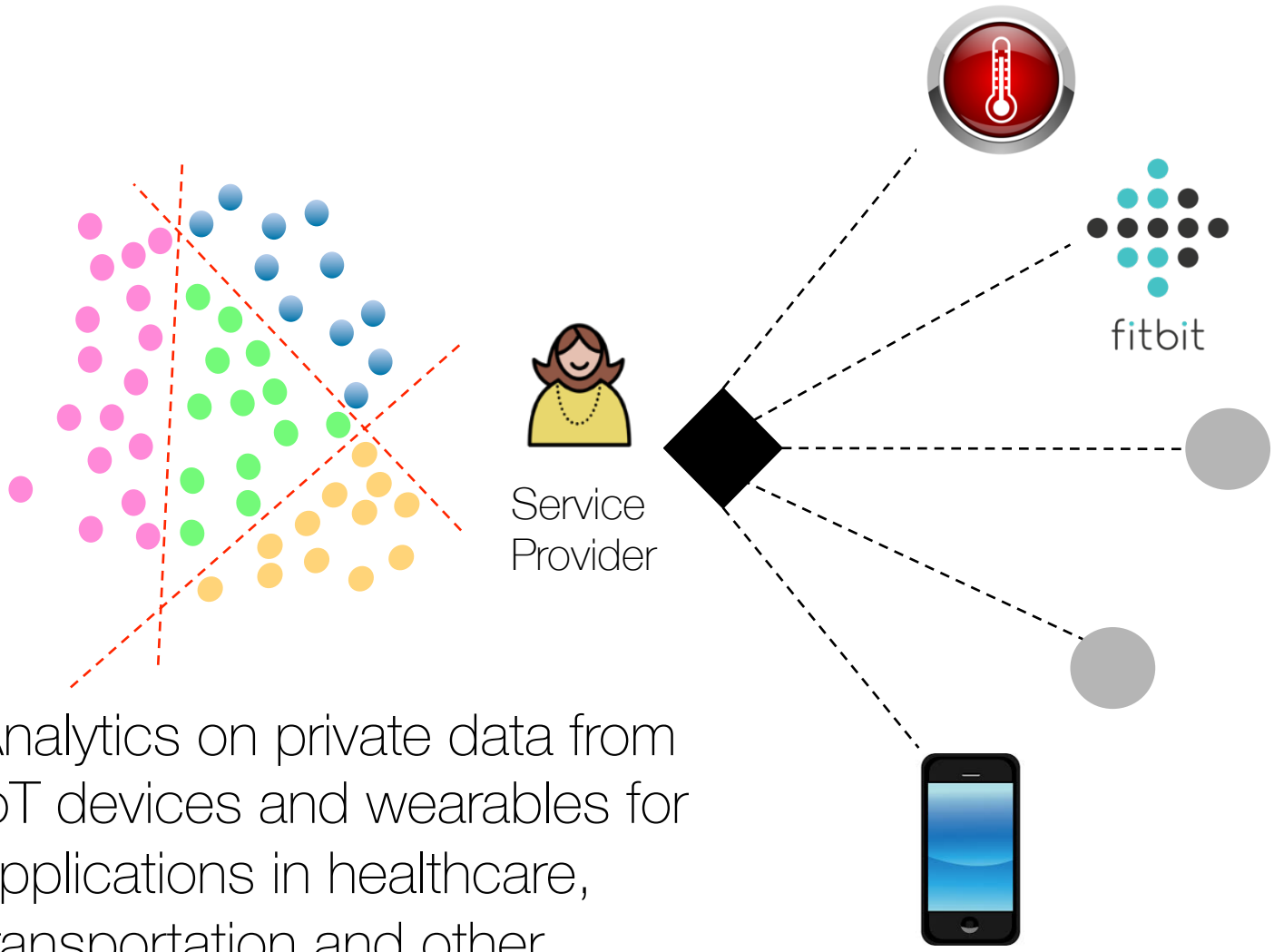
Today

Encrypted
Coupon

Vendor

Target User Profile

Secure Data Aggregation for IoT



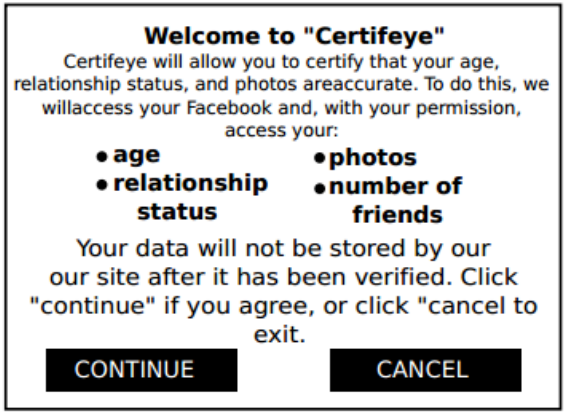
Analytics on private data from IoT devices and wearables for applications in healthcare, transportation and other domains.

Privacy Preserving Social Networks

Emerging Technology From the arXiv
June 27, 2013

Tackling Online Dating's Biggest Conundrum

When you read the profile of a potential partner, how do you know it's true? Researchers at Xerox's PARC think they have the answer



Online dating has changed the way people start relationships. In 2000, a few hundred thousand individuals were experimenting with online dating. Today, more than 40 million people have signed up to meet their dream man or woman online.

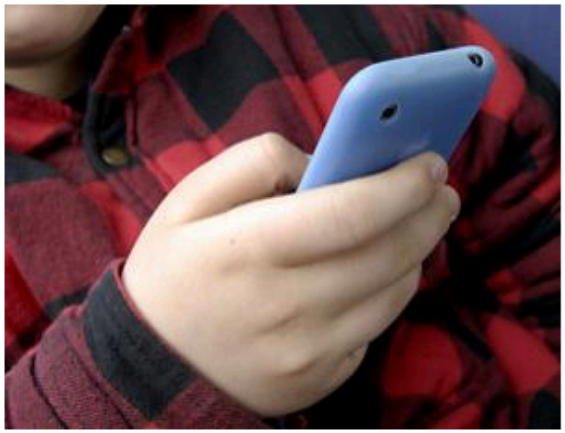
That kind of success is reflected in the fact that this industry is currently worth some $1.9 billion in annual revenue.

Of course, nobody would claim that online dating is the perfect way to meet a mate. One problem in particular is whether to trust the information that a potential date has given. How do you know that this person isn't being economical with the truth?

Private Personalized Healthcare

TECHNOLOGY NEWS 31 October 2012

Want to keep your genome safe? There's an app for that



NOW there is a smartphone app that will allow you to carry around an encrypted copy of your genome, safe in the knowledge that the DNA won't fall into the wrong hands. With prices for DNA sequencing falling fast, this app may not be as futuristic as it sounds.

The idea behind Genodroid, the work of a team led by Emiliano De Cristofaro at Xerox's PARC lab in Palo Alto, California, is to investigate how people might safely transport the personal information stored in their genome. At the moment, sequencing remains a boutique industry, but already 23&Me based nearby in Mountain View can provide a limited genome for just $299.

Personal Privacy: Characteristics

Up to millions of consumer-grade devices communicate with one or more cloud-based service providers.

→ Must push complex computation to service providers.

Consumers (generally) don't communicate directly with each other. E.g., smart meters, fitness competition.

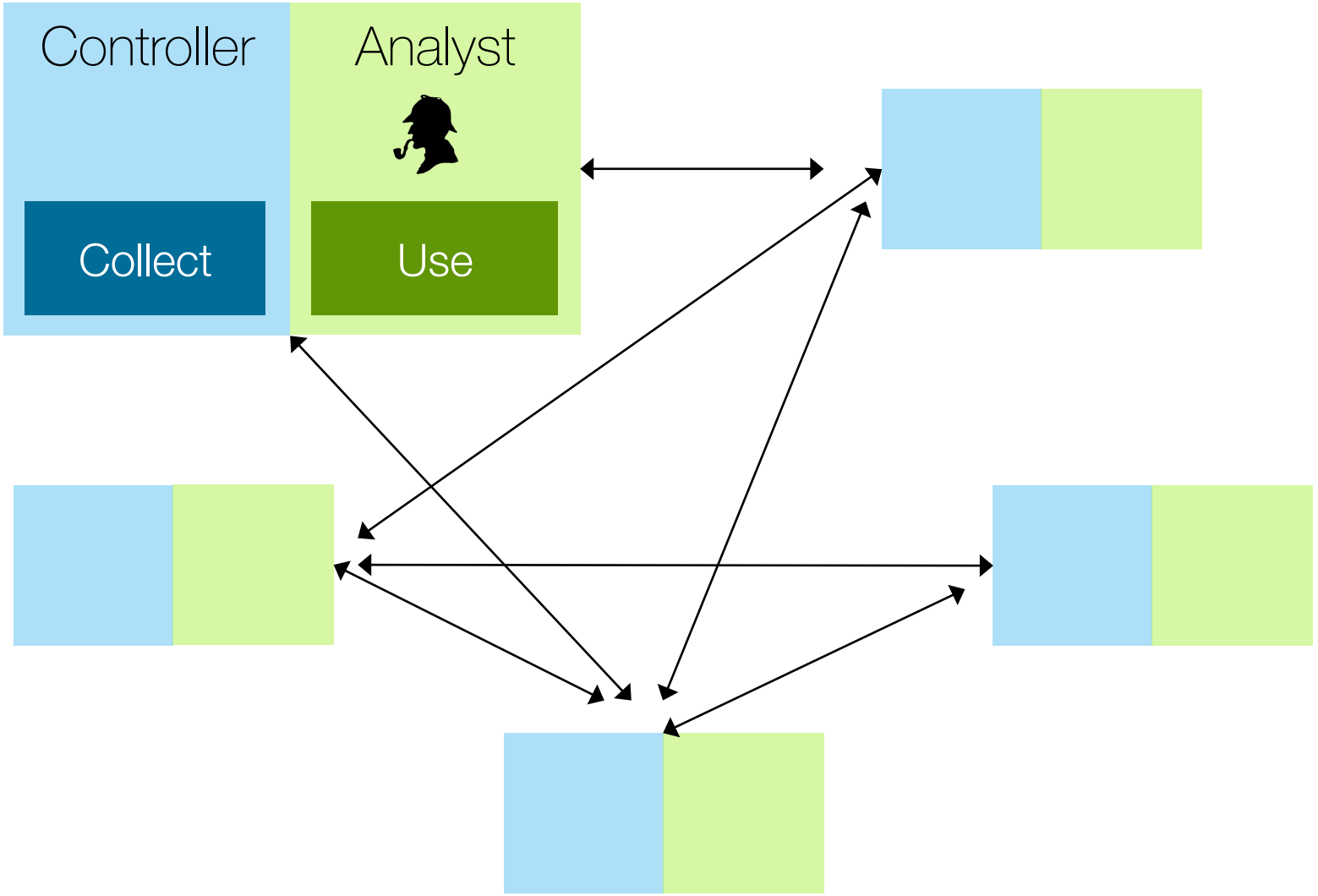
→ Only service provider has cross-participant information.

Consumers are not all online at the same time.

→ Analytics algorithms must be fault-tolerant.

→ Protocols should have few rounds of interaction.

Enterprise Privacy: Setting



Enterprise Privacy: Applications

Healthcare & Wellness: Medical analytics; Pharmacogenomics; Wearables and Outcome-based Healthcare; Fraud, waste and abuse; Insurance analytics

Security: Homeland security, cyber-threat mitigation

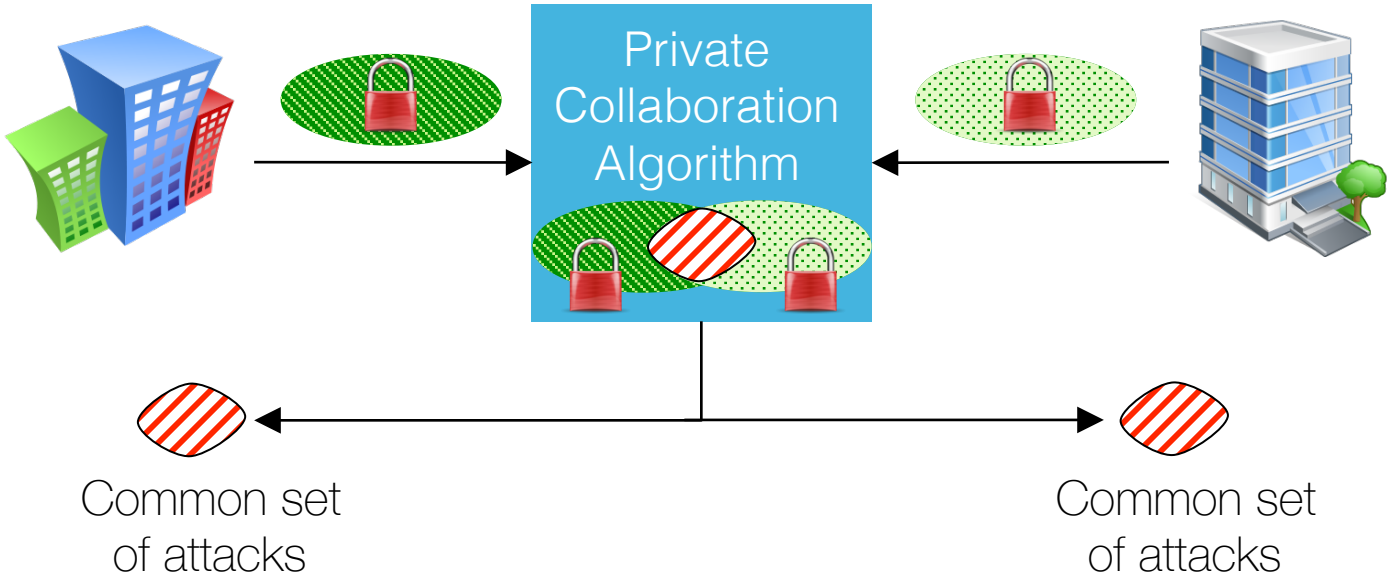
Internet Commerce: Recommender systems, knowledge monetization, customer care analytics

Smart cities: Power grid analytics and pricing, demographics data, emergency preparedness, urban mobility

Transportation: Toll collection, red light violations

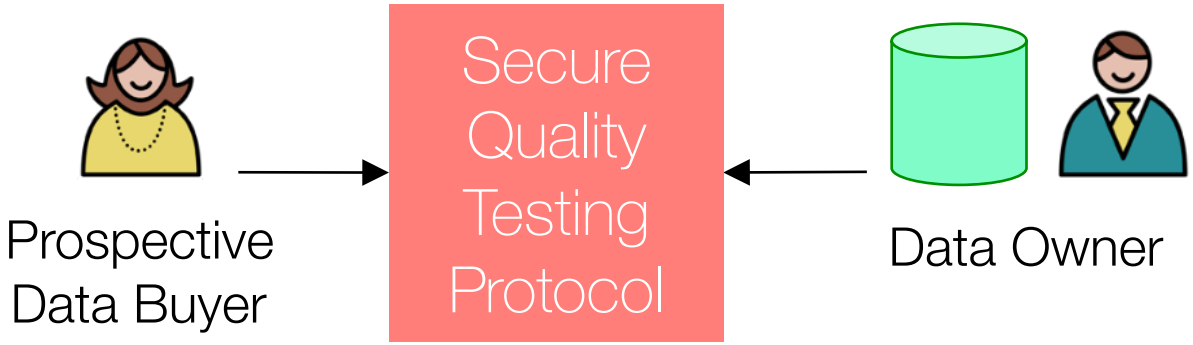
Miscellaneous: Data quality assessment, education analytics

Cyber-threat Mitigation



Cryptographic protocols enable companies **to share information** about common cyber-attacks and develop a cyber-threat mitigation strategy **without privacy concerns**.

Data Quality Assessment



Data cleaning is one of the costliest operations in data analytics. Estimated 600B annually. [Eckerson, 2002]

Privacy-aware protocols allow prospective buyer to test owner's data quality before purchase **without seeing data**.

Check for completeness, validity, timeliness, consistency, and other metrics.

Enterprise Privacy: Characteristics

Pairwise communication amongst a *smaller* number of powerful, possibly cloud-based machines.

→ Can tolerate high computational complexity.

Parties can remain online throughout the computation.

→ Can handle multiple rounds of interaction.

→ Might need to protect algorithms / expertise / intellectual property of one or more participants. This is hard to do, as interactions leak information about the algorithm.

Privacy-Preserving Data Sharing

Privacy considerations

Share common data w/o revealing unique data

Applications

Cyber threat mitigation, recommendation engines, knowledge monetization

Privacy-preserving Data Mining

Privacy Considerations

1. Analytics & machine learning w/o leaking sensitive info.
2. Preserve database privacy and query privacy.

Applications

Federated search, Healthcare analytics, Data quality assessment, Education analytics, Call graph analysis, Transportation analytics, too many to list.

(Statistical) Anonymization

Privacy Considerations

1. Identify and anonymize sensitive attributes.
2. Evaluate privacy-utility tradeoff.
3. Evaluate risk of de-anonymization via external linkage

Applications

Disclosure control methods for advertising, healthcare, smart grid, education analytics, etc.

3

Background

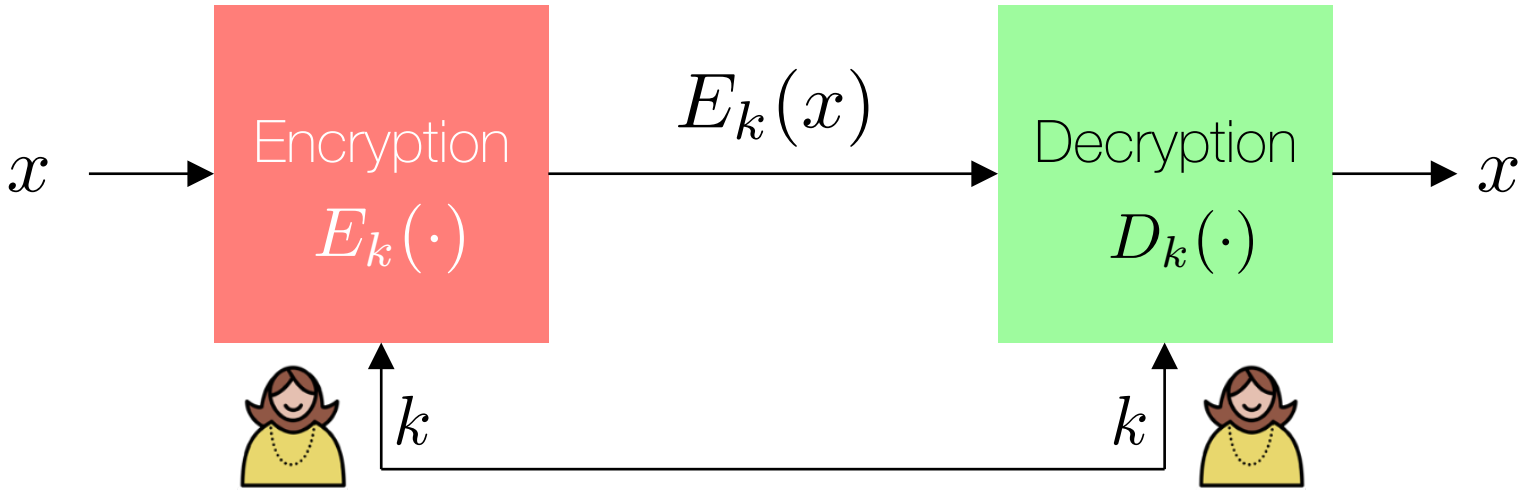
Symmetric-Key Crypto

Public-Key Crypto

Adversarial Models

Oblivious Transfer

Symmetric Key Crypto



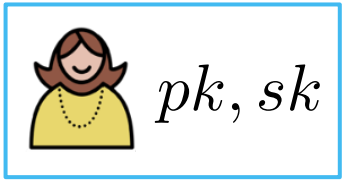
Generally, no ciphertext expansion, i.e., size of ciphertext is the same as that of the plain text.

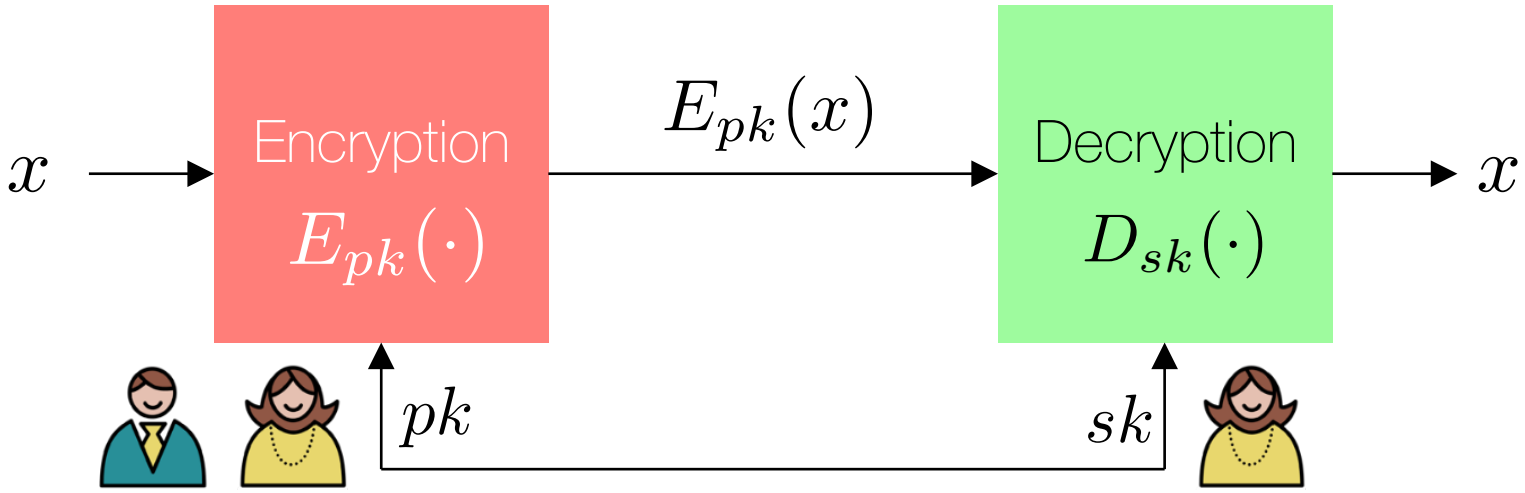
Used in construction of hashes and other crypto primitives.

Examples: 3DES (Data Encryption Standard), AES (Advanced Encryption Standard / Rijndael). Typical key sizes: 128, 256, 512 bits

[Daemen, Rijmen, 13]

Public Key Crypto





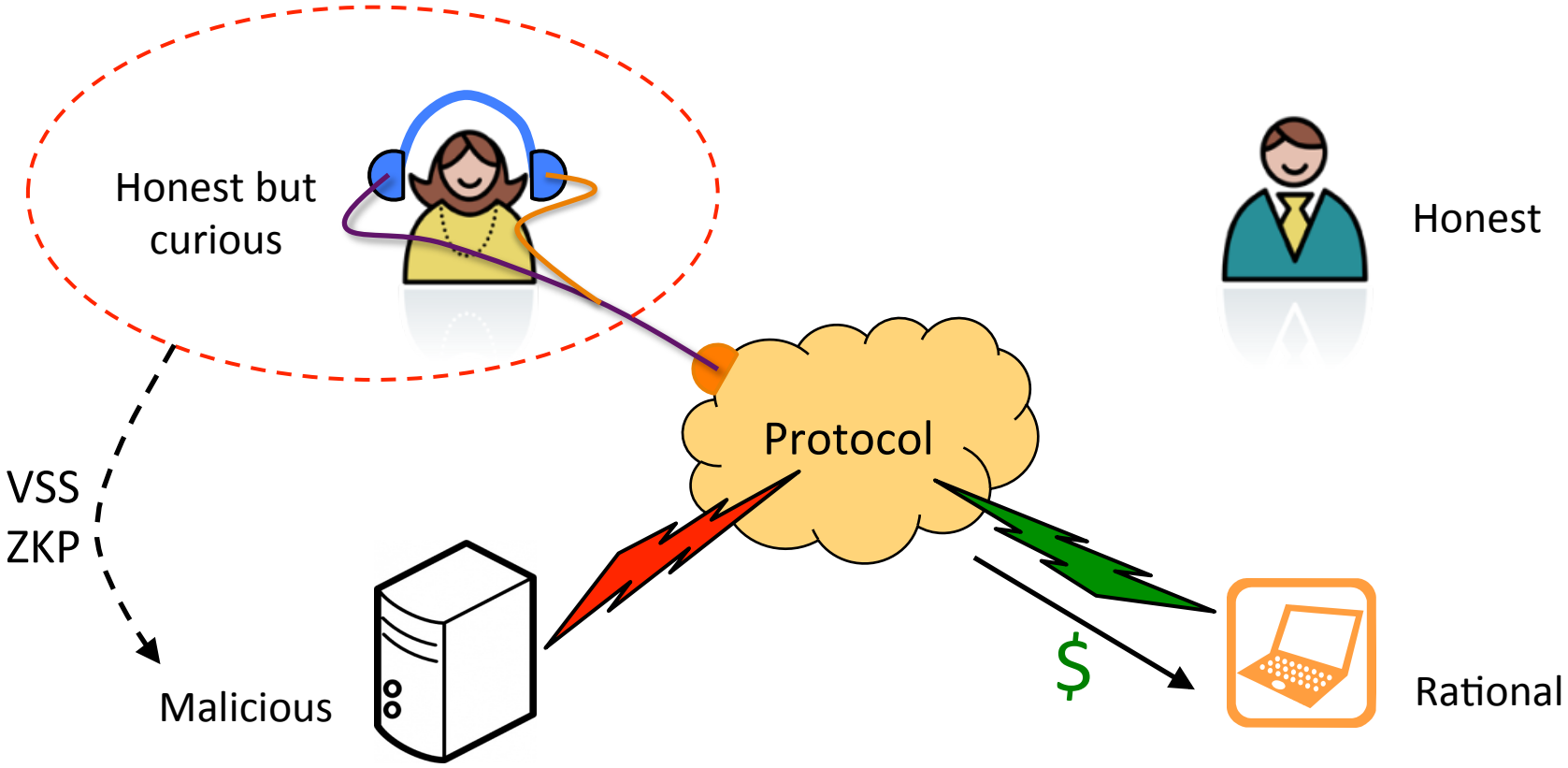
Generally slower and results in ciphertext expansion.

Applications: PKI, hybrid cryptosystems, digital signatures

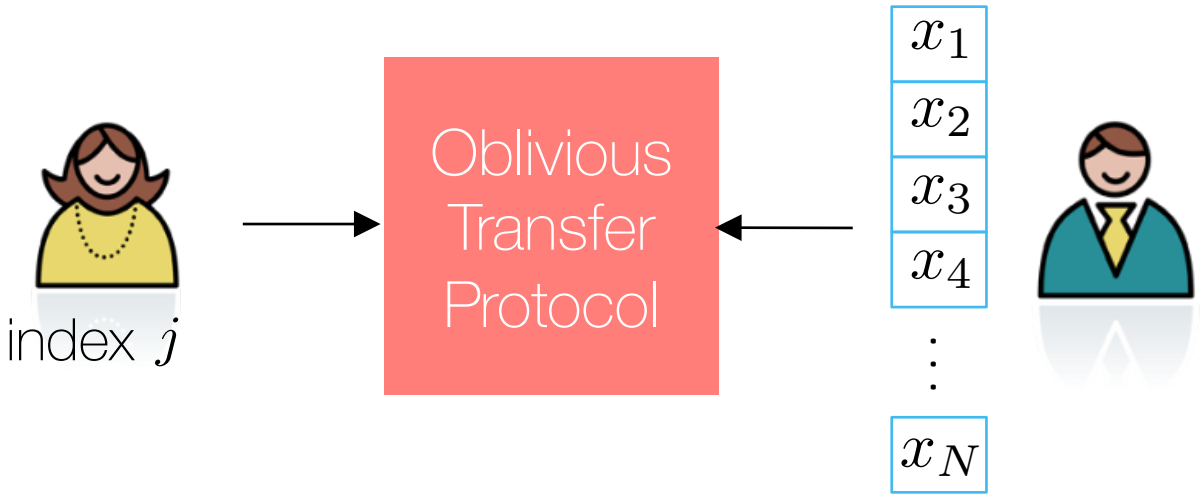
Example: RSA. Typical key sizes: 1024, 2048, 4096, bits. [Rivest, Shamir, Adelson, 1978]

What operations can you do in the ciphertext domain with RSA?

Types of Participants



Primitive Protocol: Oblivious Transfer



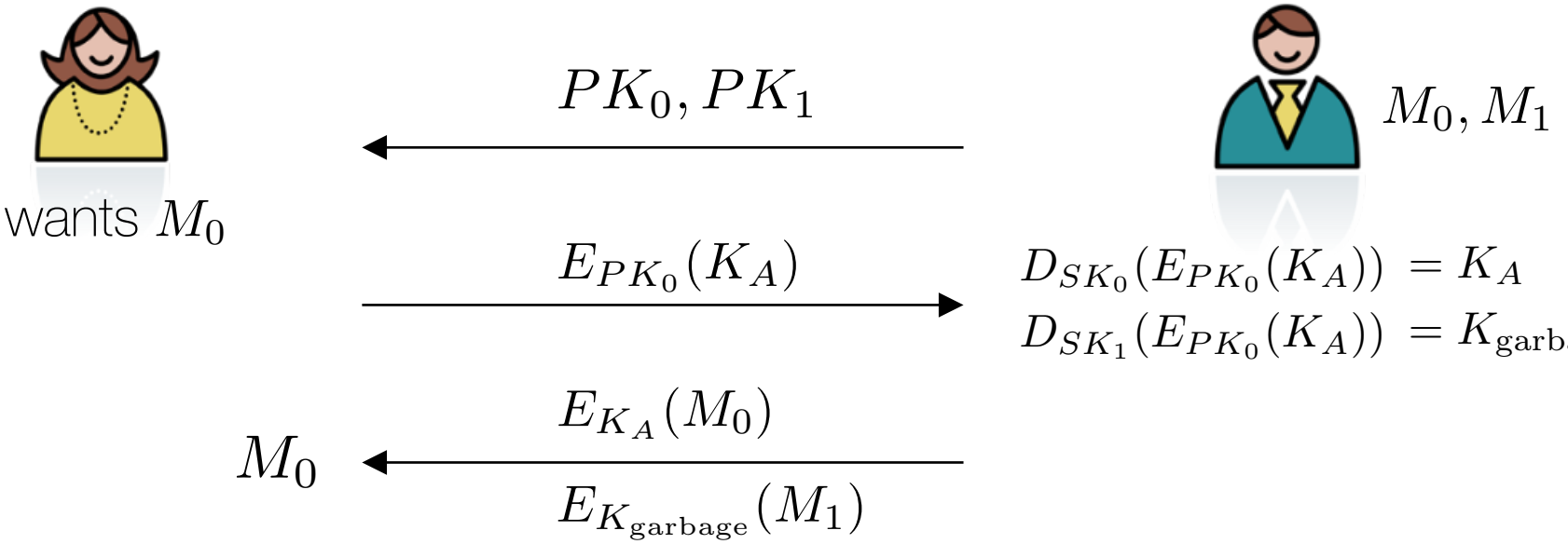
Inputs: Bob has a data vector $\mathbf{x}$, Alice has an index j.

Outputs: Alice discovers x_j, Bob finds out nothing.

Fundamental primitive for all of secure computation. For nearly all privacy-preserving data retrieval, some form of OT is involved.

[Yao, 1982, 1986] [Rabin, 2005][Killian, 1988]

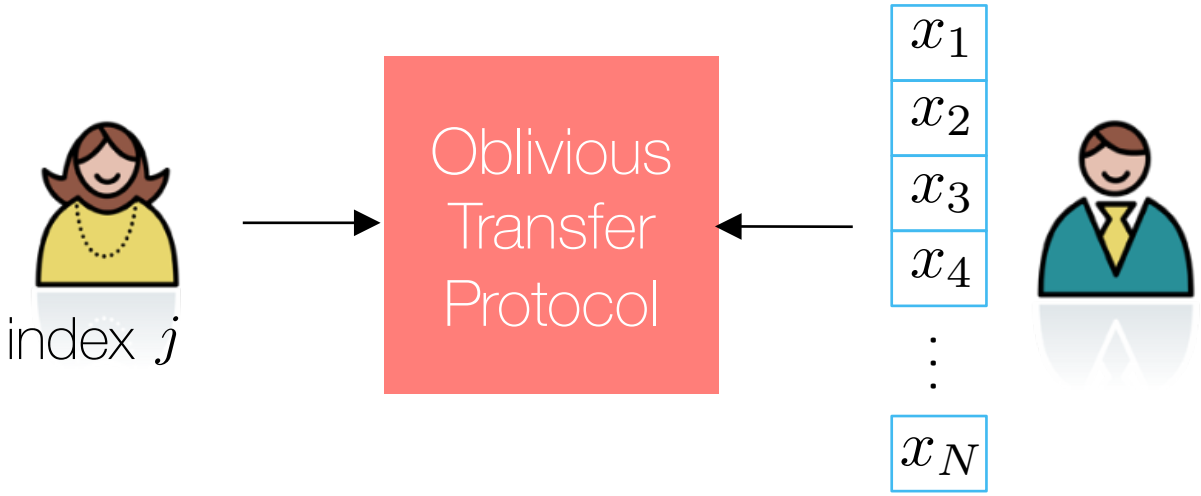
OT using public-key encryption



Alice cannot cheat because she does not know SK_1 hence K_{garbage}.

Bob cannot cheat so long as K_A and K_{garbage} both look random.

OT is a sufficient primitive for SMC

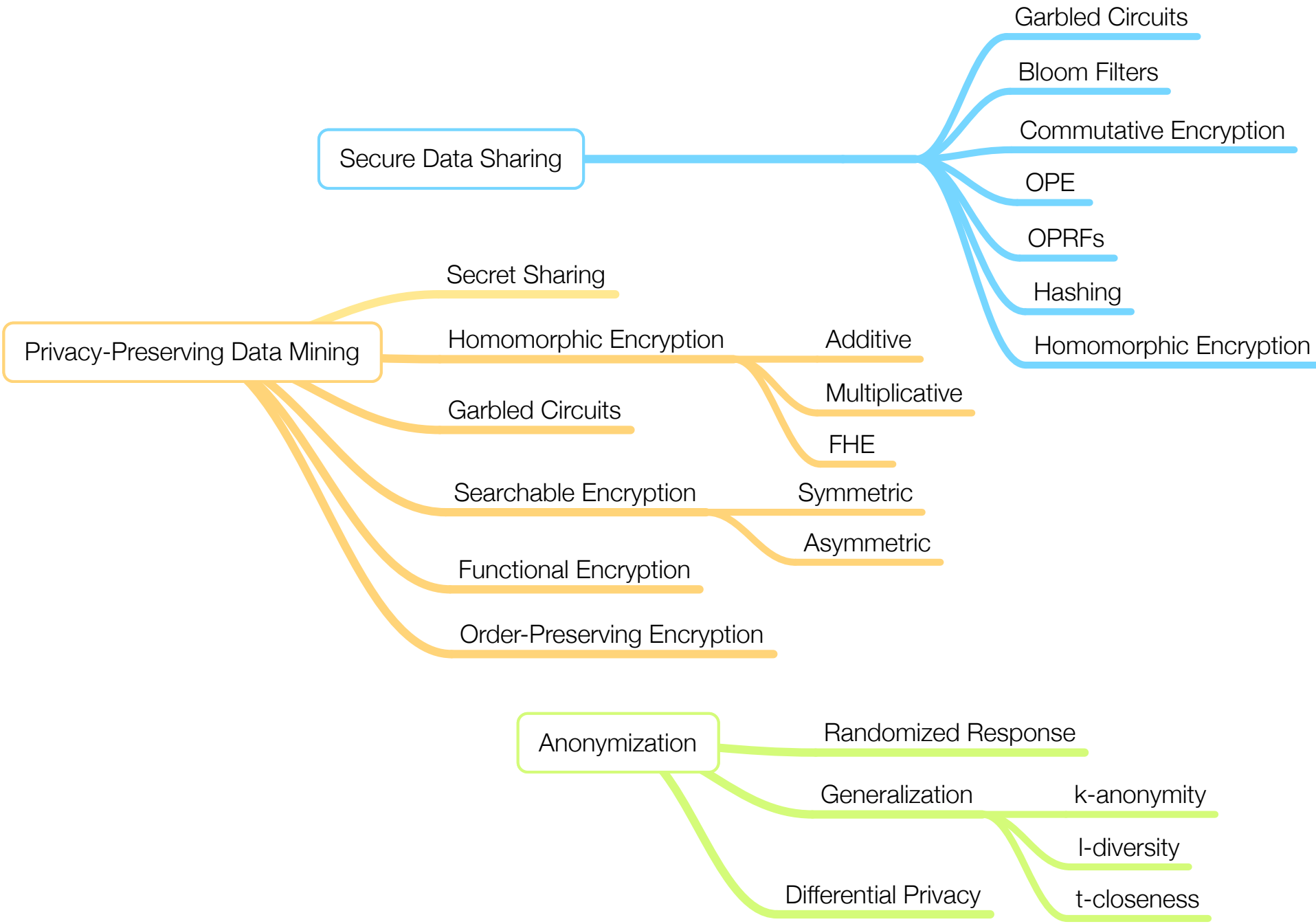


Can extend to 1-out-of-N OT and k-out-of-N OT.

OT has O(N) complexity.

OT with garbled circuits → Securely evaluate any function that can be expressed as a circuit. (More on this later.)

Query privacy AND database privacy. PIR is weaker.


Secure Data Sharing
Garbled Circuits
Bloom Filters
Commutative Encryption
OPE
OPRFs
Hashing
Homomorphic Encryption
Privacy-Preserving Data Mining
Secret Sharing
Homomorphic Encryption
Additive
Multiplicative
FHE
Garbled Circuits
Searchable Encryption
Symmetric
Asymmetric
Functional Encryption
Order-Preserving Encryption
Anonymization
Randomized Response
Generalization
k-anonymity
l-diversity
t-closeness
Differential Privacy

4

Tools

Computational

Information-theoretic

Signal Processing

Statistical

Functions

sum

product

mean

variance

distances

polynomials

correlation

set intersection

set union

set cardinality

histogram

max/min

selection

---

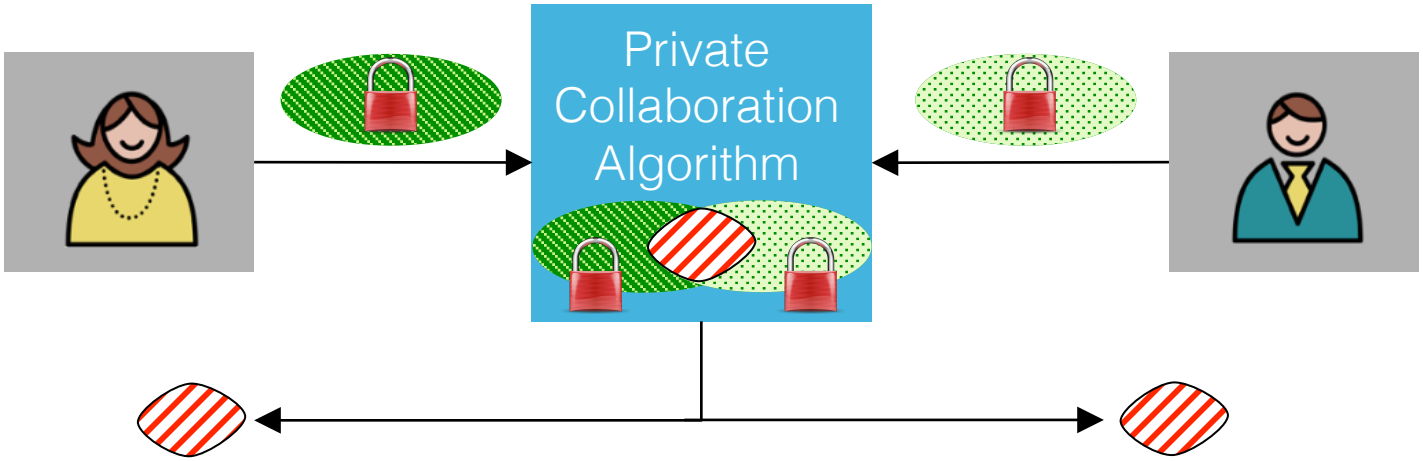
filtering

graph processing

classification

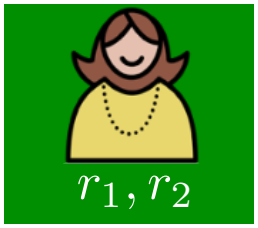
edit distances

Private Set Intersection



- ✓ Can be implemented in many ways with classical cryptographic tools, e.g., Bloom filters, hashing, RSA-style encryption, etc.
- ✓ Can be made secure against malicious participants.
- ✗ Supports a very specific operation, e.g., efficient for PSI, but very inefficient for count queries.
- ✗ Hard to use with noisy data.

2-party PSI Protocols



$(H(a))^{r_1}$

$(H(a))^{s_1}$

$Y^{r_2} \equiv g^{r_2 s_2}$

$J(g^{r_2 s_2}(H(a))^{s_1})$

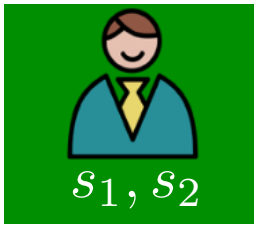
$(H(a))^{r_1}, X = g^{r_2}$

$(H(a))^{r_1 s_1}$

$Y = g^{s_2}$

$J(g^{r_2 s_2}(H(b))^{s_1})$

[De Cristofaro, Gasti, Tsudik 12]



$(H(b))^{s_1}$

$(H(a))^{r_1 s_1}$

$Y = g^{s_2}$

$J(X^{s_2}(H(b))^{s_1})$

Extends to a malicious-secure PSI protocol.

Counting matching elements gives PSI-CA

Exercise: Why not just directly compare hashes?

Variants of PSI and PSI-CA

Commutative encryption [Agrawal, Evfiviemski, Srikant, 03]

Oblivious Polynomial Evaluation (OPE) [Freedman, Nissim, Pinkas 05]

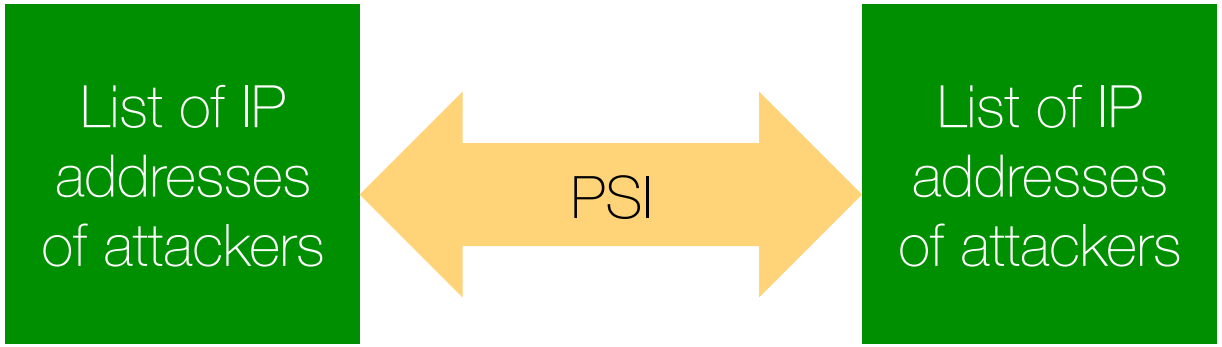
Paillier encryption [De Cristofaro, Tsudik 10]

Oblivious Pseudorandom functions (OPRFs) [Hazay, Lindell 08]

Bloom Filters [Kerschbaum 12]

Garbled Circuits [Huang, Evans, Katz, 12]

Example: Cyber-threat mitigation

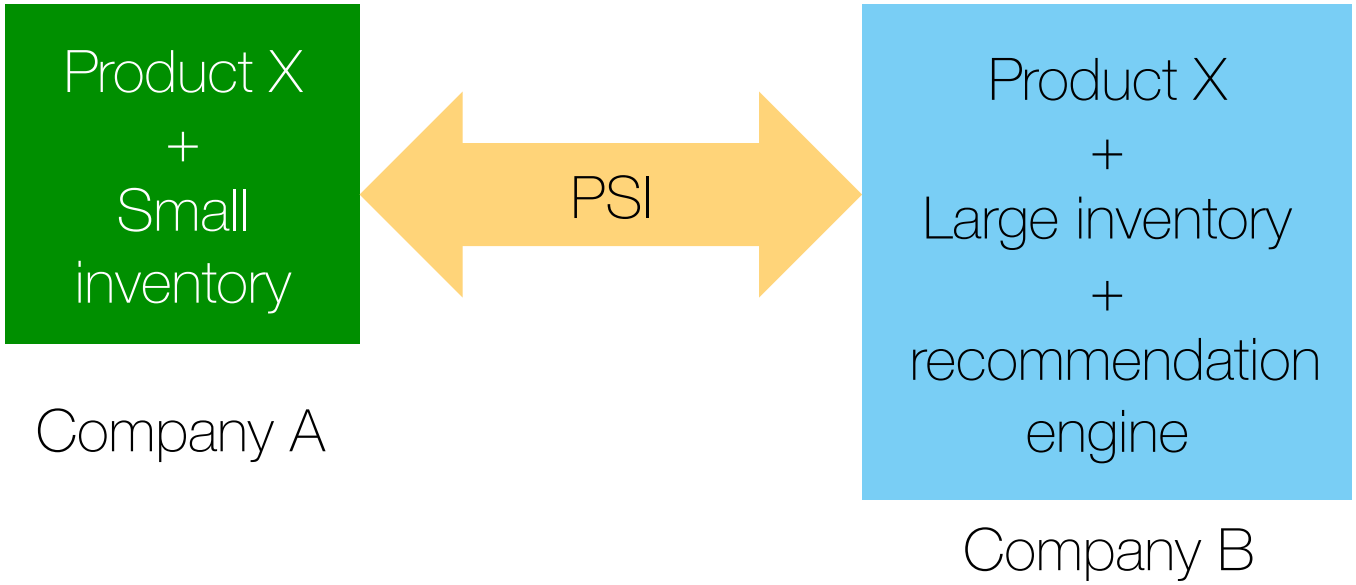


Companies A and B will collaborate to develop a cyber-threat mitigation strategy if they have the many common attackers, otherwise not.

Perform PSI on IP addresses of suspected attackers.

PSI helps in privacy-preserving risk-benefit analysis.

Example: Data monetization



Company A wants to find out: If a customer is interested in product X, what other products should be advertised to her?

Company B (e.g., Amazon) has this information.

PSI reveals only the identity of product to Company B, but not any other elements of Company B's inventory.

Homomorphic Cryptosystems

Additive

[Paillier 99, Damgard-Jurik 01]

$$E(x)E(y) \equiv E(x+y)$$

Multiplicative

[El Gamal 85]

$$E(x)E(y) \equiv E(xy)$$

2-DNF homomorphic

[Boneh, Goh, Nissim 05]

$$e(E(x), E(y)) \equiv F(xy)$$
$$F(xy+uv) \equiv F(xy)F(uv)$$

Fully homomorphic

[Gentry, 09]

[van Dijk, Gentry, Halevi, Vaikunthanathan 10]

[Brakerski, Vaikunthanathan 10]

$$E(x+y) \equiv E(x) + E(y)$$
$$E(x)E(y) \equiv E(xy)$$

Paillier Encryption [Paillier, 1999]

private key

$p, q \longrightarrow N = pq$

$$\mathbb{Z}_{N^2}^* \subset \mathbb{Z}_{N^2} = \{0, 1, ..., N^2 - 1\}$$

containing elements that have a multiplicative inverse.

Choose $g \in \mathbb{Z}_{N^2}^*, r \in \mathbb{Z}_N^* \longrightarrow N, g$ public key

$$E_r(m) = g^m r^N \mod N^2$$

r makes the encryption semantically secure, i.e., every encryption of m will look different, if r is chosen differently.

Exercise: Why is this important in analytics?

Paillier Decryption [Paillier, 1999]

private key

$p, q \longrightarrow N = pq \quad \lambda = \mathrm{LCM}(p-1, q-1)$

$$D(c) = \frac{L(c^{\lambda} \mod N^2)}{L(g^{\lambda} \mod N^2)} = m$$

where $L(x) = \frac{x-1}{N}$

Note: The decryption function does not need to know r.

Additive Homomorphic Property

$$\left(g^{m_1} r_1^N \mod N^2\right)\left(g^{m_2} r_2^N \mod N^2\right)$$
$$= g^{m_1+m_2}(r_1 r_2)^N \mod N^2$$

$$E_{r_1}(m_1)E_{r_2}(m_2) = E_{r'}(m_1+m_2)$$

$$\left(g^{m_1} r^N \mod N^2\right)^{m_2} \mod N^2$$
$$= g^{m_1 m_2} r^N \mod N^2$$

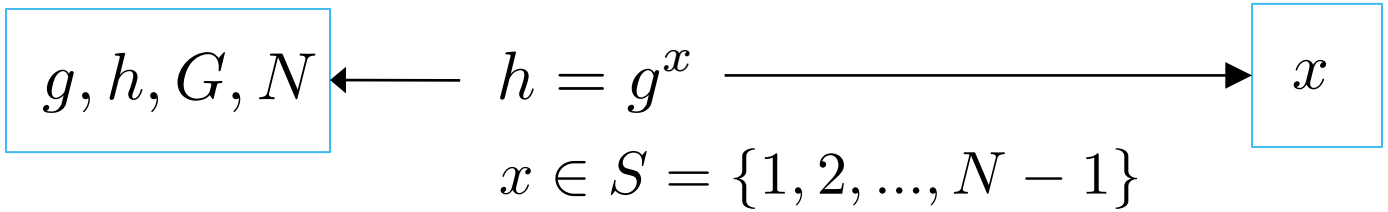
$$\left(E_r(m_1)\right)^{m_2} = E_r(m_1 m_2)$$

El Gamal Encryption [El Gamal 85]

Let g be the generator of a cyclic group G of order N.

public key

private key



$x \in S = \{1, 2, ..., N-1\}$

To encrypt $m \in G$ choose a $y \in S$ and compute:

$$(c_1, c_2) = (g^y, mh^y) = (g^y, mg^{xy})$$

y plays the role of the semantic security parameter. Every encryption of m will look different if y is chosen differently each time.

El Gamal Decryption [El Gamal 85]

public key | private key

g, h, G, N | x

Compute $c_1^x = g^{xy} \in G$

Then $c_2 \left(g^{xy}\right)^{-1} = m g^{xy} \left(g^{xy}\right)^{-1} = m$

Note: The decryption function does not need to know y.

Multiplicative Homomorphic Property

$$E(m_1) = (g^{y_1}, m_1 g^{xy_1})$$

$$E(m_2) = (g^{y_2}, m_2 g^{xy_2})$$

$$E(m_1)E(m_2) = (g^{y_1} g^{y_2}, m_1 m_2 g^{xy_1} g^{xy_2})$$

$$= (g^{y_1+y_2}, m_1 m_2 g^{x(y_1+y_2)})$$

$$= E(m_1 m_2)$$

More precisely

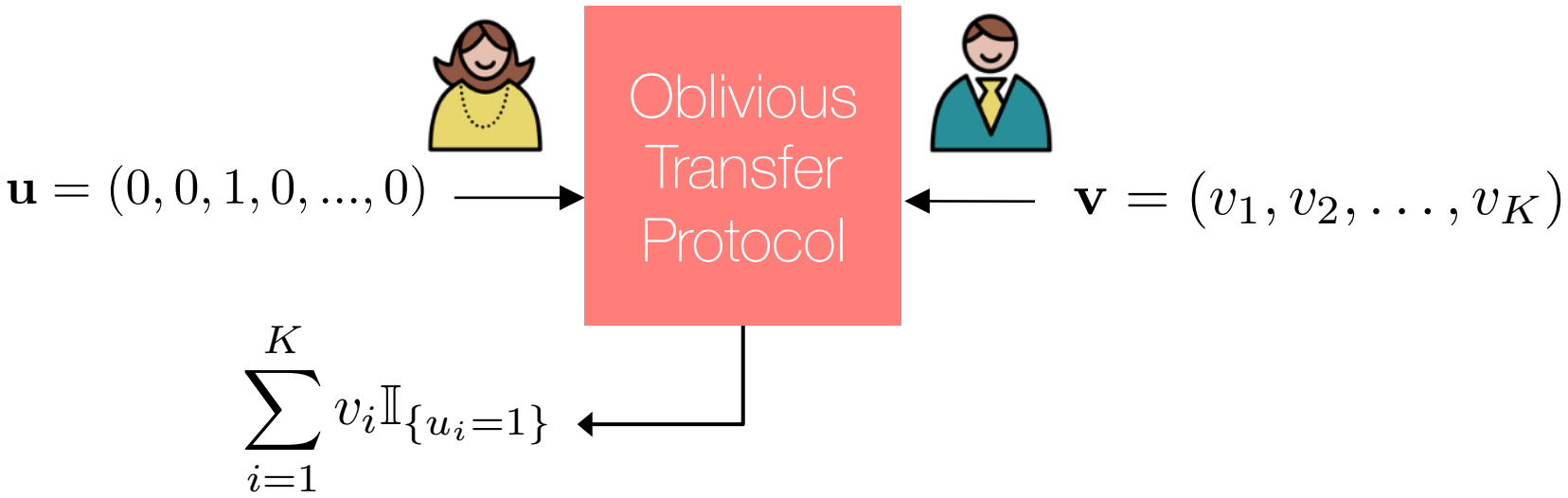
$$E_{y_1}(m_1)E_{y_2}(m_2) = E_{y_1+y_2}(m_1 m_2)$$

Examples

Use homomorphic encryption to implement:

1. 1-out-of-K Oblivious Transfer
2. Select some elements of a vector and sum them
3. Secure dot product
4. Secure Euclidean distance
5. Secure Weighted Euclidean distance (exercise: easy)
6. Secure Hamming distance (exercise: easy)
7. Secure Edit distance (exercise: not so easy)

E.g.,1-out-of-*K* Oblivious Transfer

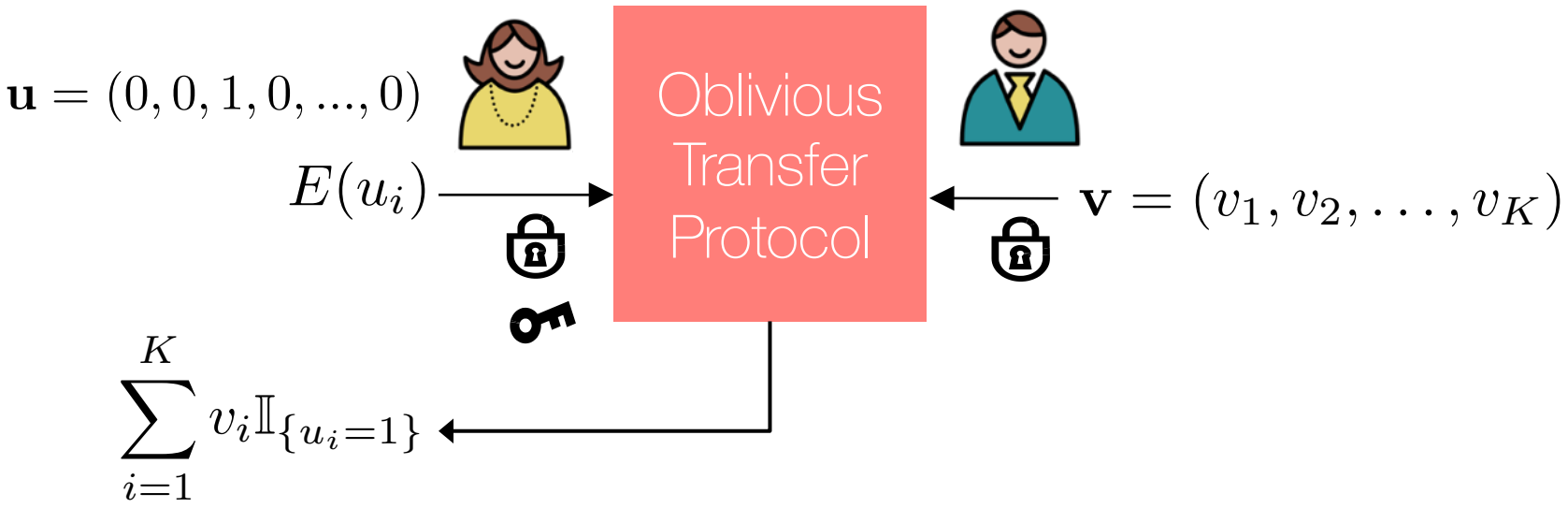


Goal: Alice has a binary selector $\mathbf{u}$, Bob has data vector $\mathbf{v}$. Alice should discover the selected element from $\mathbf{v}$.

Query Privacy: Bob should not find the selector vector.

Data Privacy: Alice should not discover any information other than the selected element.

1-out-of-*K* Oblivious Transfer (contd)



1. Alice sends element-wise encryptions of $\mathbf{u}$ to Bob.
2. Bob computes the dot product of $\mathbf{u}$ and $\mathbf{v}$ using additive homomorphic property, and sends it to Alice.

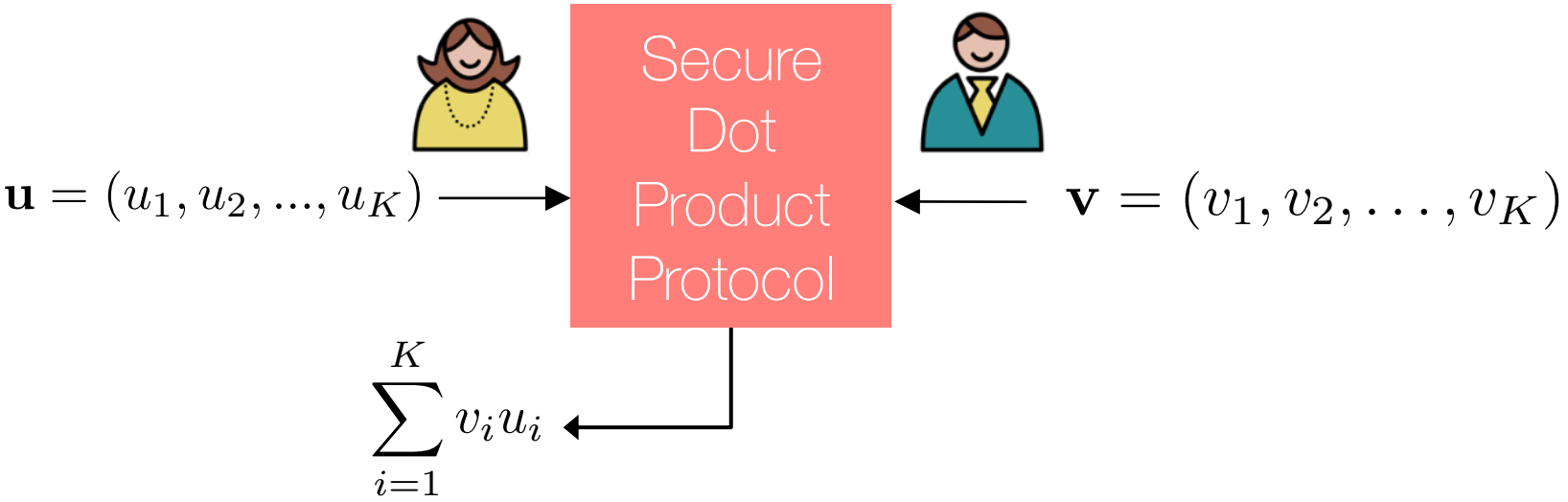
$$\prod_{i=1}^{K} E(u_i)^{v_i} = \prod_{i=1}^{K} E(u_i v_i) = E\left(\sum_{i=1}^{K} v_i u_i\right) = E\left(\sum_{i=1}^{K} v_i \mathbb{I}_{\{u_i\}}\right)$$

3. Alice decrypts the dot product.

Oblivious Transfer Complexity

		
# Encryptions	K	0
# Decryptions	1	0
# Multiplications	0	K
# Exponentiations	0	K
# Transmissions	K	1

Secure Dot Product Setup

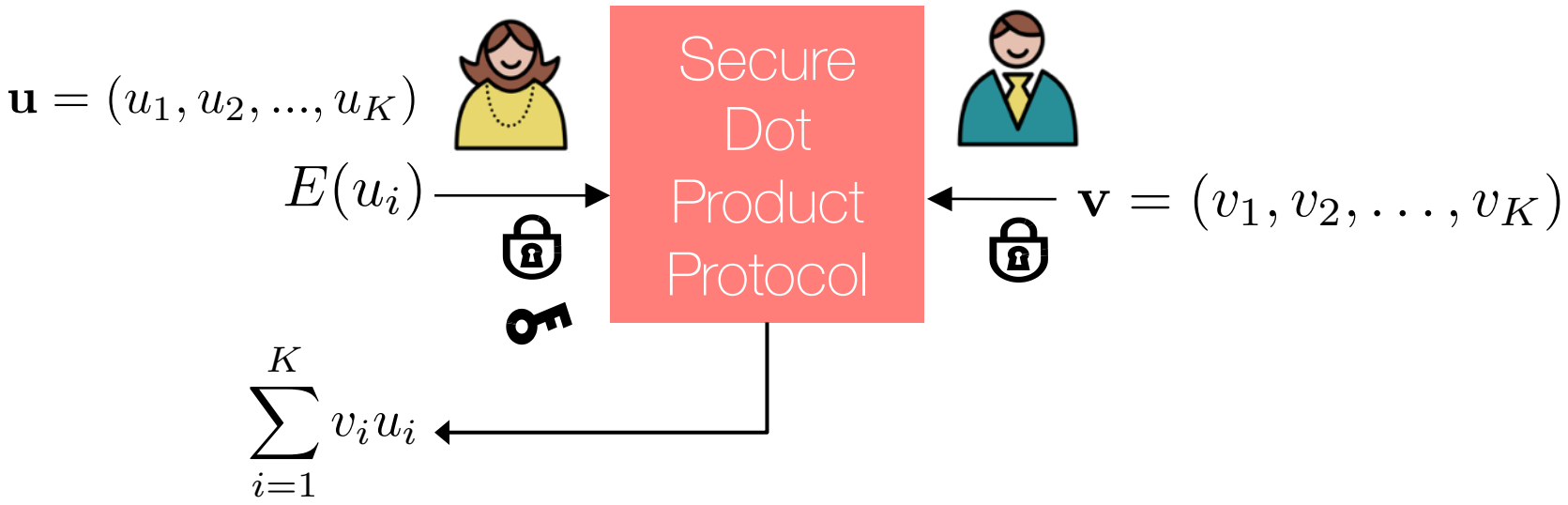


Goal: Alice has a data vector $\mathbf{u}$, Bob has data vector $\mathbf{v}$. Alice should discover the dot product of $\mathbf{u}$ and $\mathbf{v}$.

Query Privacy: Bob should not discover any element of $\mathbf{u}$.

Data Privacy: Alice should not discover any information other than the dot product.

Secure Dot Product Protocol



1. Alice sends element-wise encryptions of $\mathbf{u}$ to Bob.
2. Bob computes the dot product of $\mathbf{u}$ and $\mathbf{v}$ using additive homomorphic property, and sends it to Alice.

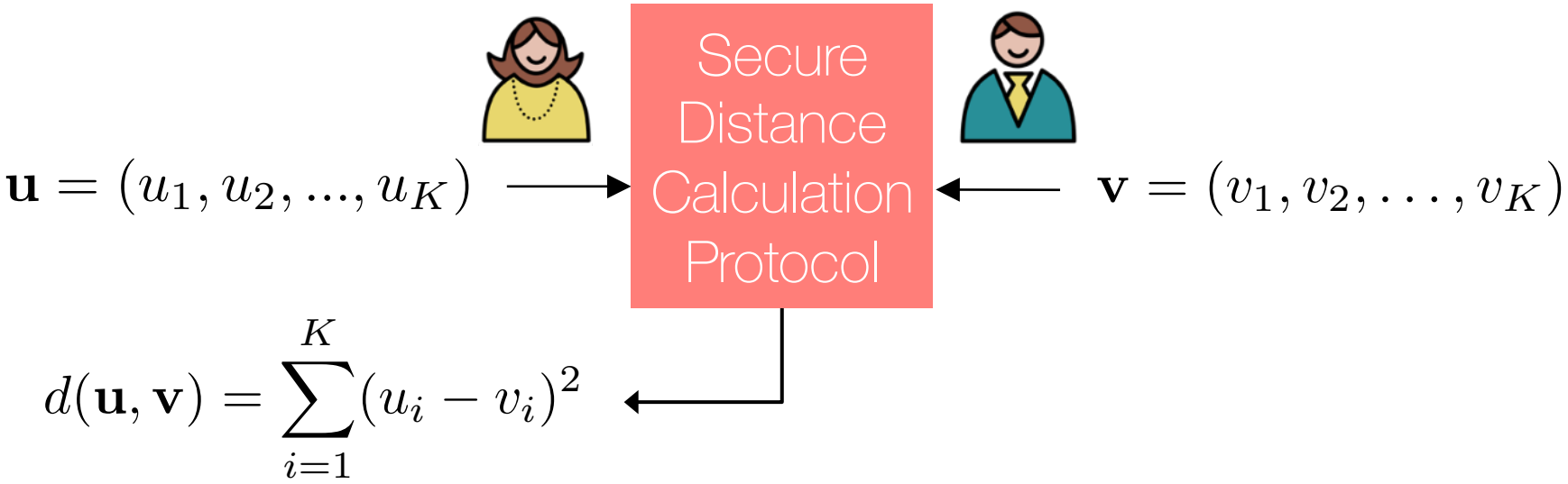
$$\prod_{i=1}^{K} E(u_i)^{v_i} = \prod_{i=1}^{K} E(u_i v_i) = E\left(\sum_{i=1}^{K} v_i u_i\right)$$

3. Alice decrypts the dot product.

Secure Dot Product Complexity

		
# Encryptions	K	0
# Decryptions	1	0
# Multiplications	0	K
# Exponentiations	0	K
# Transmissions	K	1

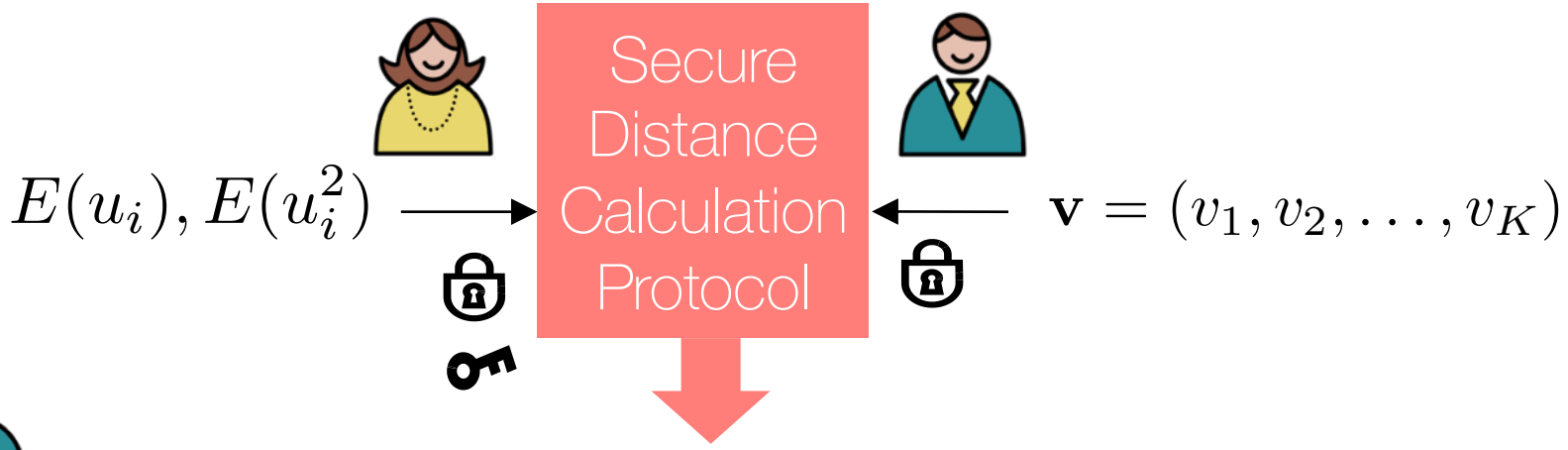
E.g., Secure Euclidean Distance



Alice and Bob should not discover each other's data.

Only Alice should know the distance.

Secure Euclidean Distance (contd)



$$\left(\prod_{i=1}^{K} E(u_i^2)\right) E\left(\sum_{i=1}^{K} v_i^2\right) \left(\prod_{i=1}^{K} E(u_i)^{-2v_i}\right) =$$

$$E\left(\sum_{i=1}^{K} u_i^2\right) E\left(\sum_{i=1}^{K} v_i^2\right) E\left(\sum_{i=1}^{K} -2u_i v_i\right) = E\left(d(\mathbf{u}, \mathbf{v})\right)$$

Euclidean Distance Complexity

		
# Encryptions	$2K$	0
# Decryptions	1	0
# Multiplications	0	$2K + 2$
# Exponentiations	0	K
# Transmissions	$2K$	1

Distance Computation with Privacy

Approximate Euclidean Distance

Approximate Hamming Distance [Boufounos, Rane, 2012]

Edit (Levenshtein) Distance [Atallah, Kerschbaum, Du, 2003][Li, Atallah, 2005][Sun, Rane, Vetro, 2011]

Applications:

- Privacy-preserving Recommender Systems [Erkin et al, 07]
- Privacy-preserving biometric authentication [Blanton, Gasti, 2011]
- Database querying with privacy

Homomorphic Cryptosystems

Enables outsourced cloud computing for rich variety of functions.

Some formulations, e.g., Ring Learning With Errors, are resistant to quantum computing attacks.

Memory access patterns reveal information about data elements. (cf. ORAM)

Most schemes were developed for semi-honest parties. For malicious parties, use ZKP, but this increases complexity.

Data is growing faster than computational power. Moore's law won't save us from the complexity of FHE.

References (Review Topics)

Pub, NIST FIPS. "197: Advanced encryption standard (AES)." *Federal Information Processing Standards Publication* 197 (2001): 441-0311.

Daemen, Joan, and Vincent Rijmen. *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 2013.

Rivest, Ronald L., Adi Shamir, and Len Adleman. "A method for obtaining digital signatures and public-key cryptosystems." *Communications of the ACM* 21.2 (1978): 120-126.

Yao, Andrew. "How to generate and exchange secrets." *Foundations of Computer Science, 1986., 27th Annual Symposium on*. IEEE, 1986.

Yao, Andrew Chi-Chih. "Protocols for secure computations." *FOCS*. Vol. 82. 1982.

Rabin, Michael O. "How To Exchange Secrets with Oblivious Transfer." *IACR Cryptology ePrint Archive* 2005 (2005): 187.

Kilian, Joe. "Founding crytpography on oblivious transfer." *Proceedings of the twentieth annual ACM symposium on Theory of computing*. ACM, 1988.

De Cristofaro, Emiliano, and Gene Tsudik. "Practical private set intersection protocols with linear complexity." *Financial Cryptography and Data Security*. Springer Berlin Heidelberg, 2010. 143-159.

References (Private Set Intersection)

Agrawal, Rakesh, Alexandre Evfimievski, and Ramakrishnan Srikant. "Information sharing across private databases." *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*. ACM, 2003.

Hazay, Carmit, and Yehuda Lindell. "Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries." *Theory of Cryptography*. Springer Berlin Heidelberg, 2008. 155-175.

Kerschbaum, Florian. "Outsourced private set intersection using homomorphic encryption." *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*. ACM, 2012.

Huang, Yan, David Evans, and Jonathan Katz. "Private set intersection: Are garbled circuits better than custom protocols?." *NDSS*. 2012.

De Cristofaro, Emiliano, Paolo Gasti, and Gene Tsudik. "Fast and private computation of cardinality of set intersection and union." *Cryptology and Network Security*. Springer Berlin Heidelberg, 2012. 218-231.

Freedman, Michael J., Kobbi Nissim, and Benny Pinkas. "Efficient private matching and set intersection." *Advances in Cryptology-EUROCRYPT 2004*. Springer Berlin Heidelberg, 2004.

References (Homomorphic Encryption)

Paillier, Pascal. "Public-key cryptosystems based on composite degree residuosity classes." *Advances in cryptology—EUROCRYPT'99*. Springer Berlin Heidelberg, 1999.

ElGamal, Taher. "A public key cryptosystem and a signature scheme based on discrete logarithms." *Advances in cryptology*. Springer Berlin Heidelberg, 1985.

Damgård, Ivan, Mads Jurik, and Jesper Buus Nielsen. "A generalization of Paillier's public-key system with applications to electronic voting." *International Journal of Information Security* 9.6 (2010): 371-385.

Boneh, Dan, Eu-Jin Goh, and Kobbi Nissim. "Evaluating 2-DNF formulas on ciphertexts." *Theory of cryptography*. Springer Berlin Heidelberg, 2005. 325-341.

Gentry, Craig. "Fully homomorphic encryption using ideal lattices." *STOC*. Vol. 9. 2009.

Van Dijk, Marten, et al. "Fully homomorphic encryption over the integers." *Advances in cryptology–EUROCRYPT 2010*. Springer Berlin Heidelberg, 2010. 24-43.

Brakerski, Zvika, and Vinod Vaikuntanathan. "Efficient fully homomorphic encryption from (standard) LWE." *SIAM Journal on Computing* 43.2 (2014): 831-871.

References: HE Applications

Rane, Shantanu, and Wei Sun. "Privacy preserving string comparisons based on Levenshtein distance." *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*. IEEE, 2010.

Rane, Shantanu, and Petros T. Boufounos. "Privacy-preserving nearest neighbor methods: comparing signals without revealing them." *Signal Processing Magazine, IEEE* 30.2 (2013): 18-28.

Atallah, Mikhail J., Florian Kerschbaum, and Wenliang Du. "Secure and private sequence comparisons." *Proceedings of the 2003 ACM workshop on Privacy in the electronic society*. ACM, 2003.

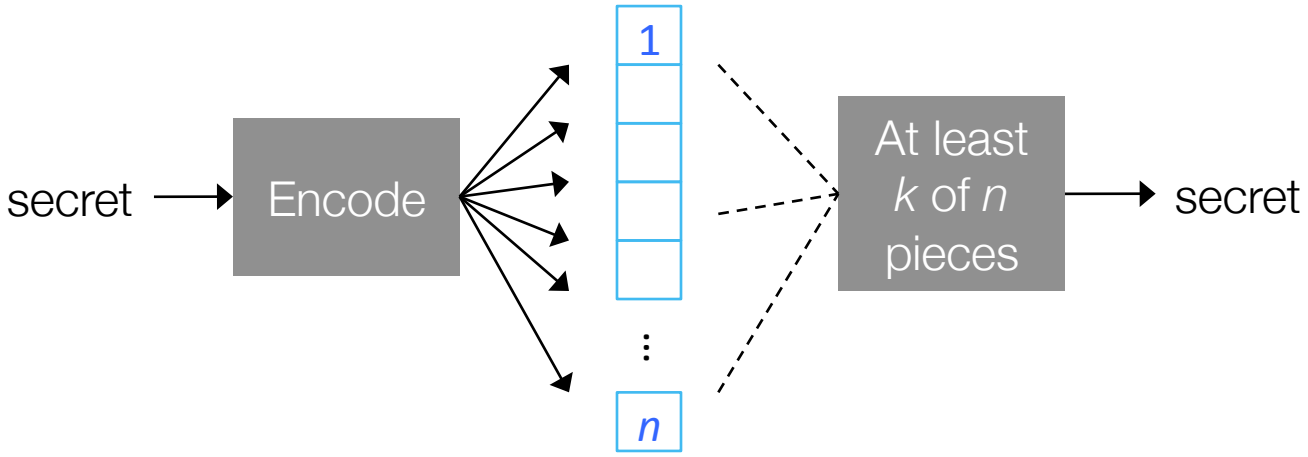
Atallah, Mikhail J., and Jiangtao Li. "Secure outsourcing of sequence comparisons." *International Journal of Information Security* 4.4 (2005): 277-287.

Erkin, Z., Piva, A., Katzenbeisser, S., Lagendijk, R. L., Shokrollahi, J., Neven, G., & Barni, M. (2007). Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing. *EURASIP Journal on Information Security*, *2007*, 17.

Blanton, Marina, and Paolo Gasti. "Secure and efficient protocols for iris and fingerprint identification." *Computer Security–ESORICS 2011*. Springer Berlin Heidelberg, 2011. 190-209.

Troncoso-Pastoriza, Juan Ramón, Stefan Katzenbeisser, and Mehmet Celik. "Privacy preserving error resilient DNA searching through oblivious automata." *Proceedings of the 14th ACM conference on Computer and communications security*. ACM, 2007.

Secret Sharing



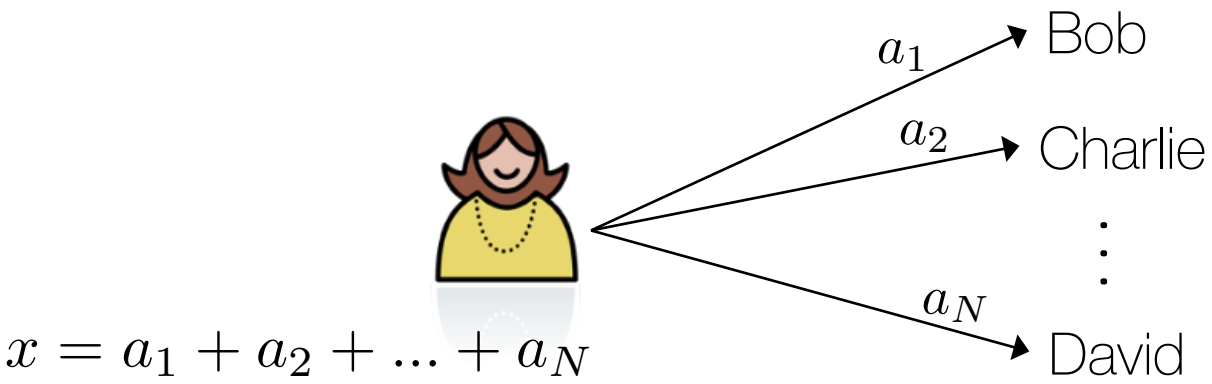
Can be achieved using error correcting codes. [Shamir, 1979]

✓ At the heart of information-theoretically secure multiparty computation. [BGW,1988][CCD,1988]. Each party computes functions of shares, which are combined to obtain a function of the secret.

✓ Computationally efficient. Tolerates $< n/3$ cheaters for arbitrary functions.

✗ Must keep track of inter-participant communications. Not much is known for computation with $n=3$ parties! [Wang, Ishwar, Rane, 2014]

Additive Secret Sharing



Information-theoretically secure, i.e., adversaries have no choice but to guess Alice's inputs by brute force.

All n players must collude to guess x.

Trivially additively homomorphic, that is:

$$x = \sum_{i=1}^{N} a_i, y = \sum_{i=1}^{N} b_i \Rightarrow x + y = \sum_{i=1}^{N} a_i + b_i$$

Shamir Secret Sharing [Shamir 1979]

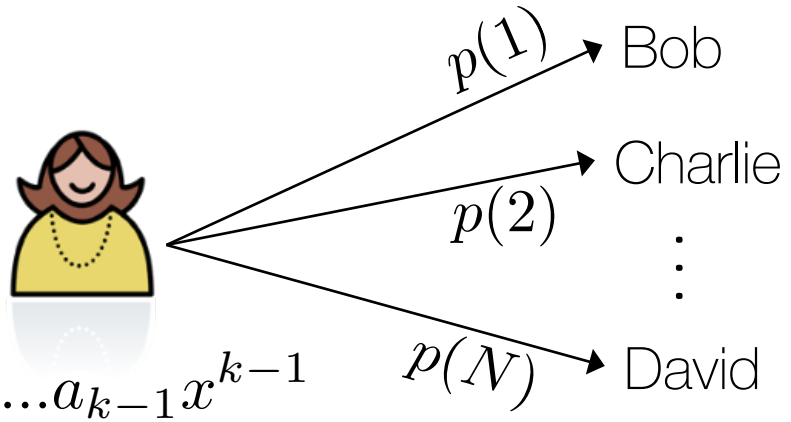
Choose prime p

Secret = $x_a < p$

Choose $a_1, ..., a_{k-1} < p$

Construct a polynomial

$$p(x) = x_a + a_1 x + a_2 x^2 + ... a_{k-1} x^{k-1} \mod p$$



Alice creates a polynomial, whose constant term is the secret, i.e.,

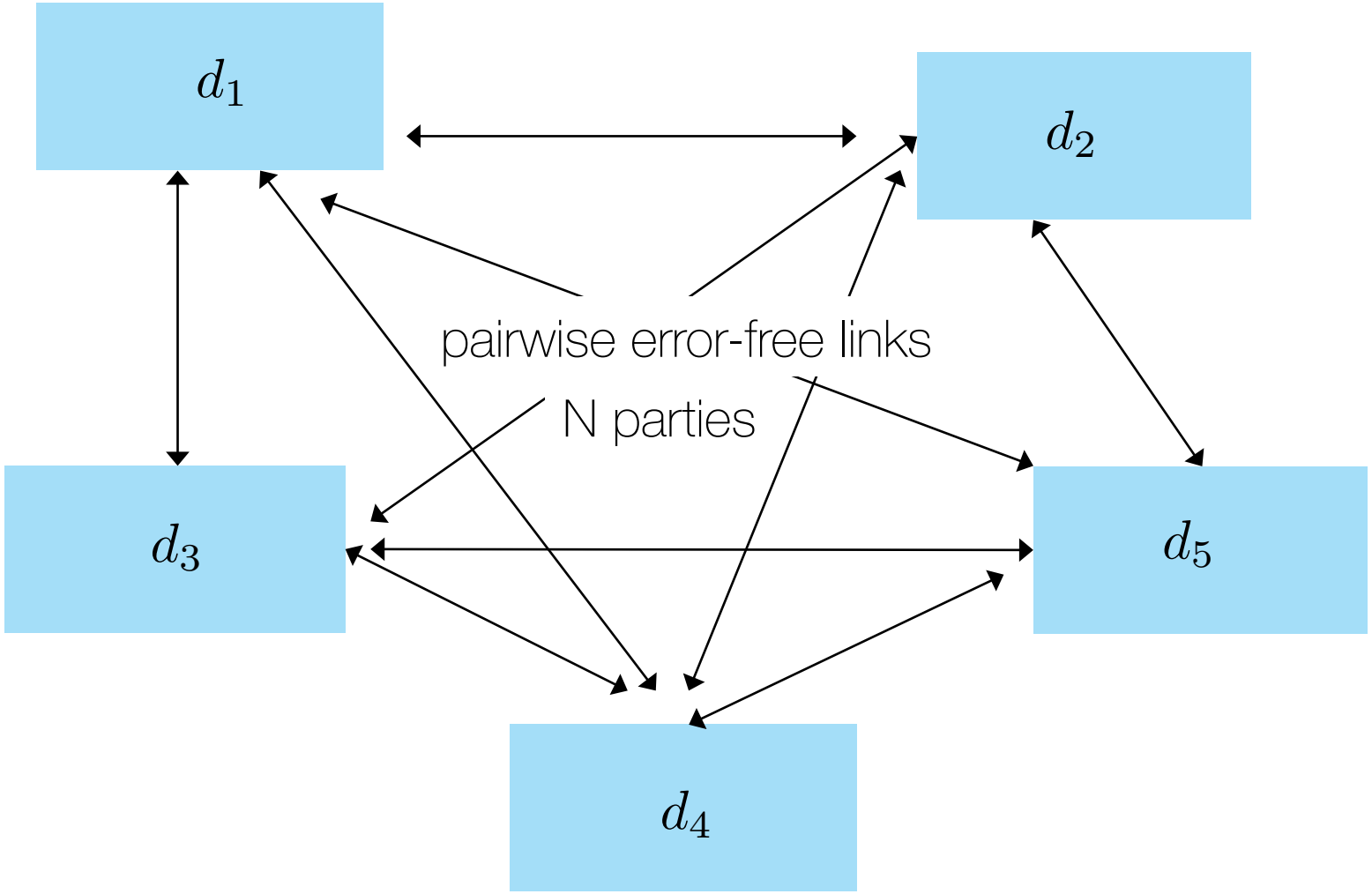
$p(0) = x_a$

Evaluate polynomial at n known points, distribute them as shares.

Need at least k out of N shares to reconstruct the secret.

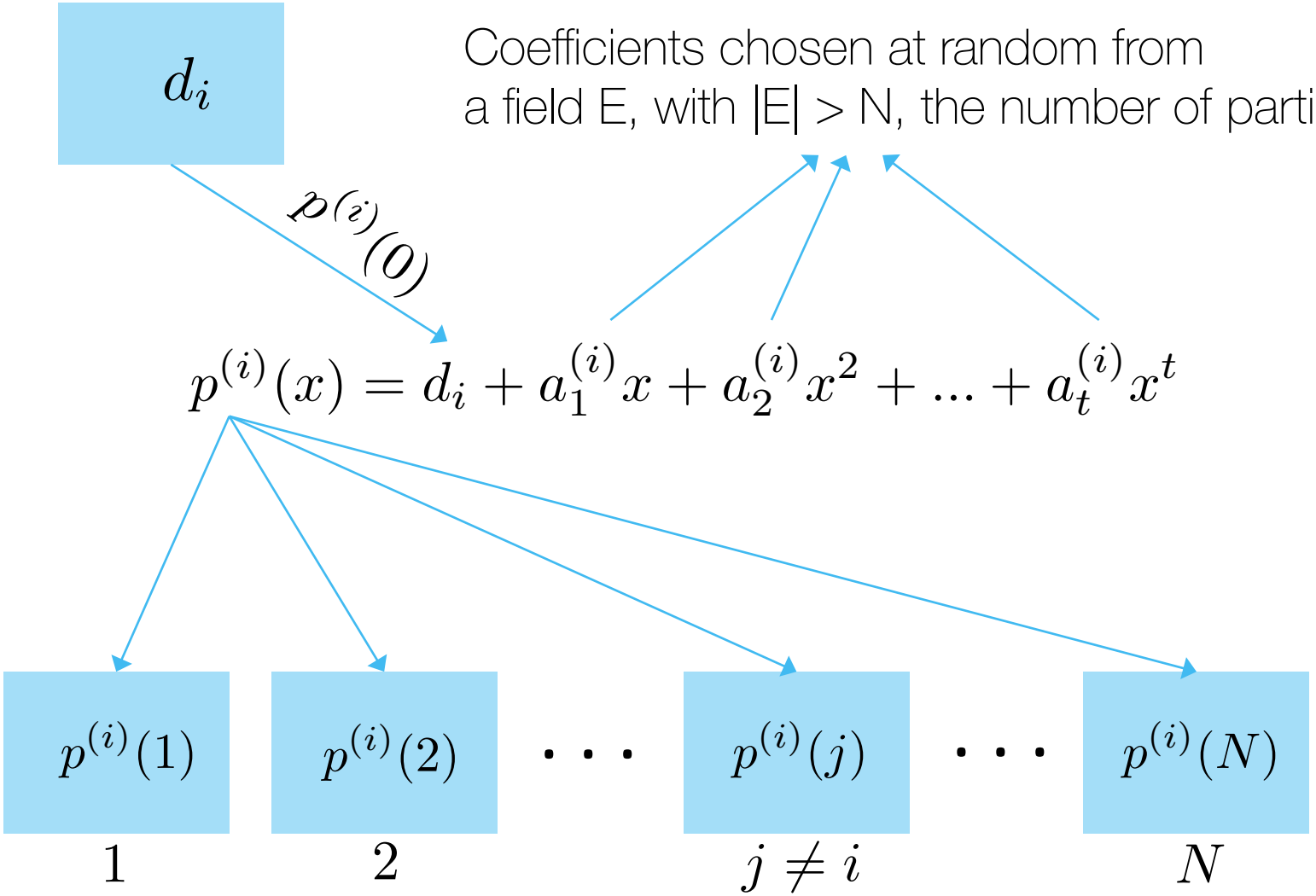
Also, additively homomorphic. (check this as an exercise!)

Secure Multiparty Computation (SMC)

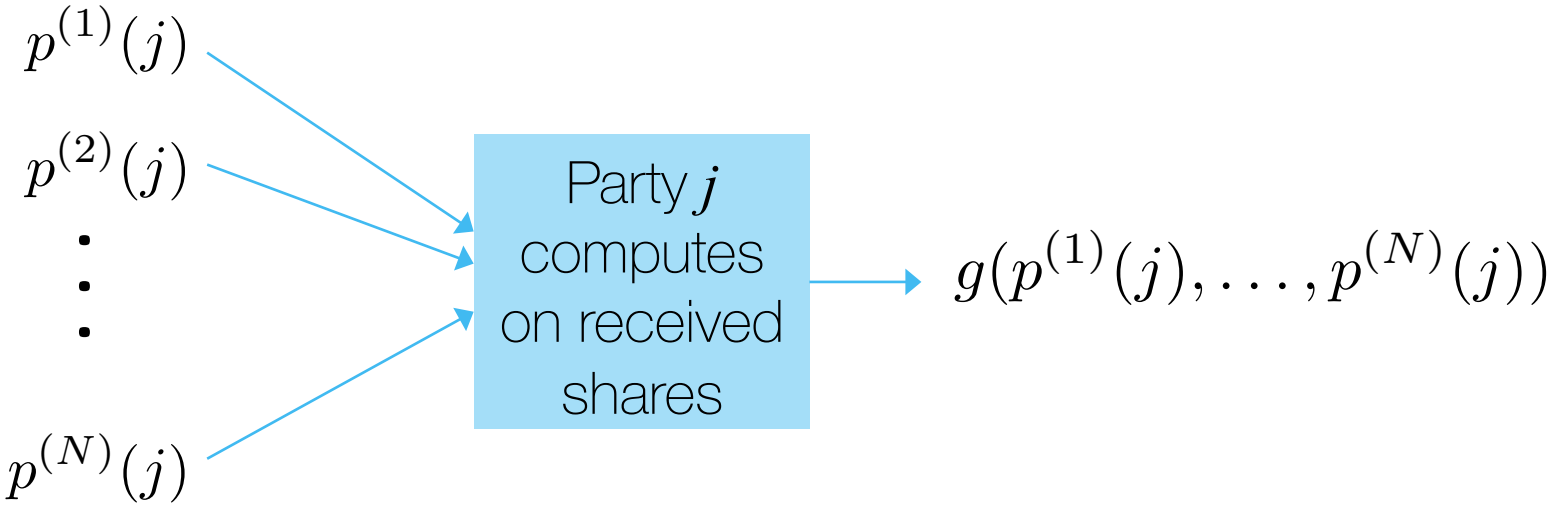


[BGW,1988][CCD,1988]

SMC: Input Phase @ Each Party



SMC: Computation Phase @ Each Party

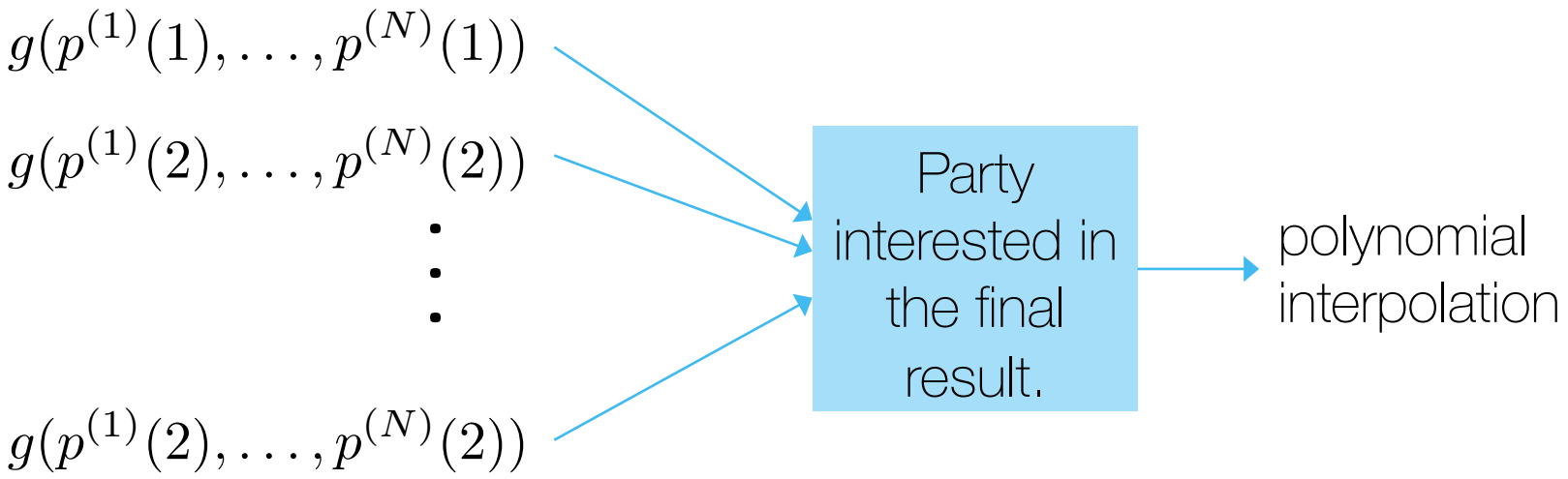


Additions are straightforward, e..g.,

$$g(p^{(1)}(j), \ldots, p^{(N)}(j)) = \sum_{i=1}^{N} p^{(i)}(j)$$

Multiplications may require a degree-reduction step.

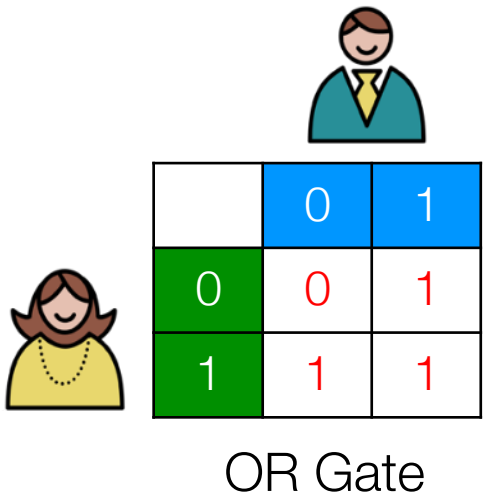
SMC: Reconciliation Phase



Party interested in the final function has access to a polynomial of degree t, whose constant term contains the desired function result.

If it receives evaluations of the polynomial at t or more points, it can compute the result.

Garbled Circuits & Oblivious Transfer



	0	1
0	0	1
1	1	1

OR Gate

Garble

	m_0	m_1
k_0	$E(k_0 \| m_0, 0)$	$E(k_0 \| m_1, 1)$
k_1	$E(k_1 \| m_0, 1)$	$E(k_1 \| m_1, 1)$

[Ex from Prabhakaran's Crypto Notes, 14]

Alice produces garbled circuit for function *f*

Alice provides a key corresponding to her input to Bob

Bob obtains his keys from Alice via 1-of-2 OT

Bob evaluates circuit by decryption using his and Alice's keys

Implementations: Fairplay [Malkhi, Nisan, Pinkas, Sella, 04]

GCs: Advantages and Limitations

General primitive for secure computation. [Yao, 86]

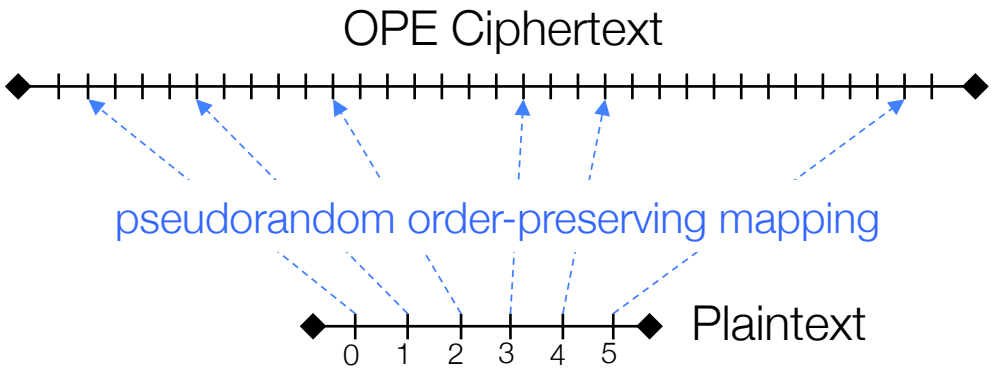
Speed-up: Free XORs, row reduction [Pinkas, Schneider, Smart, Williams 09] [Kolesnikov, Schneider 08].

Very impressive recent results on Levenshtein distance, Hamming distance, AES. [Huang, Evans, Katz, Malka, 11].

Circuits can be extremely complex for data-mining tasks such as classification, clustering, etc., especially with > 2 parties.

Circuit design and garbling requires in-house expertise.

Order-Preserving Encryption



Weaker cryptographic technique where ciphertexts preserve order

- Need knowledge about data values [Agarwal, Kiernan, Srikant, Xu, 04]
- One-shot method with hyper-geometric sampling [Boldyreva, Chenette, Lee, O'Neill, 09, 11]

✓ Supports range queries, median finding, and is deployed within cryptDB. [Ala Popa, Redfield, Zeldovich, Balakrishnan, 11, 12, 13]

✗ Ciphertext expansion can be prohibitive.

Masking

John Smith	32	92043	American	Heart Disease
Kei Takamura	34	92043	Japanese	Cancer
Sarah Jones	38	92043	American	Cancer
Cesar Vincent	37	92306	French	Viral Infection

askdhsf	32	92043	American	Heart Disease
lkjljhflgl	34	92043	Japanese	Cancer
rwithgd	38	92043	American	Cancer
vmbnvc	37	92306	French	Viral Infection

Replaces PII with pseudonymous identifiers

Easy and fast. Identify sensitive attributes and hash them.

High utility, as long as only a few attributes are masked.

HIPAA compliant.

✗ Masking does *not* preserve privacy

askdhsf	32	92043	American	Heart Disease
lkjljhflgl	34	92043	Japanese	Cancer
rwithgd	38	92043	American	Cancer
vmbnvc	37	92306	French	Viral Infection

\+

Kei Takamura	92043	Japanese Instructor	

→

askdhsf	32	92043	American	Heart Disease
Kei Takamura	**34**	**92043**	**Japanese**	**Cancer**
rwithgd	38	92043	American	Cancer
vmbnvc	37	92306	French	Viral Infection

MA Governor medical records [Sweeney 02]

NYT re-identification of AOL Search Data [Barbaro, Zeller, 06]

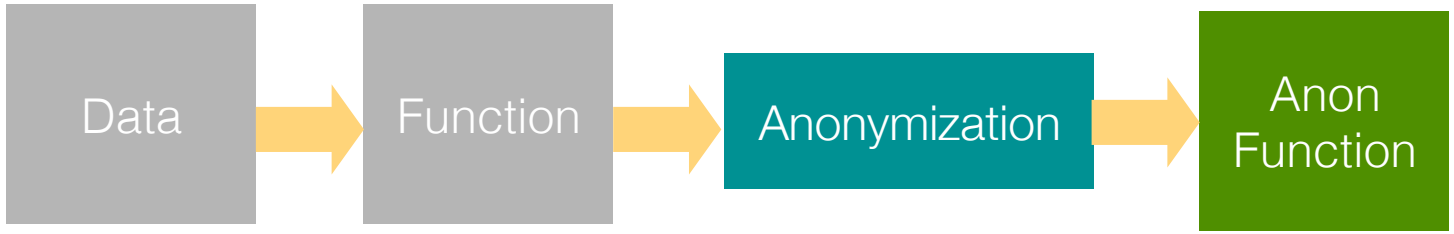
“Innocuous” DNA Statistics [Homer et al. 08]

De-anonymization of Netflix database [Narayanan, Shmatikov 08, 11]

Anonymization Methods

Input perturbation / generalization (e.g., k-anonymity)

Output perturbation (e.g., differentially private mechanisms)



k-anonymity

32	American	92043	Heart Disease
34	Japanese	92043	Cancer
38	American	92043	Cancer
37	French	92306	Viral Infection

$k = 4$

[30, 40]	*	92***	Heart Disease
[30, 40]	*	92***	Cancer
[30, 40]	*	92***	Cancer
[30, 40]	*	92***	Viral Infection

A record is indistinguishable from *k*-1 other records w.r.t. anonymized attributes. [Sweeney, 02]

Multidimensional methods available [LeFevre, DeWitt, Ramakrishnan 06]

l-diversity

[30, 40]	92***	Cancer
[30, 40]	92***	Cancer
[30, 40]	92***	Cancer
[30, 40]	92***	Viral Infection

k-anonymization can generate equivalence classes with low diversity, thus reducing its effectiveness.

[30, 40]	923**	Heart Disease
[30, 40]	923**	Cancer
[30, 40]	923**	Diabetes
[30, 40]	923**	Viral Infection

l-diversity: Equivalence class has at least l distinct elements.

Entropy l-diversity: Equivalence class has entropy > log l

Recursive l-diversity: Rarest element not too rare, Abundant element not too abundant.

Exercise: How might an adversary attack an l-diversity scheme?

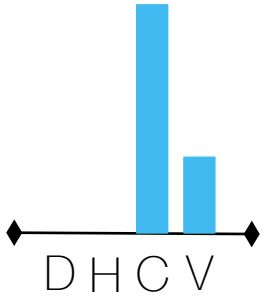
[Skewness and Similarity attacks.]

[Machanavajjhala et al. 07]

t-closeness

k-anonymity

[30, 40]	92***	Cancer
[30, 40]	92***	Cancer
[30, 40]	92***	Cancer
[30, 40]	92***	Viral Infection



l-diversity

[30, 40]	923**	Heart Disease
[30, 40]	923**	Cancer
[30, 40]	923**	Diabetes
[30, 40]	923**	Viral Infection

t-closeness

[45, 65]	92***	Viral Infection
[45, 65]	92***	Diabetes
[45, 65]	92***	Diabetes
[45, 65]	92***	Viral Infection

t-closeness tries to make the distribution of the sensitive attribute in the equivalence class match the distribution of the whole population.

[Li, Li, Venkitasubramanian, 07]

k-anonymity and variants

Stronger protection than simple masking.

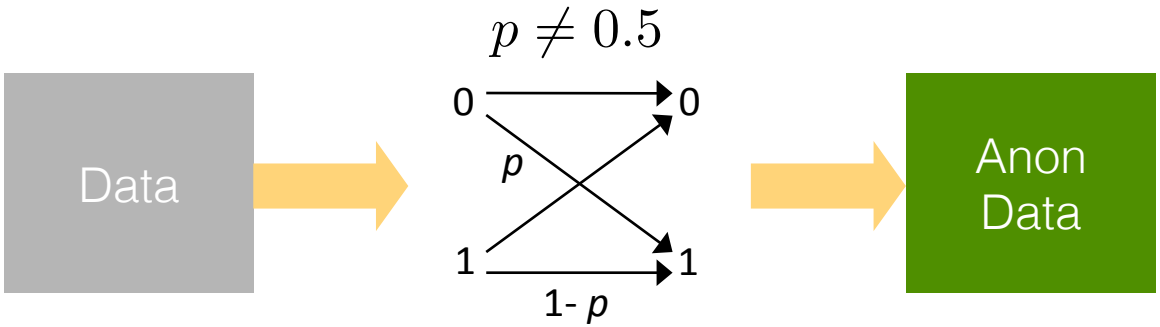
Leaks information if sensitive attribute has low diversity, e.g., all patients have cancer.

ℓ-diversity addresses diversity issue, but susceptible to skewness attacks on attribute values in an equivalence class. [Machanavajjhala et al. 07]

t-closeness address skewness, but destroys useful correlations in the process. [Li, Li, Venkitasubramanian, 07] [Domingo-Ferrer and Torra, 2008]

Randomized Response

Binary case: Given p, estimate % of 0/1 [Warner 65]



Post-Randomization [Kooiman, Willenborg, Gouweleeuw 98]

$$\begin{bmatrix} a_{1,1} & \cdots & a_{\ell,\ell} \\ \vdots & \ddots & \vdots \\ a_{\ell,1} & \cdots & a_{1,\ell} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_\ell \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_\ell \end{bmatrix}$$

Original PDF Perturbed PDF

Randomized Response

Simple: usually add noise to the data.

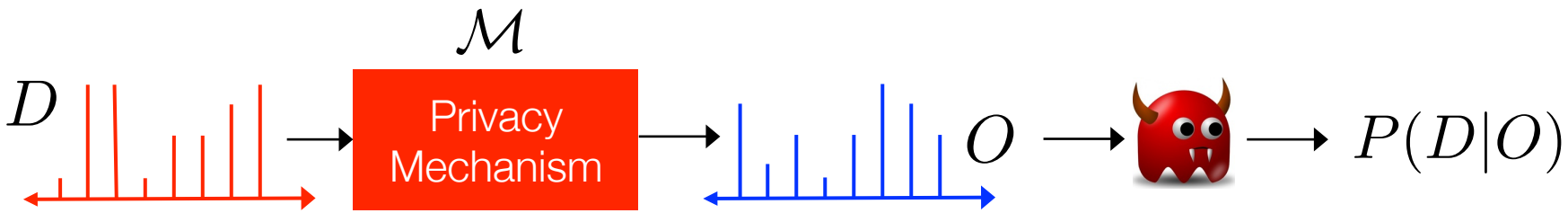
Good for aggregate statistics e.g., PMFs, means, etc.

Not suitable for many common tasks, e.g., max / min.

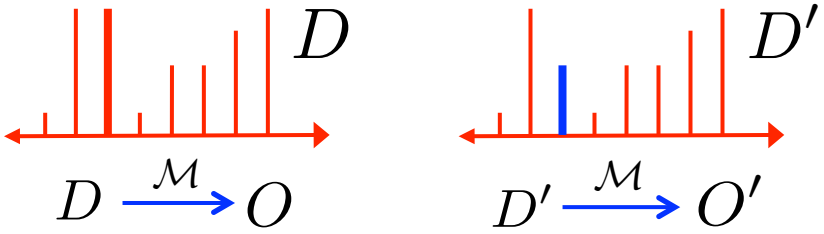
$$\begin{bmatrix} a_{1,1} & & \\ & \ddots & \\ & & a_{1,\ell} \end{bmatrix} \begin{bmatrix} p_1 \\ \vdots \\ p_\ell \end{bmatrix} = \begin{bmatrix} q_1 \\ \vdots \\ q_\ell \end{bmatrix}$$

Privacy-utility tradeoff degrades very rapidly upon composition, as PRAM matrices can become poorly conditioned. [Lin, Wang, Rane, 12]

Differential Privacy



Perfect privacy $\Rightarrow P(D|O) = P(D)$ useless in practice.

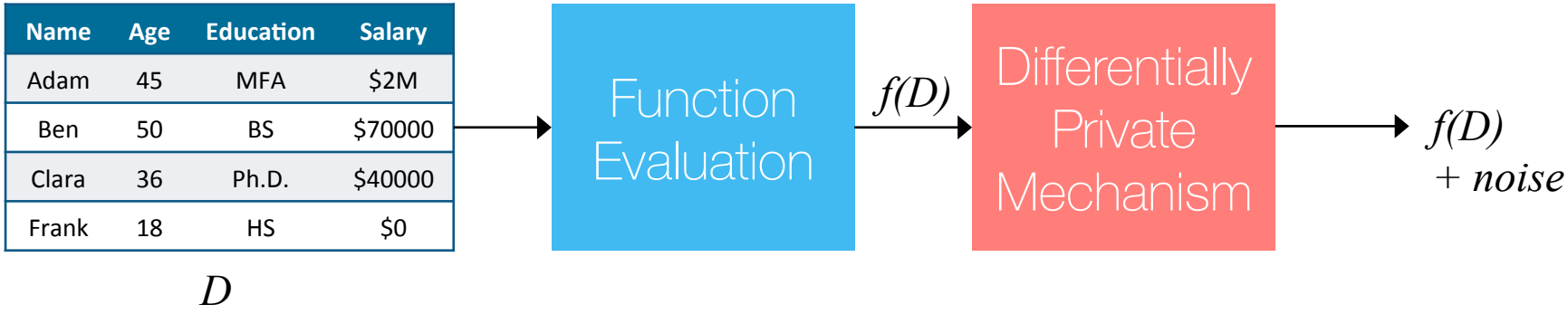


Need $\dfrac{P(O \in \mathcal{S}|D)}{P(O' \in \mathcal{S}|D')} \leq e^{\epsilon}$

Differential Privacy: Output is **insensitive** to any single element in D. Thus D and D' appear statistically indistinguishable to an adversary.

[Dwork, 06, 08, 09]

DP via output perturbation



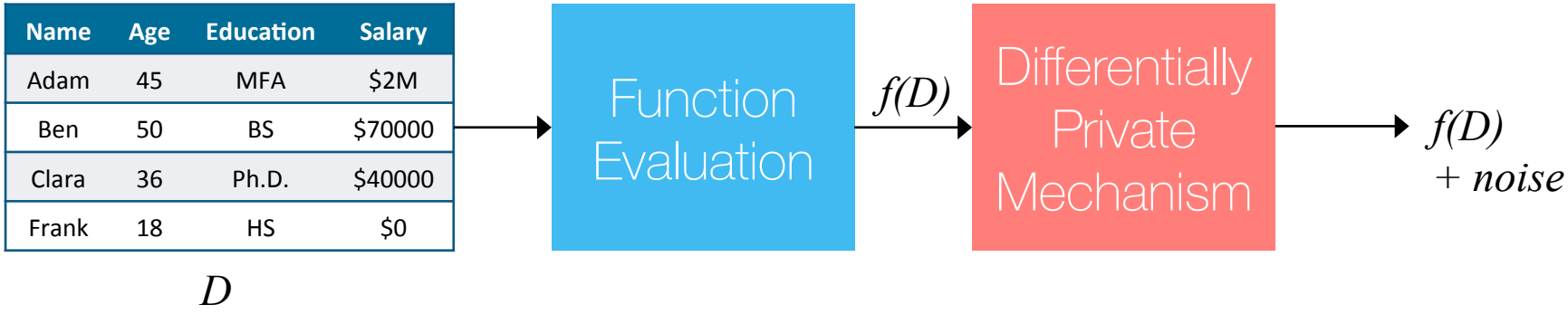
$f(D)$ is computed first, then noise is added by a trusted database curator before publishing $f(D)$.

Some examples of $f(D)$:

- Weights of a classifier trained on the data
- Count queries and histograms

Amount of noise added depends on ϵ and the sensitivity of the function.

Achieving DP



How much noise to add? Depends on the (global) sensitivity of f(D).

$$\Delta f = \max_{D,D'} \|f(D) - f(D')\|_1$$

Exercise: What is the sensitivity when f is: summation, count query, max query?

Higher sensitivity implies that more noise is needed to obfuscate the difference between D and D'.

The Laplace Mechanism

By far, the noise is most commonly sampled from the Laplacian distribution.

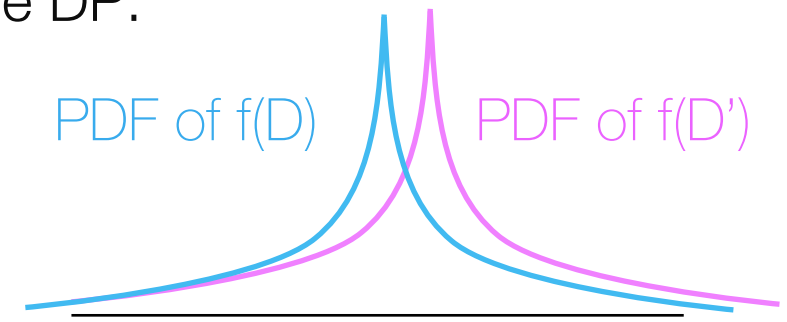
$$p_N(n) = \mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right) = \frac{\epsilon}{2\Delta f}\exp\left(-\frac{\epsilon|n|}{\Delta f}\right)$$

Lower ϵ implies more noise, which increases privacy, but lowers utility.

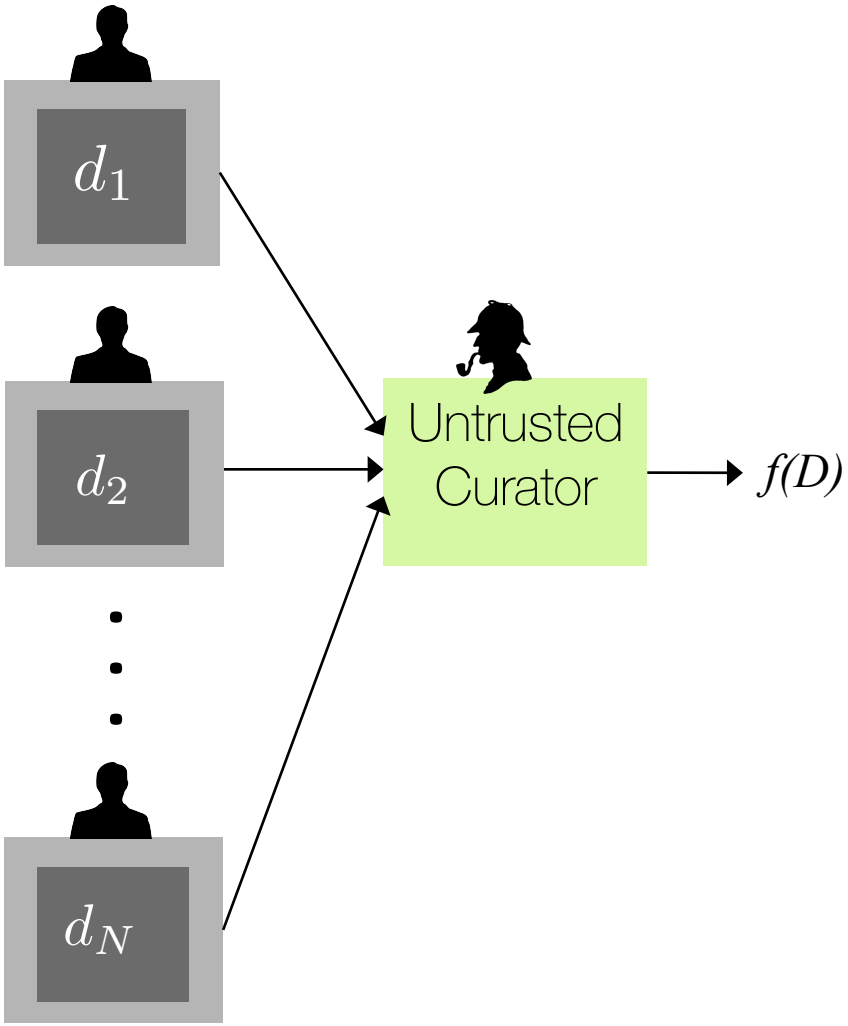
Higher Δf implies more noise is needed for a given level of privacy.

Other noise distributions that give DP:

- Exponential mechanism
- Gaussian mechanism



DP via input perturbation



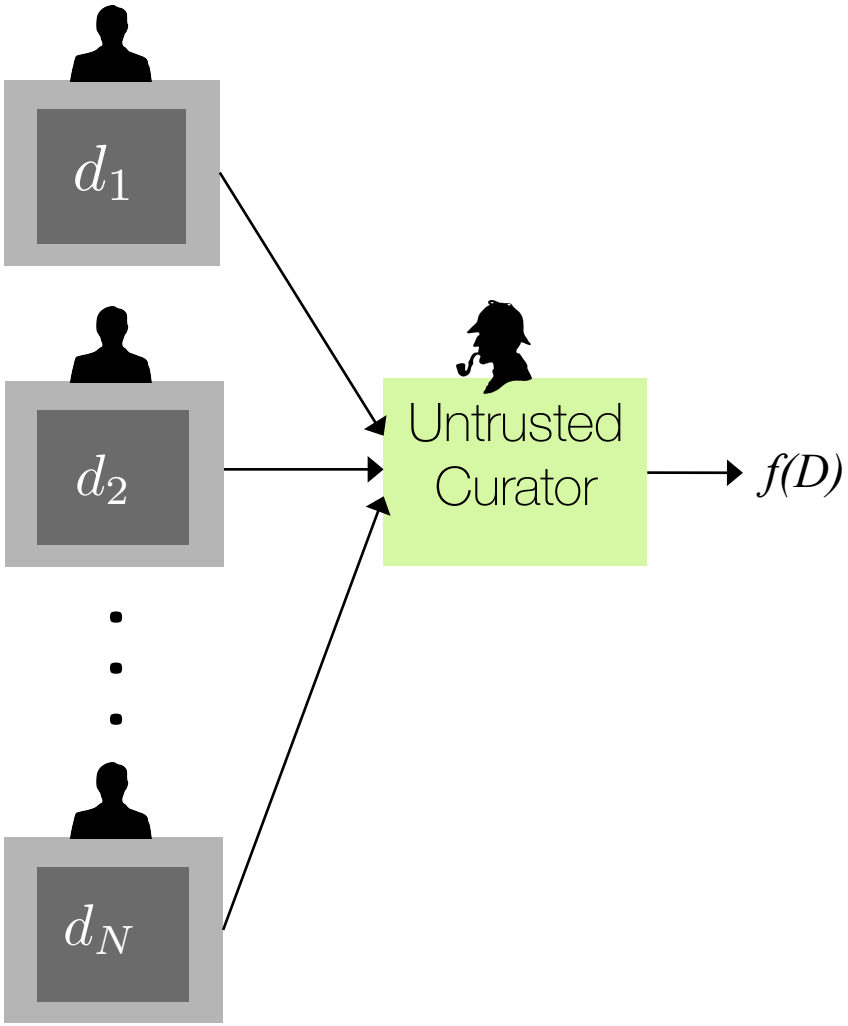
Parties don't trust curator, hence cannot reveal their data or even the specifications of the noise.

Two approaches:

- Compute $f(D)$ using SMC, then add noise using SMC.
- Add noise directly to data d_i, such that resulting noise in $f(D)$ adds up to just the right amount.

Example: Secure aggregation (more on this later)

Noise added by parties & curator



Each party chooses Laplacian noise based on its database. One of noises gets obliviously selected via SMC [Pathak et al., 2010]

Each participant adds symmetric geometric noise. Summation contains geometric noise which provides DP. [Shi et al., 2011]

Each participant adds gamma noise. Summation contains Laplacian noise which provides DP. [Acs & Castelluccia, 2011]

Relaxations & Enhancements

Epsilon-Delta Differential Privacy [Geng, Viswanath, 2013]

$$P(O \in \mathcal{S}|D) \leq e^{\epsilon} P(O \in \mathcal{S}|D) + \delta$$

Adding noise based on global sensitivity is too conservative. Can add noise based on "smooth sensitivity". [Nissim, Raskhodnikova, Smith 07]

For (much) more detail on DP, see tutorial slides by Sarwate and Chaudhuri from IEEE WIFS 2014:

http://www.ece.rutgers.edu/~asarwate/pdfs/WIFStutorial.pdf

Differential Privacy

Provides strong protection against adversaries with background information, unlike *k*-anonymity. [Kasiviswanathan, Smith, 08]

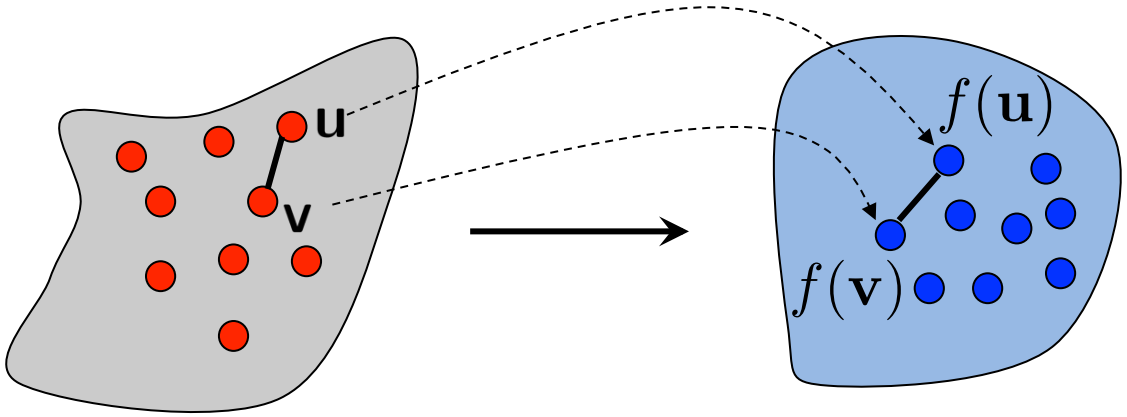
Additively composable, i.e., if two mechanisms provide DP, then their cascade provides DP (albeit lower privacy than before).

Treats all records as equally private, heavily obfuscates rare values.

Noise variance is proportional to sensitivity of the function being published. Hard to determine.

Privacy deteriorates with the number of queries. [Dwork 10]

Nearest Neighbor Embeddings



n points in
K-dimensional space

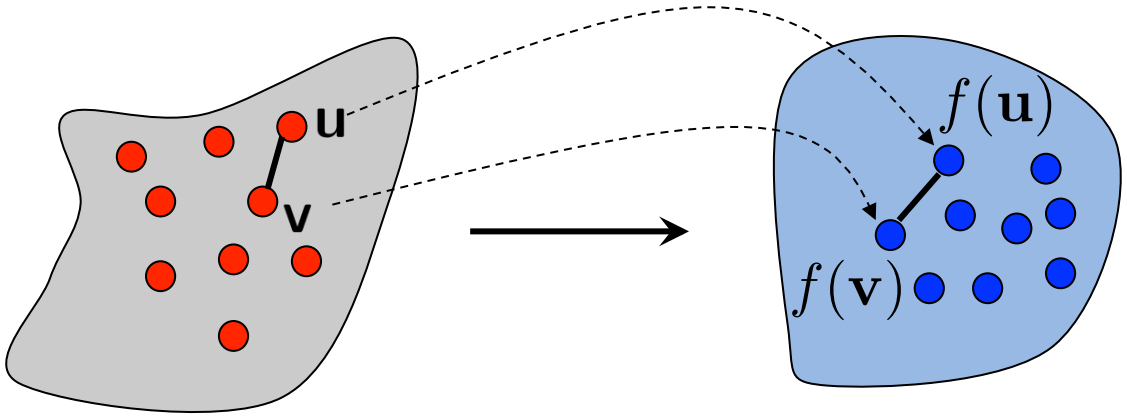
n points in
M-dimensional space

Choose $M \approx \frac{\log n}{\epsilon^2}$ (Dimensionality Reduction)

$$(1-\epsilon)\|\mathbf{u}-\mathbf{v}\|^2 \leq \|f(\mathbf{u})-f(\mathbf{v})\|^2 \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\|^2$$

[Johnson, Lindenstraus, 1984]

$$f(\mathbf{u}) = \frac{1}{\sqrt{M}}\mathbf{A}\mathbf{u}$$ is a NN Embedding



n points in
K-dimensional space

n points in
M-dimensional space

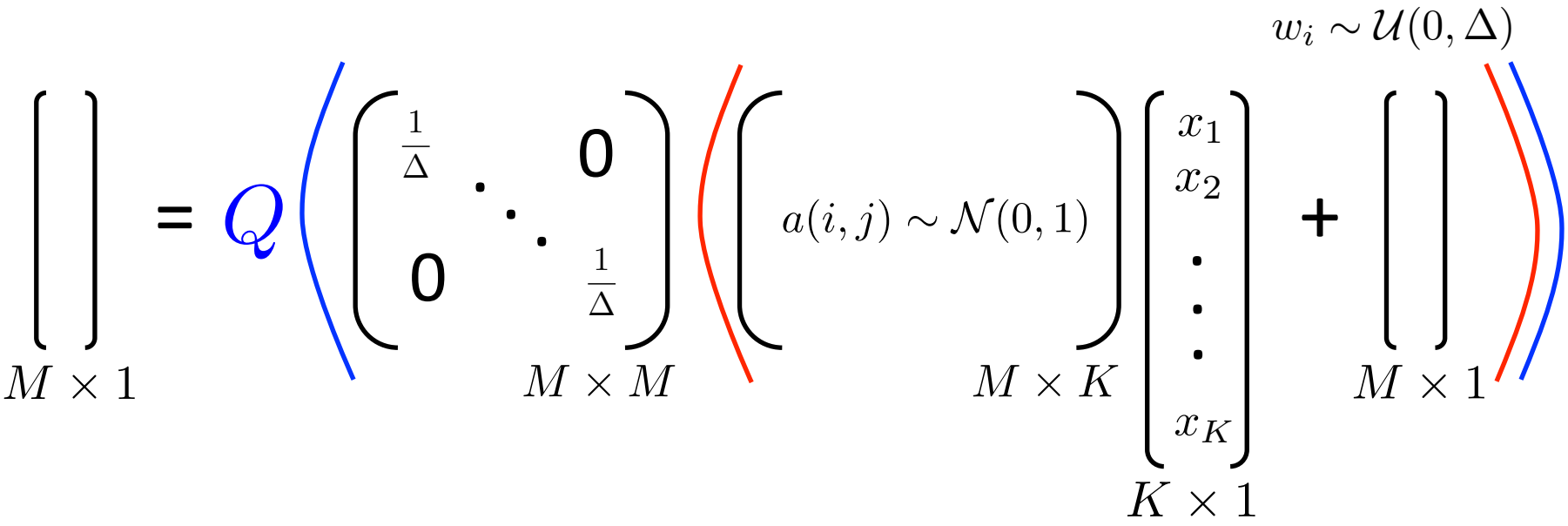
$$\mathbf{A} = \begin{pmatrix} & K & \\ & a(i,j) \sim \mathcal{N}(0,1) & \end{pmatrix} \quad M \approx \frac{\log n}{\epsilon^2}$$

$f(\cdot)$ preserves Euclidean distance with high probability.

[Indyk, 1998; Achlioptas, 2003]

“Secure” Signal Embeddings

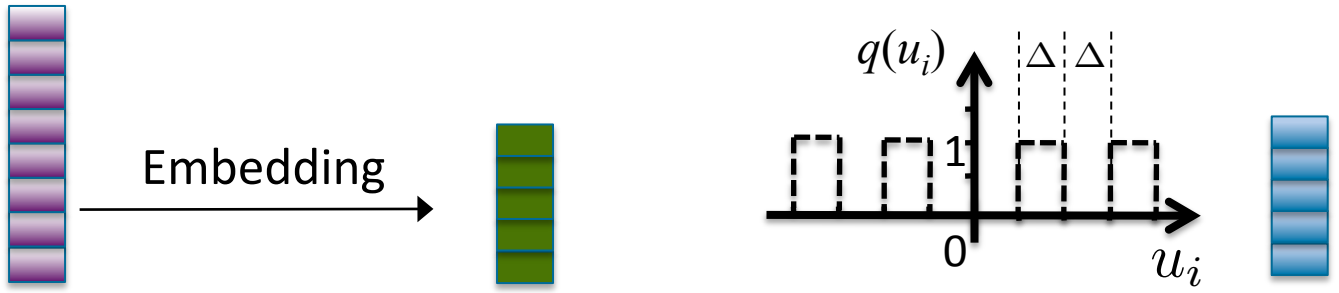
$$\mathbf{q} = Q(\mathbf{\Delta^{-1}}(\mathbf{Ax} + \mathbf{w}))$$



Each q_i is a measurement of $\mathbf{x}$. There are M measurements.

We are interested in the properties of $\mathbf{q}$.

Quantization w/ Index Reuse



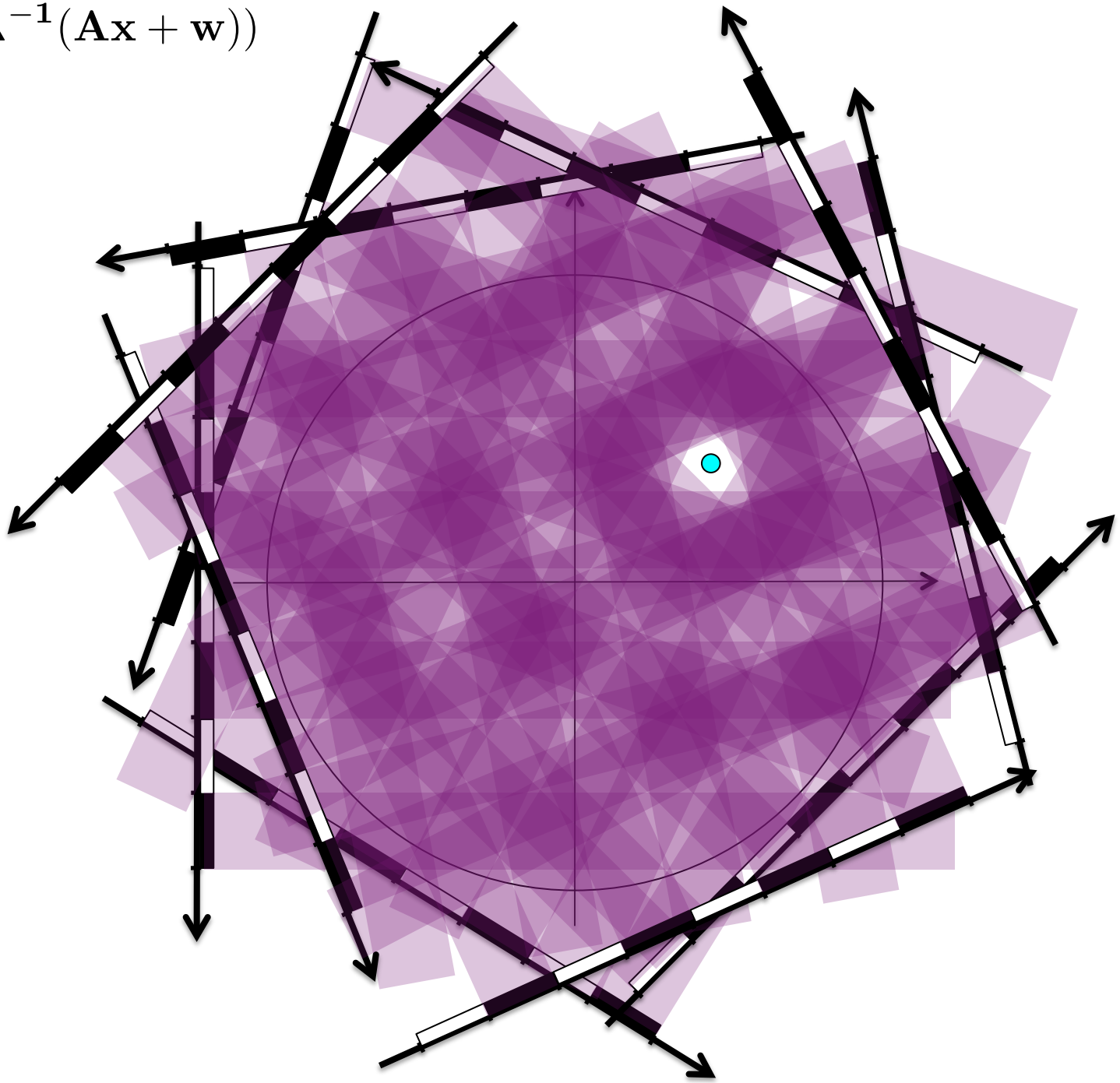
$\mathbf{u}$

$$f(\mathbf{u}) = \frac{1}{\sqrt{M}}\mathbf{A}\mathbf{u}$$

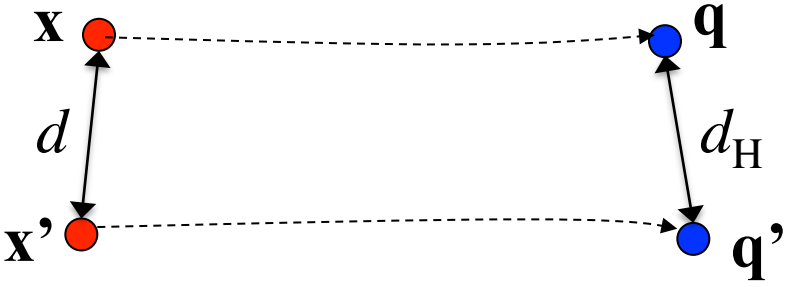
Unlike JL, this is a mapping from L2 to L1 space. [Boufounos, R, 2011]

Can use unary embedding to get L1 to L1 mapping. [Kushilevitz, Ostrovsky, Rabani, 2000][R, Boufounos, Vetro, 2013].

$$\mathbf{q} = Q(\mathbf{\Delta}^{-1}(\mathbf{A}\mathbf{x} + \mathbf{w}))$$



"Consistent" Signals Imply Similar Hashes



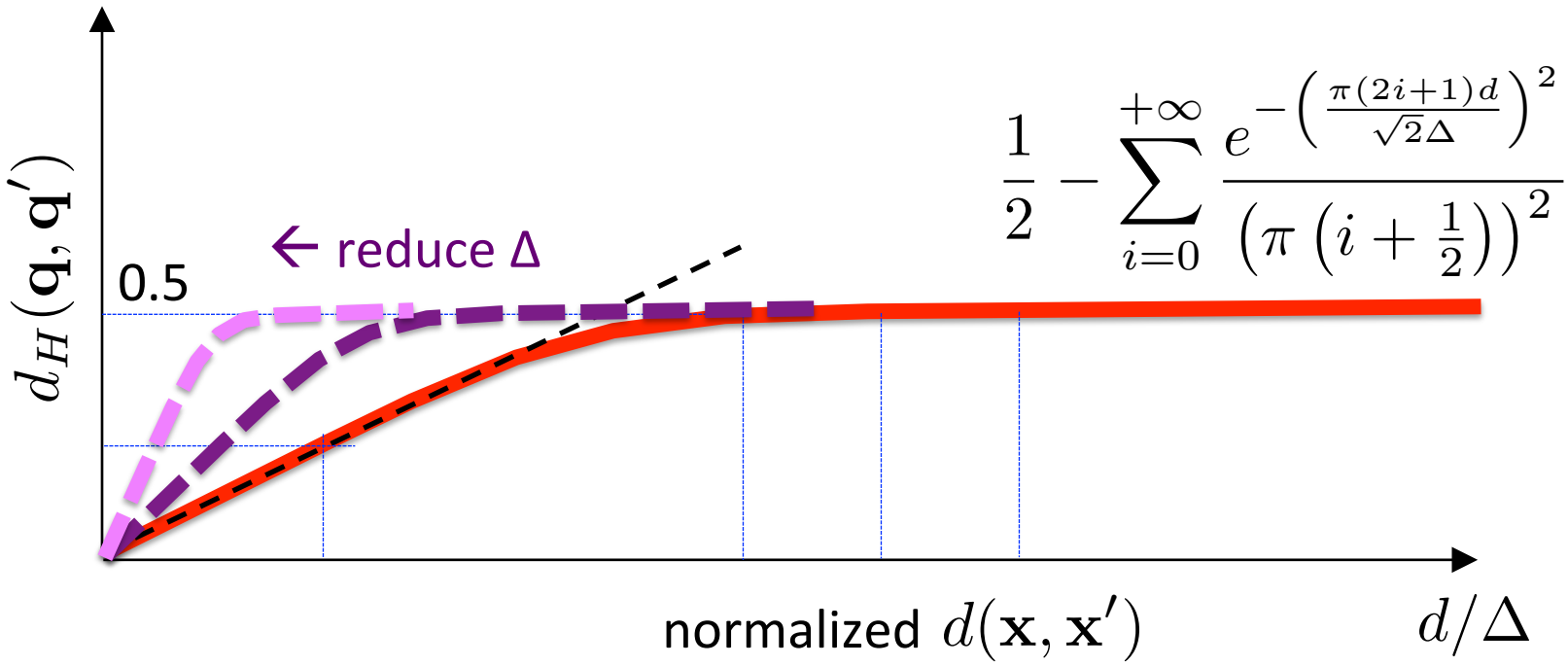
x, x' consistent under i^{th} measurement $\iff$ $q_i = q_i'$

$$P(\mathbf{x}, \mathbf{x}' \text{ consistent}|d) = \frac{1}{2} + \sum_{i=0}^{+\infty} \frac{e^{-\left(\frac{\pi(2i+1)d}{\sqrt{2}\Delta}\right)^2}}{\left(\pi\left(i+\frac{1}{2}\right)\right)^2} = P_{c|d}$$ (shorthand)

[Boufounos, R, 2011]

$$1 - P(\mathbf{x}, \mathbf{x}' \text{ consistent}|d) = P(q_i \neq q_i'|d) \approx d_H(\mathbf{q}, \mathbf{q}')$$

Distance Preservation (Theory)



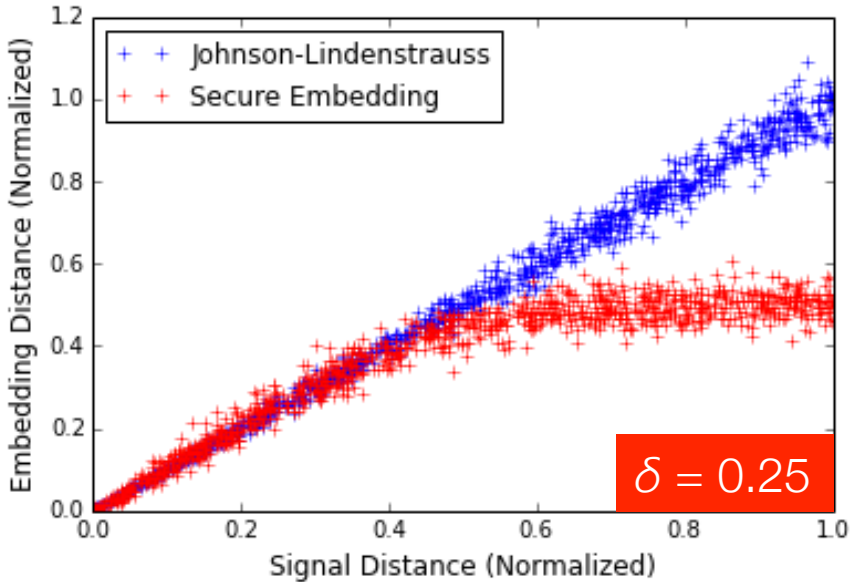
If **x** and **x'** are close, **q** and **q'** are proportionately close.

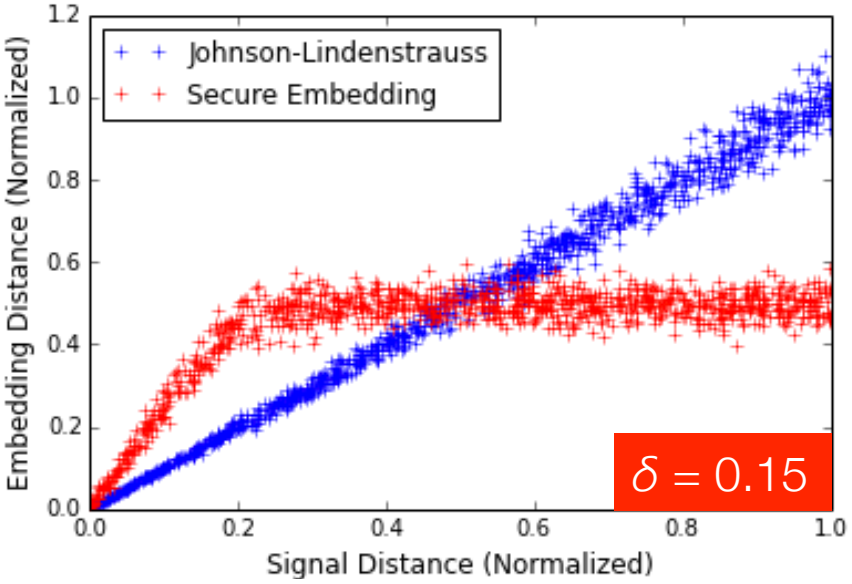
If **x** and **x'** are far, **q** and **q'** are not indicative.

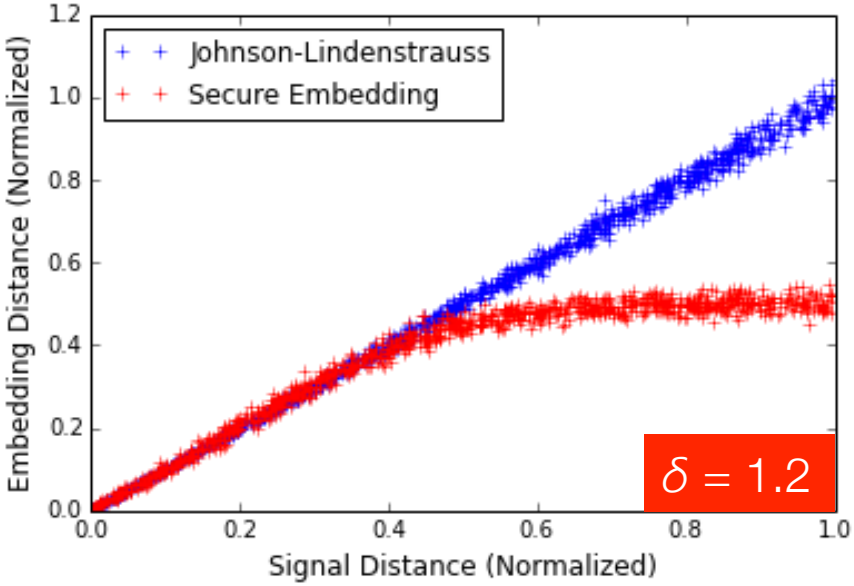
How far is far enough? Slope controlled via Δ.

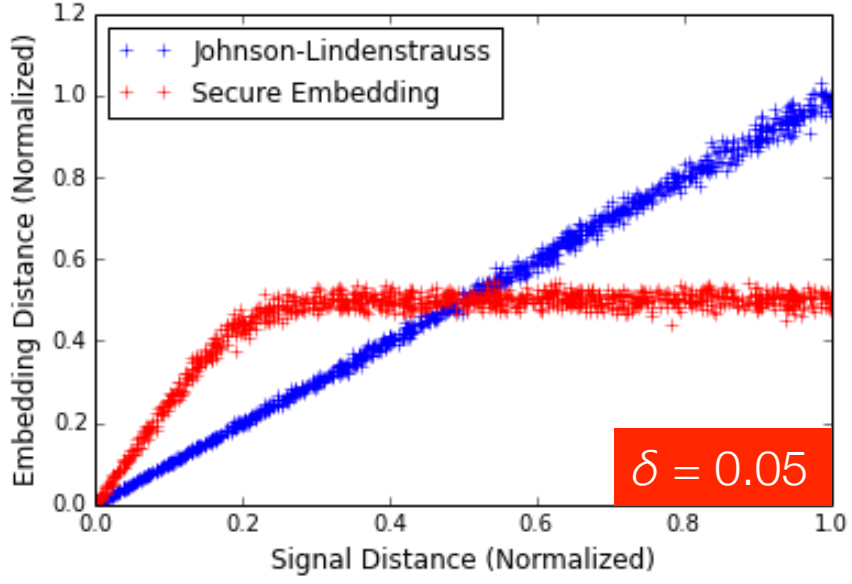
[Boufounos, R, 2013]

Distance Preservation (Observed)

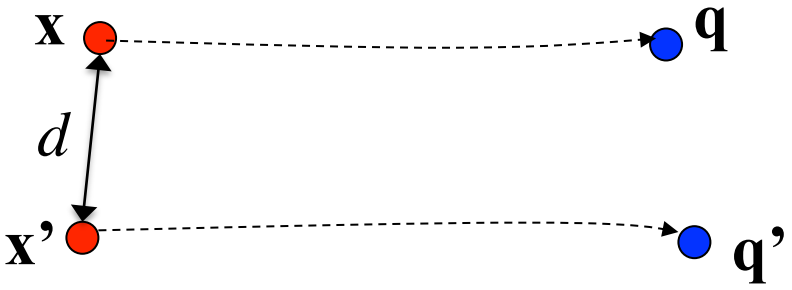








Asymptotically Unconditional Security



Given **q**, how difficult is it to guess **q'** ?

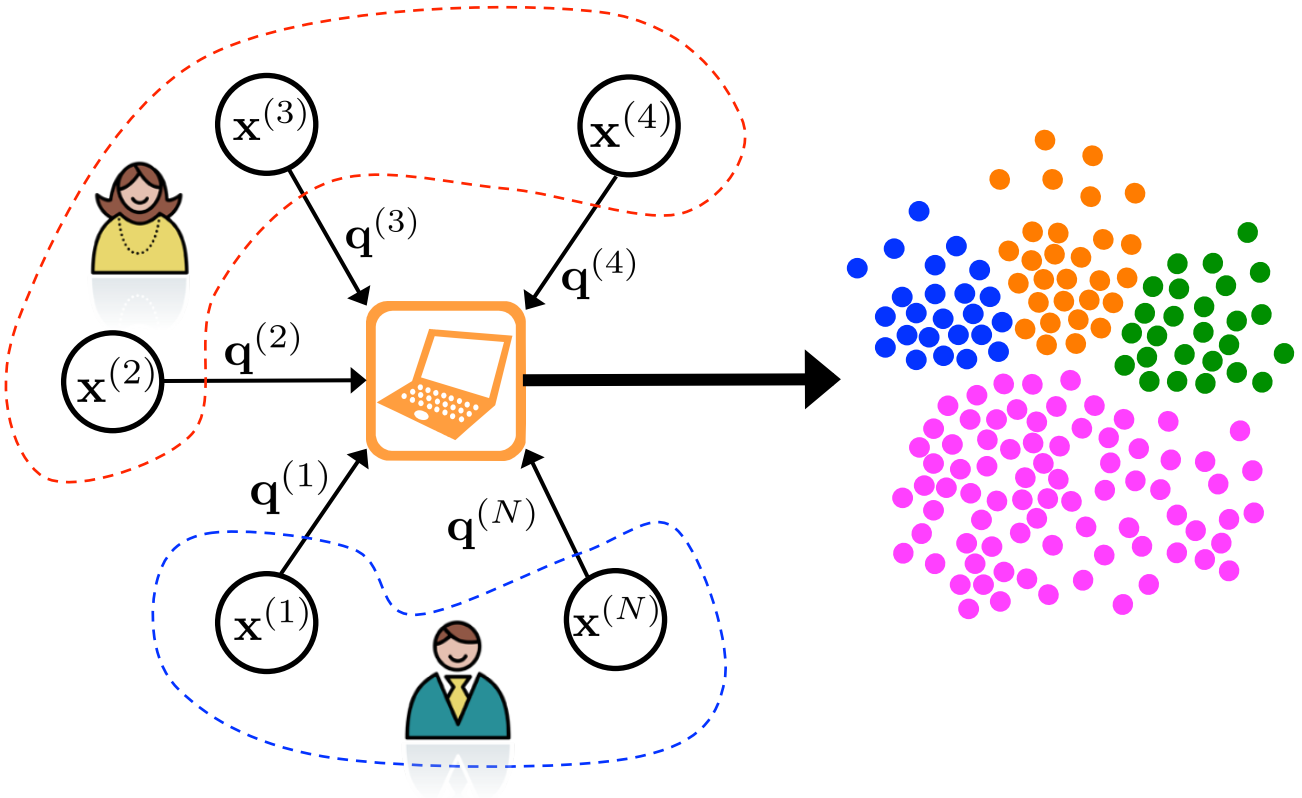
$$I(q_i; q_i'|d) = \sum_{q_i, q_i' \in \{0,1\}} P(q_i, q_i'|d) \log \frac{P(q_i, q_i'|d)}{P(q_i|d)P(q_i'|d)}$$

$$= P_{c|d} \log(2P_{c|d}) + (1 - P_{c|d}) \log(2(1 - P_{c|d}))$$

$$\leq 10e^{-\left(\frac{\pi\sigma d}{\sqrt{2}\Delta}\right)^2}$$

If two vectors are far enough, their hashes are (almost) independent.

Privacy Preserving Clustering

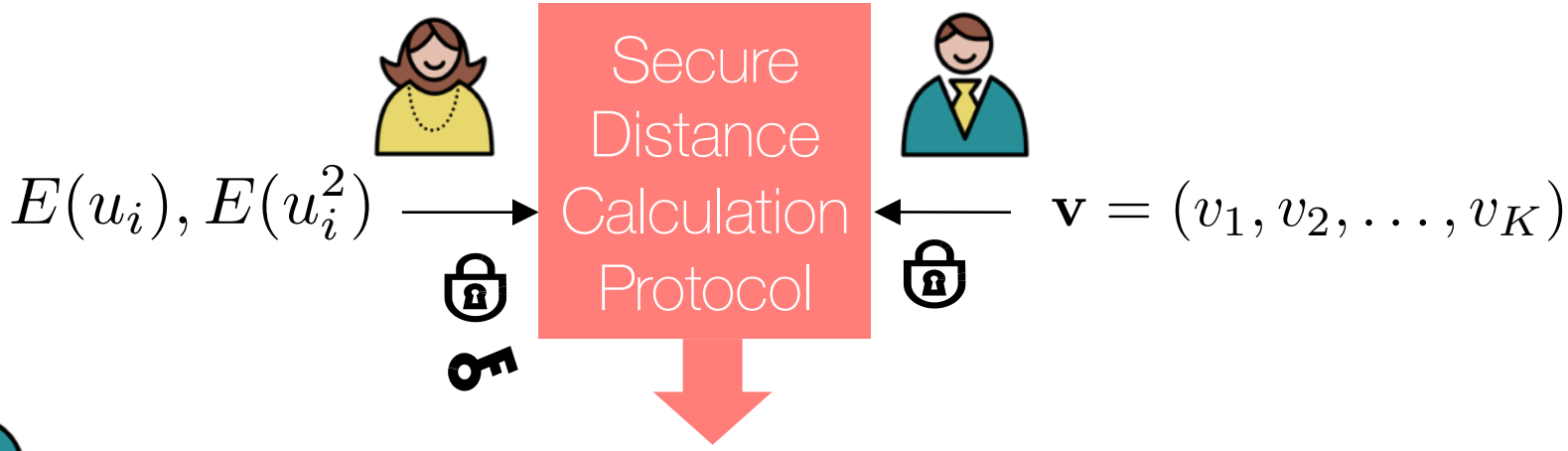


Owners of N vectors have embedding parameters $\mathbf{A}$, Δ, $\mathbf{w}$, $\mathrm{Q}(.)$

Researcher performs clustering in hash space. Does not discover any data point.

No encrypted-domain calculation necessary.

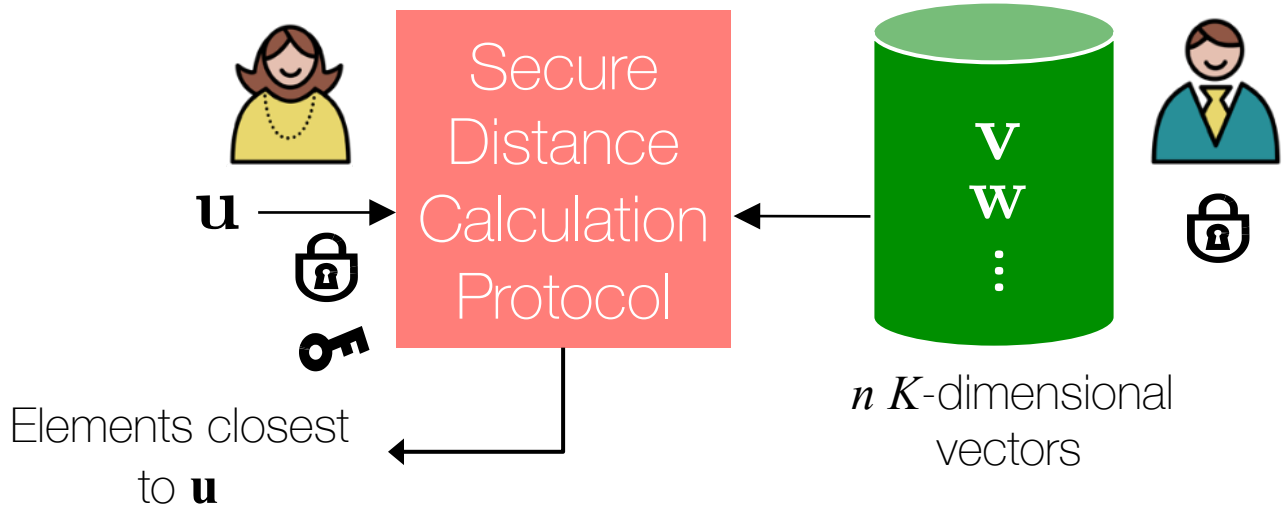
Recall: Secure Euclidean Distance



$$\left(\prod_{i=1}^{K} E(u_i^2)\right) E\left(\sum_{i=1}^{K} v_i^2\right) \left(\prod_{i=1}^{K} E(u_i)^{-2v_i}\right) =$$

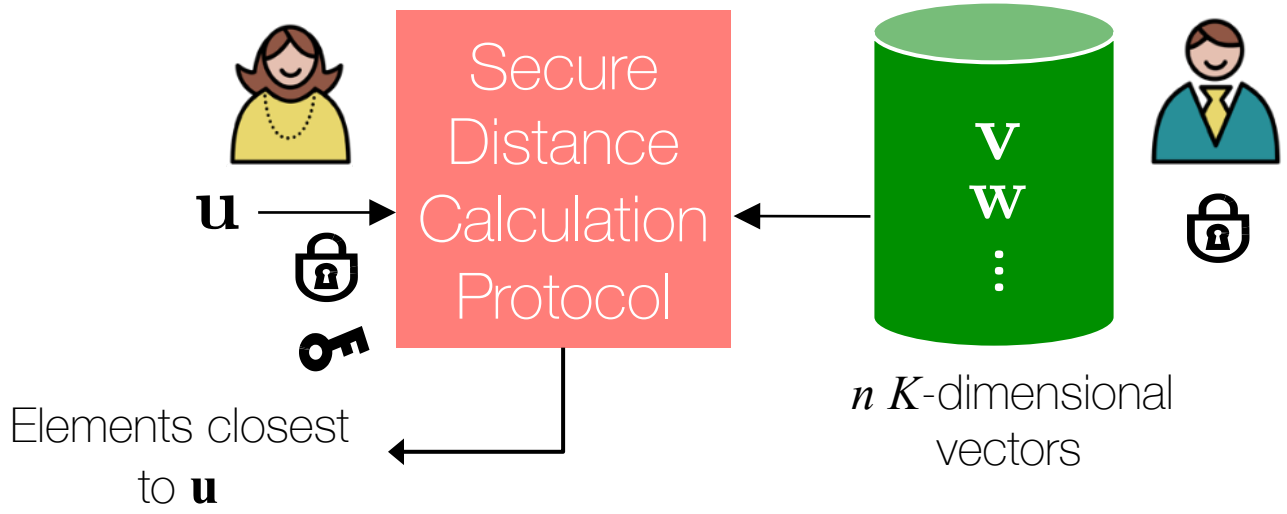
$$E\left(\sum_{i=1}^{K} u_i^2\right) E\left(\sum_{i=1}^{K} v_i^2\right) E\left(\sum_{i=1}^{K} -2u_i v_i\right) = E\left(d(\mathbf{u}, \mathbf{v})\right)$$

Private NN with Embeddings



1. Alice and Bob share embedding parameters, $\mathbf{A}$, $\mathbf{w}$, Δ
2. They compute binary embeddings of all vectors.
3. Bob computes encrypted pairwise distances between his embeddings and $\mathbf{q}(\mathbf{u})$ and returns them to Alice.
4. Alice decrypts the distances, and determines indices for which the distances are informative, e.g., below $0.5 - \gamma$.

Private NN with Embeddings (contd)



5. Suppose there are m informative distances. Alice executes m-of-n OT to obtain the corresponding vectors from Bob.
6. No need to hide distances of the far neighbors as they are uninformative.

Complexity *

			 [Qi, Atallah, 2008]
# Encryptions	$O(n)$	$O(n)$	$O(n)$
# Decryptions	$O(n)$	$O(n)$	$O(n)$
# Multiplications	0	$O(n)$	$O(n)$
# Exponentiations	0	$O(n)$	$O(n)$
# Secure Sorting	0	0	$O(n)$
# Transmissions	$O(n)$	$O(n)$	$O(n)$

* Not taking into account, dimensionality reduction obtained via embedding

References (Secret Sharing)

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, *22*(11), 612-613.

Chaum, D., Crépeau, C., & Damgard, I. (1988, January). Multiparty unconditionally secure protocols. In *Proceedings of the twentieth annual ACM symposium on Theory of computing* (pp. 11-19). ACM.

Ben-Or, M., Goldwasser, S., & Wigderson, A. (1988, January). Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the twentieth annual ACM symposium on Theory of computing* (pp. 1-10). ACM.

Wang, Y., Ishwar, P., & Rane, S. (2013, July). Information-theoretically secure three-party computation with One corrupted party. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on* (pp. 3160-3164). IEEE.

Bogdanov, Dan, Sven Laur, and Jan Willemson. "Sharemind: A framework for fast privacy-preserving computations." *Computer Security-ESORICS 2008*. Springer Berlin Heidelberg, 2008. 192-206.

Bogdanov, Dan, et al. *Rmind: a tool for cryptographically secure statistical analysis*. Cryptology ePrint Archive, Report 2014/512, 2014.

References (Garbled Circuits)

Huang, Y., Evans, D., Katz, J., & Malka, L. (2011, August). Faster Secure Two-Party Computation Using Garbled Circuits. In *USENIX Security Symposium* (Vol. 201, No. 1).

Yao, A. (1986, October). How to generate and exchange secrets. In *Foundations of Computer Science, 1986., 27th Annual Symposium on* (pp. 162-167). IEEE.

Kolesnikov, V., & Schneider, T. (2008). Improved garbled circuit: Free XOR gates and applications. In *Automata, Languages and Programming* (pp. 486-498). Springer Berlin Heidelberg.

Malkhi, D., Nisan, N., Pinkas, B., & Sella, Y. (2004, August). Fairplay-Secure Two-Party Computation System. In *USENIX Security Symposium* (Vol. 4).

Henecka, W., Sadeghi, A. R., Schneider, T., & Wehrenberg, I. (2010, October). TASTY: tool for automating secure two-party computations. In *Proceedings of the 17th ACM conference on Computer and communications security* (pp. 451-462). ACM.

Pinkas, B., Schneider, T., Smart, N. P., & Williams, S. C. (2009). Secure two-party computation is practical. In *Advances in Cryptology–ASIACRYPT 2009* (pp. 250-267). Springer Berlin Heidelberg.

References (OPE and CryptDB)

Boldyreva, A., Chenette, N., Lee, Y., & O'neill, A. (2009). Order-preserving symmetric encryption. In *Advances in Cryptology-EUROCRYPT 2009* (pp. 224-241). Springer Berlin Heidelberg.

Boldyreva, A., Chenette, N., Lee, Y., & O'neill, A. (2009). Order-preserving symmetric encryption. In *Advances in Cryptology-EUROCRYPT 2009* (pp. 224-241). Springer Berlin Heidelberg.

Popa, R. A., Redfield, C., Zeldovich, N., & Balakrishnan, H. (2011, October). CryptDB: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles* (pp. 85-100). ACM.

Popa, R. A., Zeldovich, N., & Balakrishnan, H. (2011). CryptDB: A practical encrypted relational DBMS.

References: Anonymization

Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *10*(05), 557-570.

Narayanan, A., & Shmatikov, V. (2006). How to break anonymity of the netflix prize dataset. *arXiv preprint cs/0610105*.

Narayanan, A., & Shmatikov, V. (2008, May). Robust de-anonymization of large sparse datasets. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on* (pp. 111-125). IEEE.

Machanavajjhala, A., Kifer, D., Gehrke, J., & Venkitasubramaniam, M. (2007). l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *1*(1), 3.

Li, N., Li, T., & Venkatasubramanian, S. (2007, April). t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on* (pp. 106-115). IEEE.

Homer, N., Szelinger, S., Redman, M., Duggan, D., Tembe, W., Muehling, J., ... & Craig, D. W. (2008). Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genet*, *4*(8), e1000167.

Domingo-Ferrer, J., & Torra, V. (2008, March). A critique of k-anonymity and some of its enhancements. In *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on* (pp. 990-993). IEEE.

LeFevre, K., DeWitt, D. J., & Ramakrishnan, R. (2006, April). Mondrian multidimensional k-anonymity. In *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on* (pp. 25-25). IEEE.

References: Randomized Response & Differential Privacy

Warner, S. L. (1965). Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, *60*(309), 63-69.

Gouweleeuw, J. M., Kooiman, P., & de Wolf, P. P. (1998). Post randomisation for statistical disclosure control: Theory and implementation. *Journal of official Statistics*, *14*(4), 463.

Lin, B. R., Wang, Y., & Rane, S. (2013). On the benefits of sampling in privacy preserving statistical analysis on distributed databases. *arXiv preprint arXiv:1304.4613*.

Dwork, C. (2008). Differential privacy: A survey of results. In *Theory and applications of models of computation* (pp. 1-19). Springer Berlin Heidelberg.

Dwork, C. (2011). Differential privacy. In *Encyclopedia of Cryptography and Security* (pp. 338-340). Springer US.

McSherry, F., & Talwar, K. (2007, October). Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on* (pp. 94-103). IEEE.

Nissim, K., Raskhodnikova, S., & Smith, A. (2007, June). Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing* (pp. 75-84). ACM.

Kasiviswanathan, S. P., & Smith, A. (2008). A note on differential privacy: Defining resistance to arbitrary side information. *CoRR abs/0803.3946*.

References: Differential Privacy (contd)

Geng, Quan, and Pramod Viswanath. "The optimal mechanism in differential privacy." *arXiv preprint arXiv:1212.1186* (2012).

Sarwate and Chaudhuri, “Tutorial on Differential Privacy, IEEE WIFS 2014, Atlanta, Georgia, [http://www.ece.rutgers.edu/~asarwate/pdfs/WIFStutorial.pdf]

Ninghui Li, Wahbeh Qardaji, and Dong Su. On sampling, anonymization, and differential privacy: Or, k-anonymization meets differential privacy. http://arxiv.org/abs/1101.2604, 2011.

References: Embeddings

Boufounos, P., & Rane, S. (2011, November). Secure binary embeddings for privacy preserving nearest neighbors. In *Information Forensics and Security (WIFS), 2011 IEEE International Workshop on* (pp. 1-6). IEEE.

Boufounos, P. T., & Rane, S. (2013, March). Efficient coding of signal distances using universal quantized embeddings. In *Data Compression Conference (DCC), 2013* (pp. 251-260). IEEE.

Rane, S., & Boufounos, P. T. (2013). Privacy-preserving nearest neighbor methods: comparing signals without revealing them. *Signal Processing Magazine, IEEE*, *30*(2), 18-28.

Indyk, Piotr, and Rajeev Motwani. "Approximate nearest neighbors: towards removing the curse of dimensionality." *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM, 1998.

Achlioptas, Dimitris. "Database-friendly random projections: Johnson-Lindenstrauss with binary coins." *Journal of computer and System Sciences* 66.4 (2003): 671-687.

Kushilevitz, Eyal, Rafail Ostrovsky, and Yuval Rabani. "Efficient search for approximate nearest neighbor in high dimensional spaces." *SIAM Journal on Computing* 30.2 (2000): 457-474.

5

Case Study: Private Aggregation

SOLUTIONS COMPARED

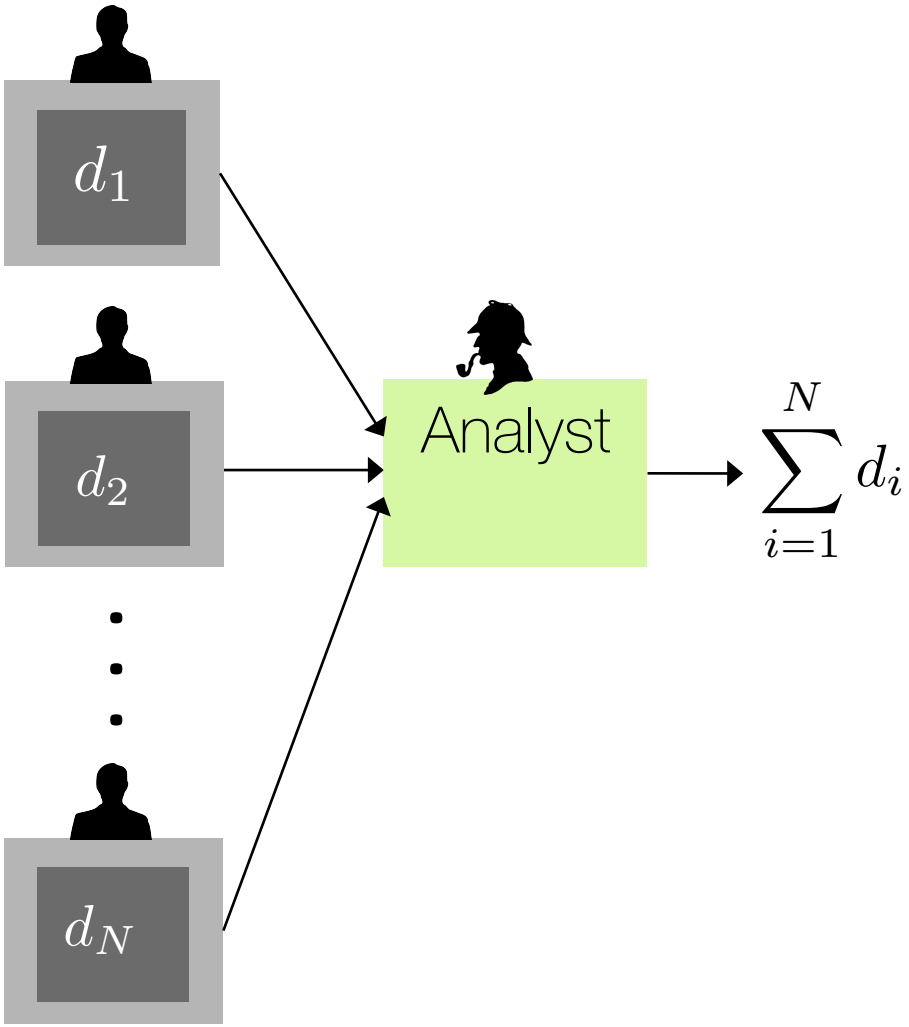
Vanishing keys

Secret sharing

Homomorphic encryption

Differential privacy

Setting: Personal Privacy



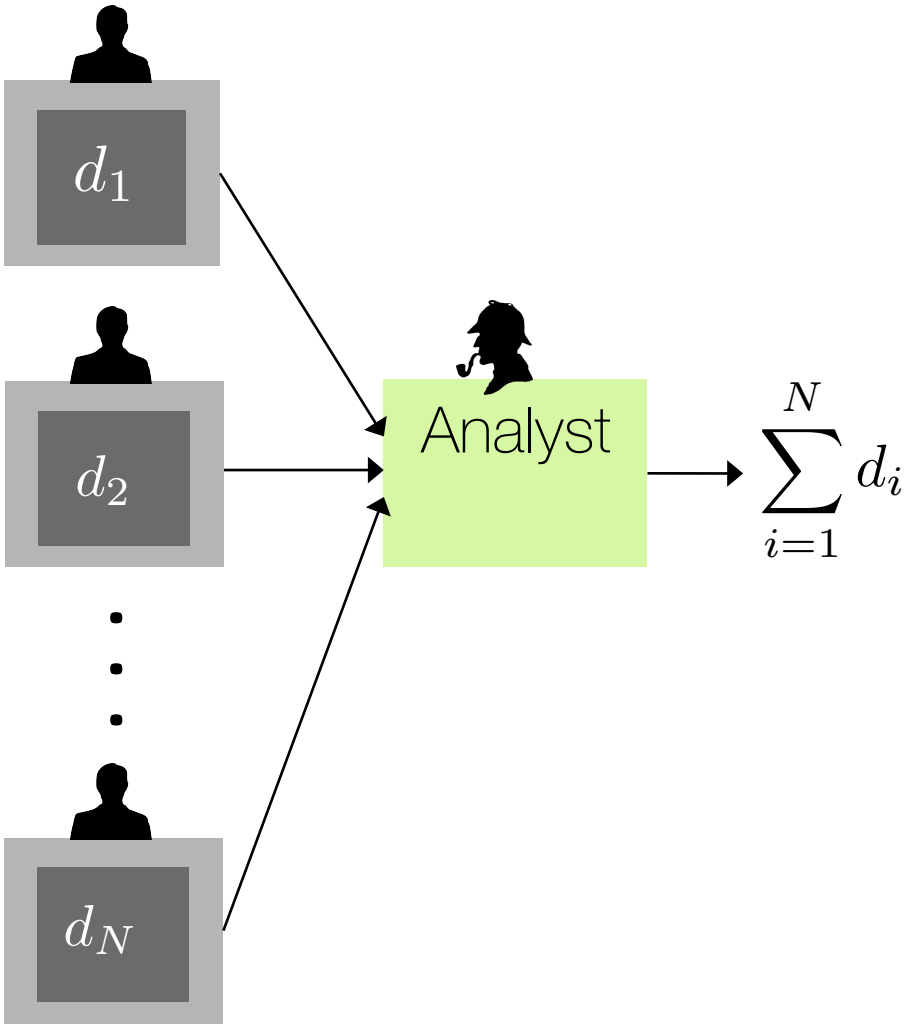
Applications

- e-voting
- smart-meter data aggregation
- fitness data analysis
- browser usage statistics

Topology Constraints

Participants cannot communicate with each other.

Privacy Constraints



Aggregator Obliviousness

Aggregator should discover nothing other than the summation.

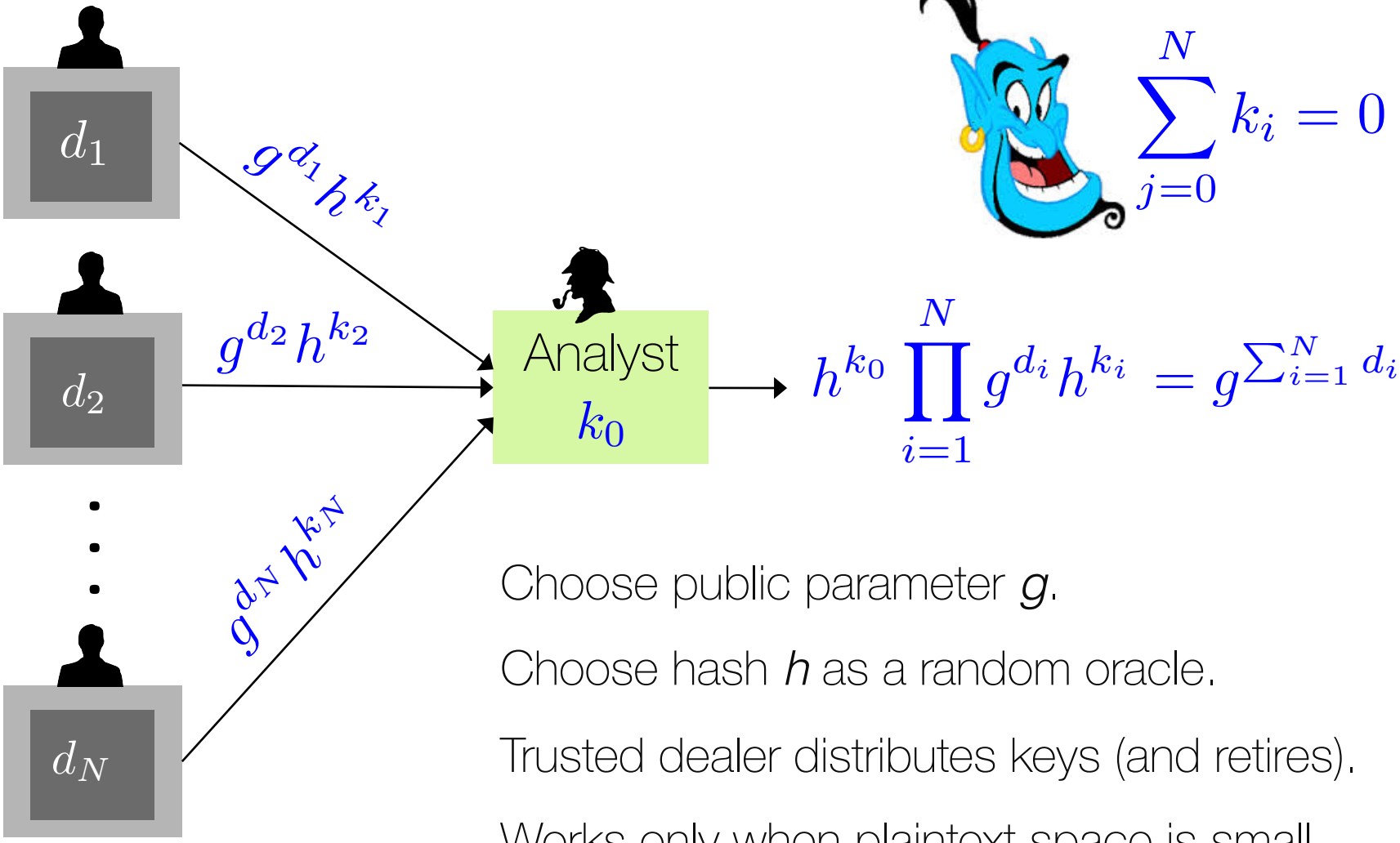
Participant Obliviousness

Participants should not discover the data of any other participants

Adversarial Model

Semi-honest (for now)

Soln #1: Trusted Dealer



Choose public parameter *g*.

Choose hash *h* as a random oracle.

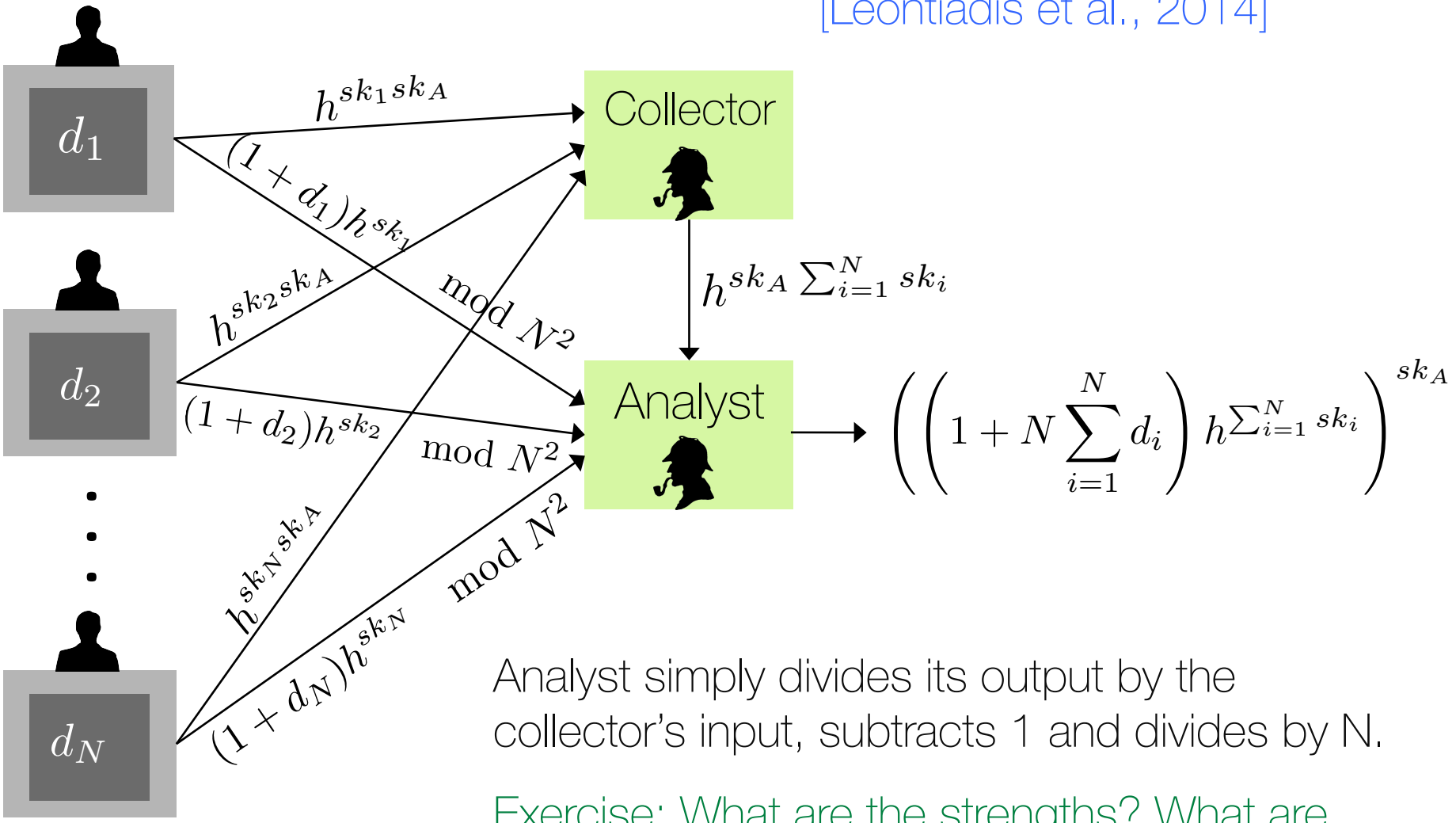
Trusted dealer distributes keys (and retires).

Works only when plaintext space is small.

[Shi et al., 2011; Chan et al., 2012; Bilogrevic et al., 2014]

Soln #2: Untrusted Collector

[Leontiadis et al., 2014]

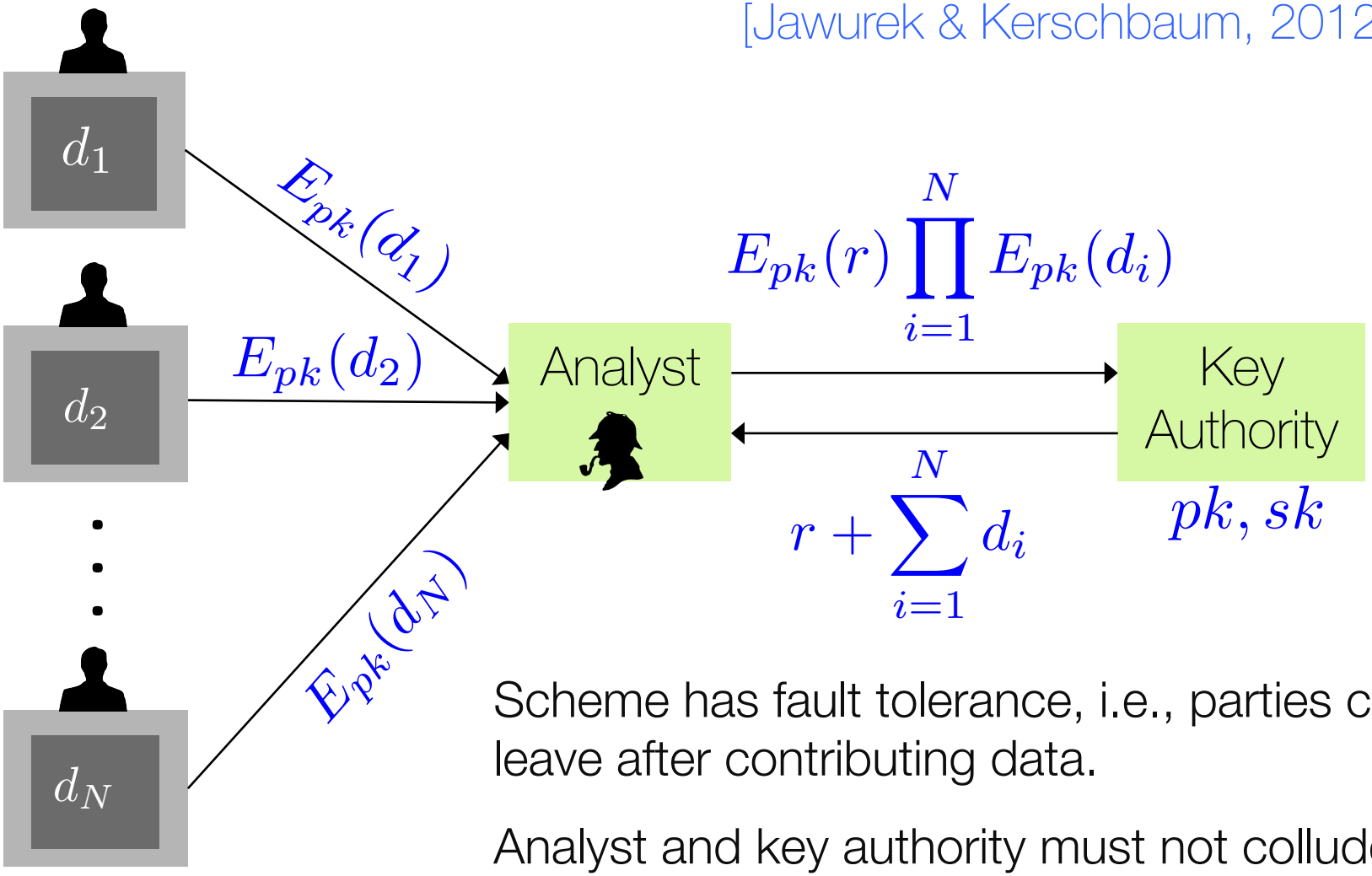


Analyst simply divides its output by the collector's input, subtracts 1 and divides by N.

Exercise: What are the strengths? What are the weaknesses?

Soln #3: Untrusted Key Authority

[Jawurek & Kerschbaum, 2012]

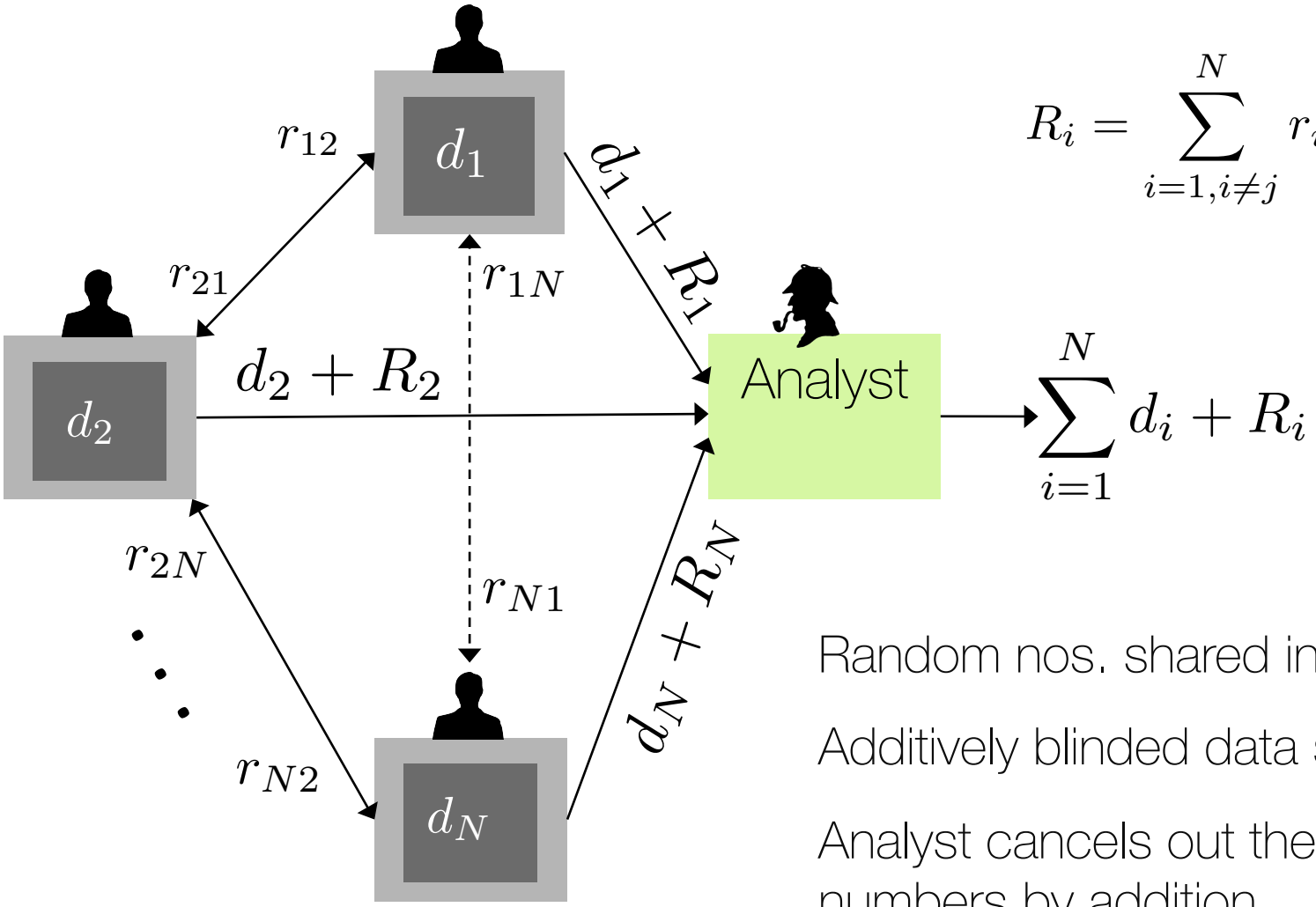


Scheme has fault tolerance, i.e., parties can leave after contributing data.

Analyst and key authority must not collude!

[Jawurek & Kerschbaum, 2012]

Soln #4: Secret Sharing in Setup



$$R_i = \sum_{i=1, i \neq j}^{N} r_{ij} - \sum_{i=1, i \neq j}^{N} r_{ji}$$

$$\sum_{i=1}^{N} d_i + R_i$$

Random nos. shared in setup phase.

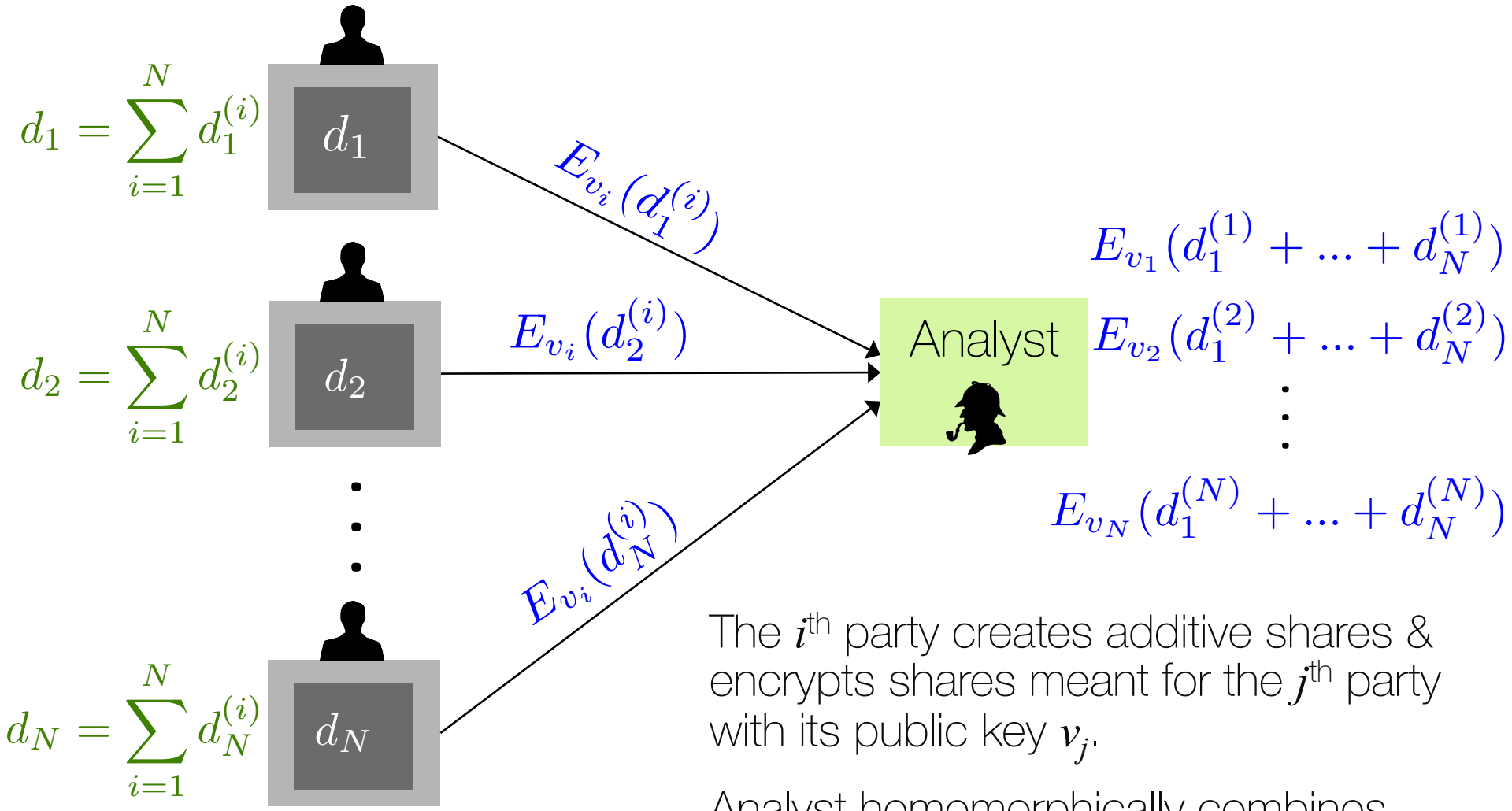
Additively blinded data sent to analyst.

Analyst cancels out the random numbers by addition.

Exercise: What could go wrong?

[Erkin & Tsudik, 2012]

Soln #6: Secret Sharing + Homomorphic Encryption

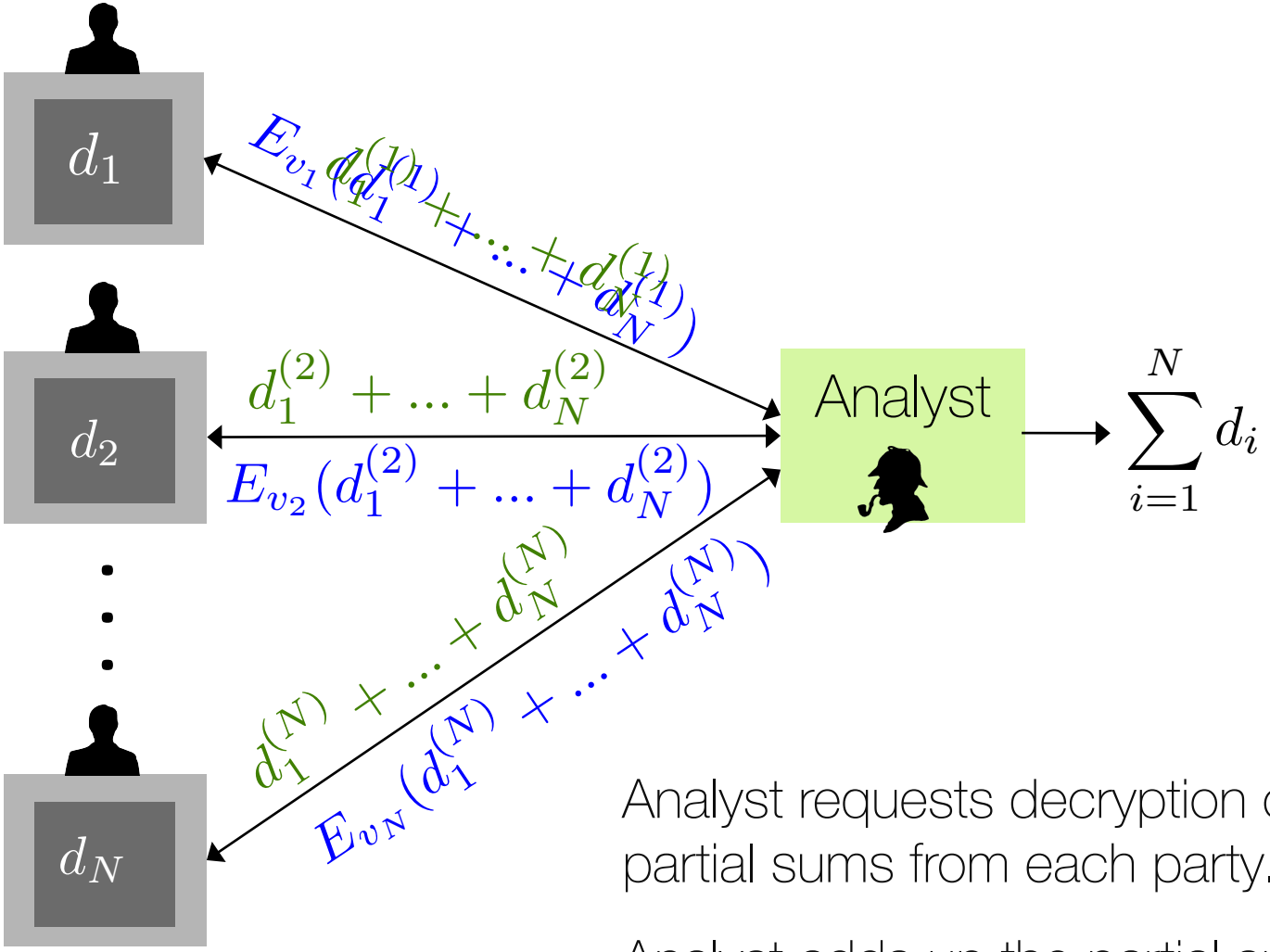


The i^{th} party creates additive shares & encrypts shares meant for the j^{th} party with its public key v_j.

Analyst homomorphically combines the shares.

[Garcia & Jacobs, 2011]

Soln #6: Secret Sharing + Homomorphic Encryption (contd)



Analyst requests decryption of the partial sums from each party.

Analyst adds up the partial sums.

[Garcia & Jacobs, 2011]

Complexity and Fault Tolerance

Measure complexity in terms of the number of encryptions, decryptions, and ciphertext computations and transmissions.

Approach	Complexity (parties)	Complexity (aggregator)	Fault Tolerance (Can parties leave?)
Trusted Dealer	O(N)	O(N)	No
Untrusted Collector	O(N)	O(N)	Yes
Untrusted Key Authority	O(N)	O(N)	Yes
Secret Sharing in Setup	O(N)	O(N)	No
Secret Sharing with Leaders	O(N)	$O(N^2)$	No
Secret Sharing with Homomorphic Encryption	$O(N^2)$	$O(N^2)$	No

Which approach should you prefer?

What is the **complexity** at the participants?

What is the **complexity** at the aggregator?

Is the scheme **fault-tolerant**, i.e., can the parties go off-line after providing their data?

How many **rounds of communication** are needed?

Are **inter-participant connections** possible? expensive?

Are **new participants**, e.g., collector, key authority practical?

What about **differential privacy**? How do you achieve it?

6

SOLUTIONS COMPARED

Private Set Intersection

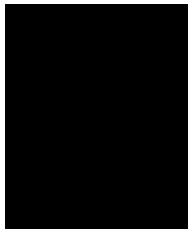
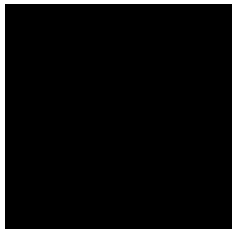
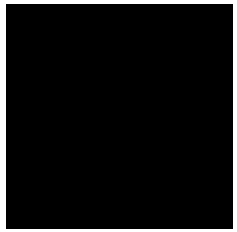
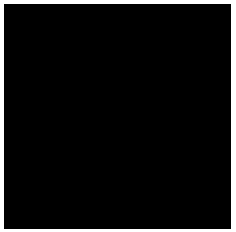
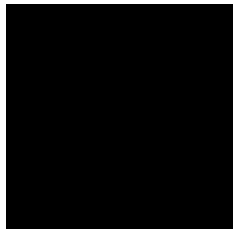
Homomorphic encryption

Case Study:
Data Quality Assessment

First Name	Last Name	Age	State	ZIP
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]

$

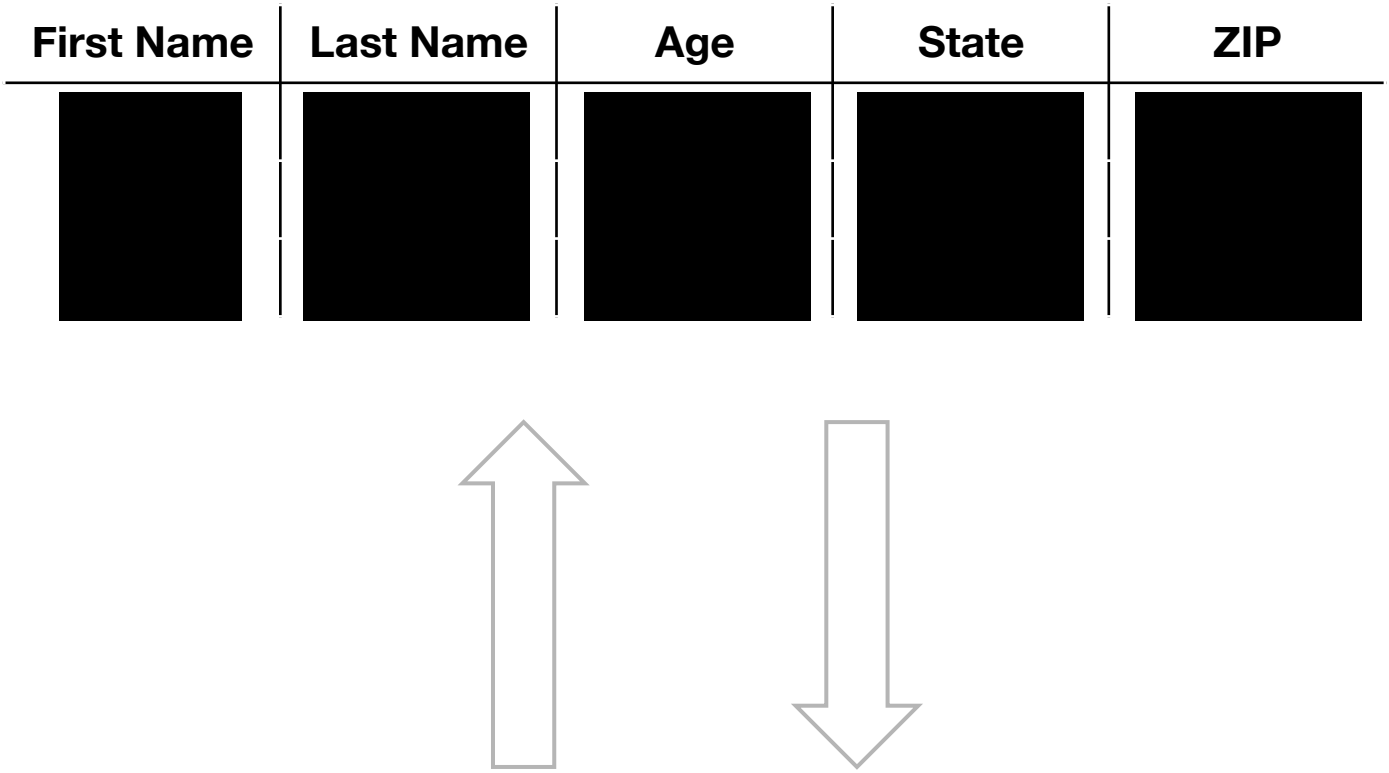
Customer care analytics

First Name	Last Name	Age	State	ZIP
				

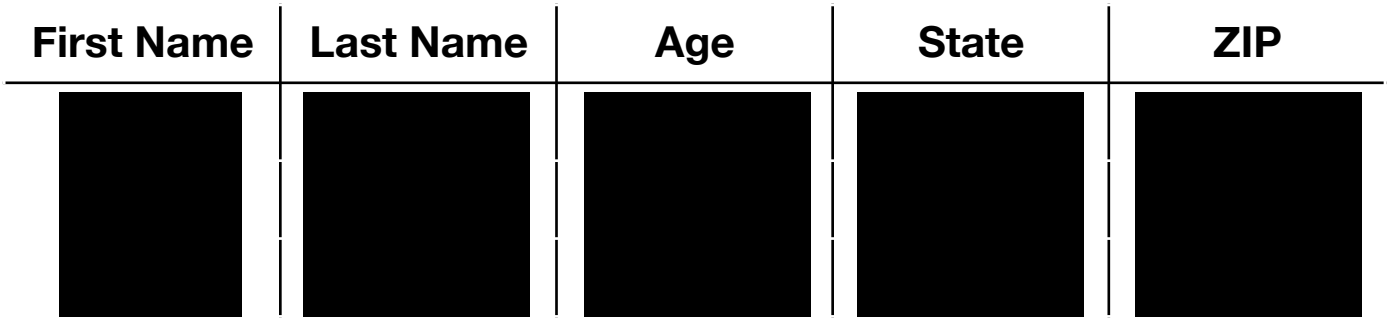
What about the data quality?

Customer does not know quality of data prior to purchase

Data cleaning accounts for up to 80% of development time in big data projects



Privacy concerns for server



How many rows are complete?

All of them

Trust and privacy concerns for client

Data Quality Metrics

First Name	Last Name	Age	State	ZIP
John	Steinbeck	32	CA	94043
Jimi	Hendrix	27	WA	01000
Isaac	Asimov	-15	NY	NULL

Completeness

Percentage of elements that are properly populated

Check for values such as NULL, “”,…

Data Quality Metrics

First Name	Last Name	Age	State	ZIP
John	Steinbeck	32	CA	94043
Jimi	Hendrix	27	WA	01000
Isaac	Asimov	-15	NY	NULL

Validity

Percentage of elements whose attributes possess meaningful values

Data Quality Metrics

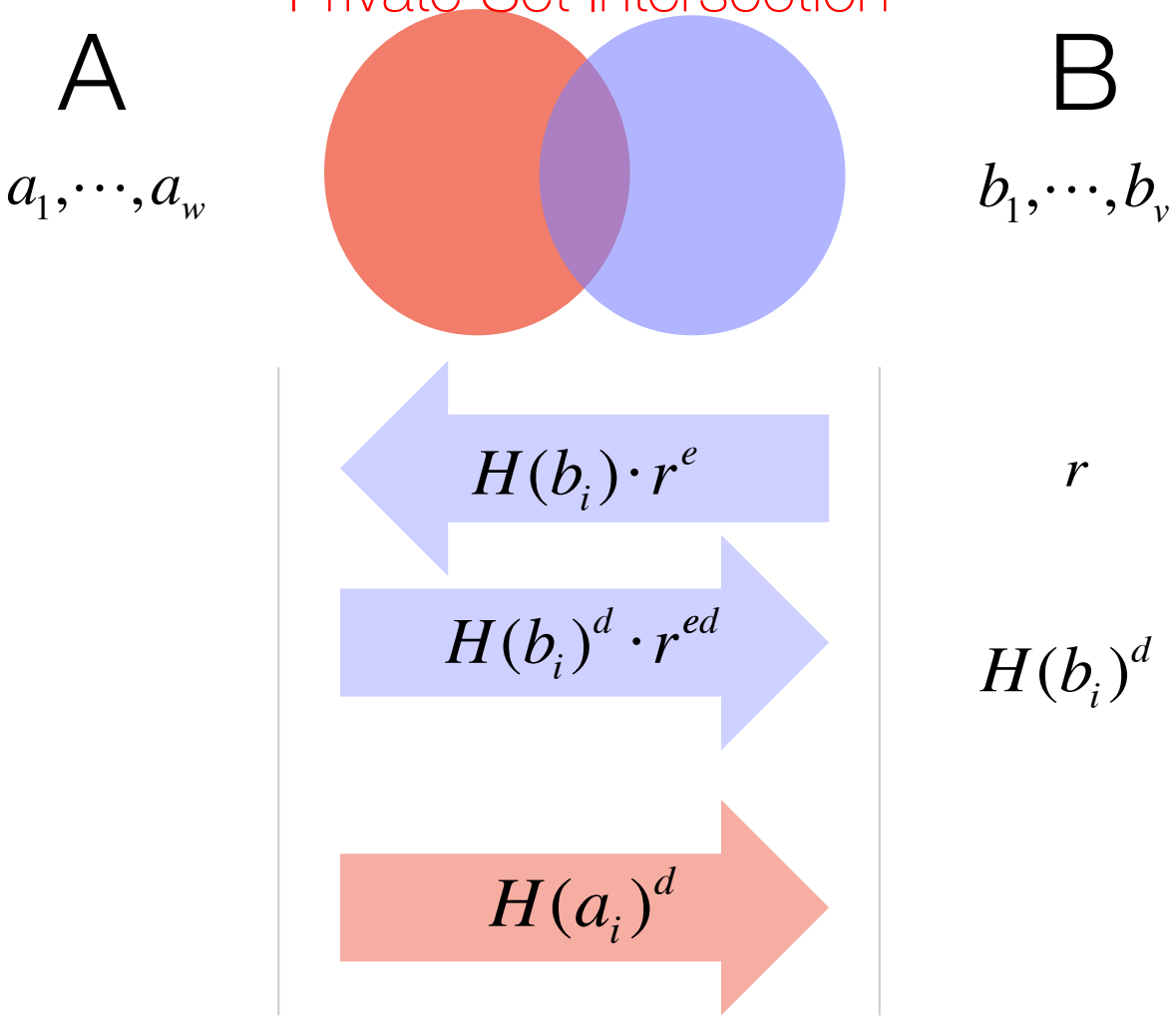
First Name	Last Name	Age	State	ZIP
John	Steinbeck	32	CA	94043
Jimi	Hendrix	27	WA	01000
Isaac	Asimov	-15	NY	NULL

Consistency

Degree to which the data attributes satisfy a dependency constraints

Candidate Solution:

Private Set Intersection

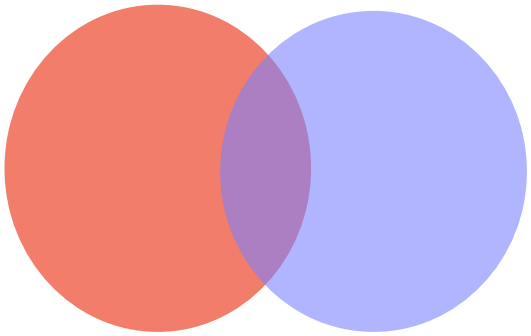


Set intersection or cardinality of set intersection

Private Set Intersection

Completeness

Client	Server
{NULL}	$\{d_1, \ldots, d_n\}$
1, NULL 2, NULL … n, NULL	1, d_1 2, d_2 … n, d_n



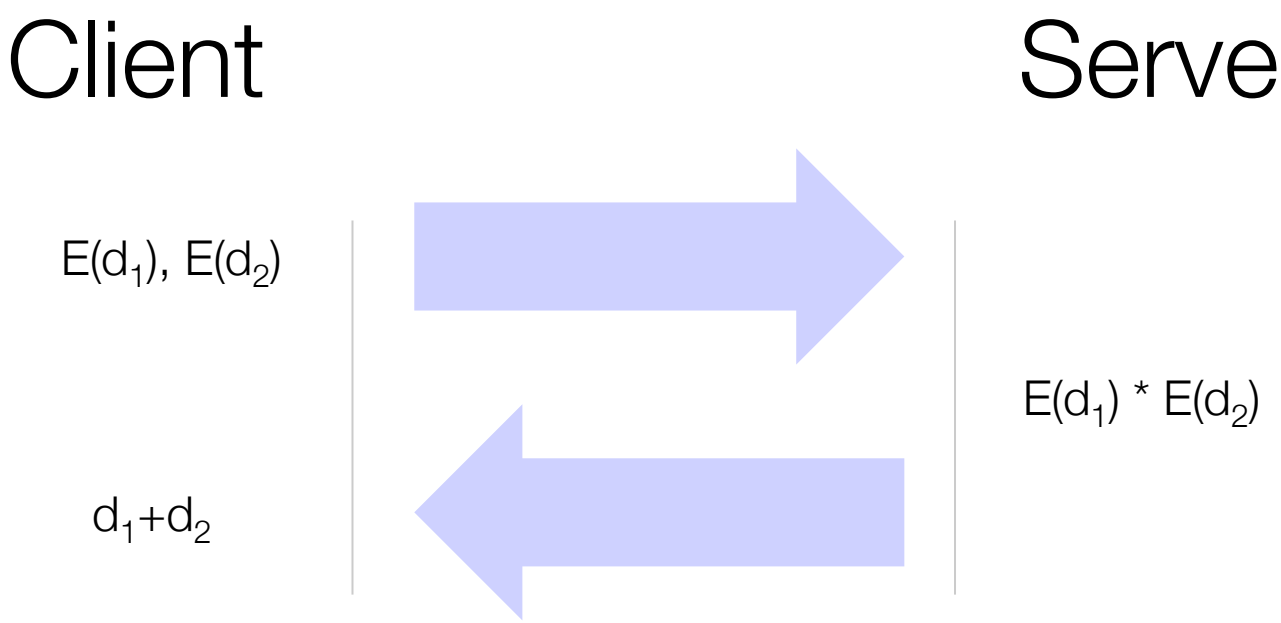
= 7

PSI-CA approach is inefficient

Encrypted-domain Computation

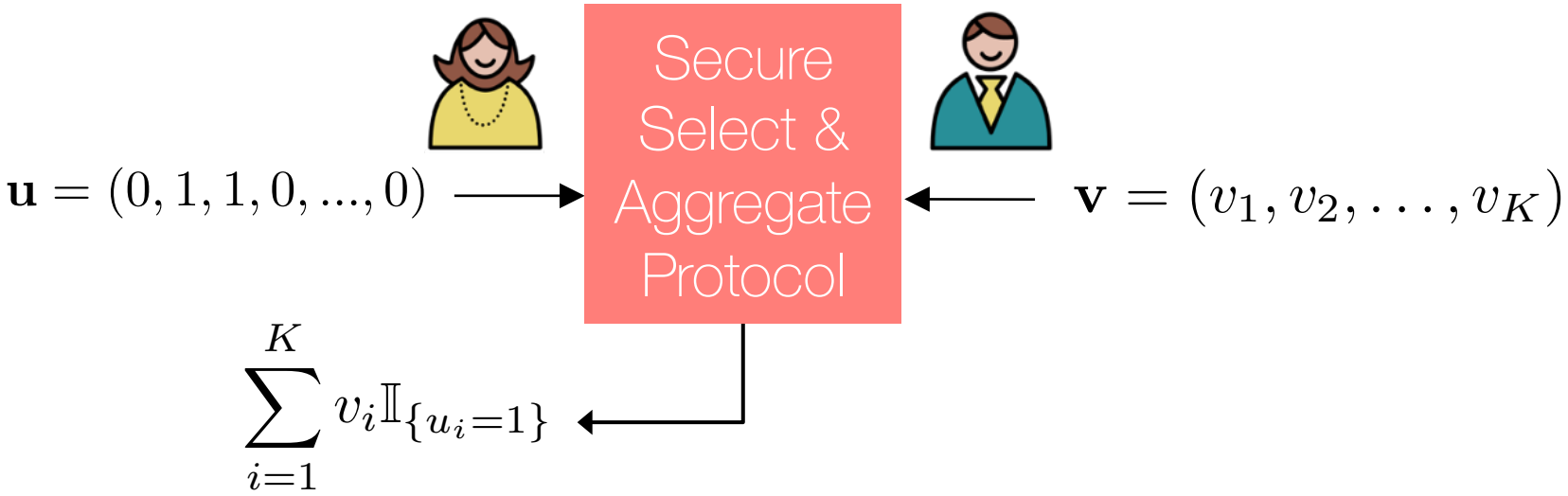
$$E(d_1) \cdot E(d_2) = E(d_1 + d_2)$$

$$E(d_1)^{d_2} = E(d_1 \cdot d_2)$$



[Paillier 99, Damgard-Jurik 01]

Select & Aggregate Setup

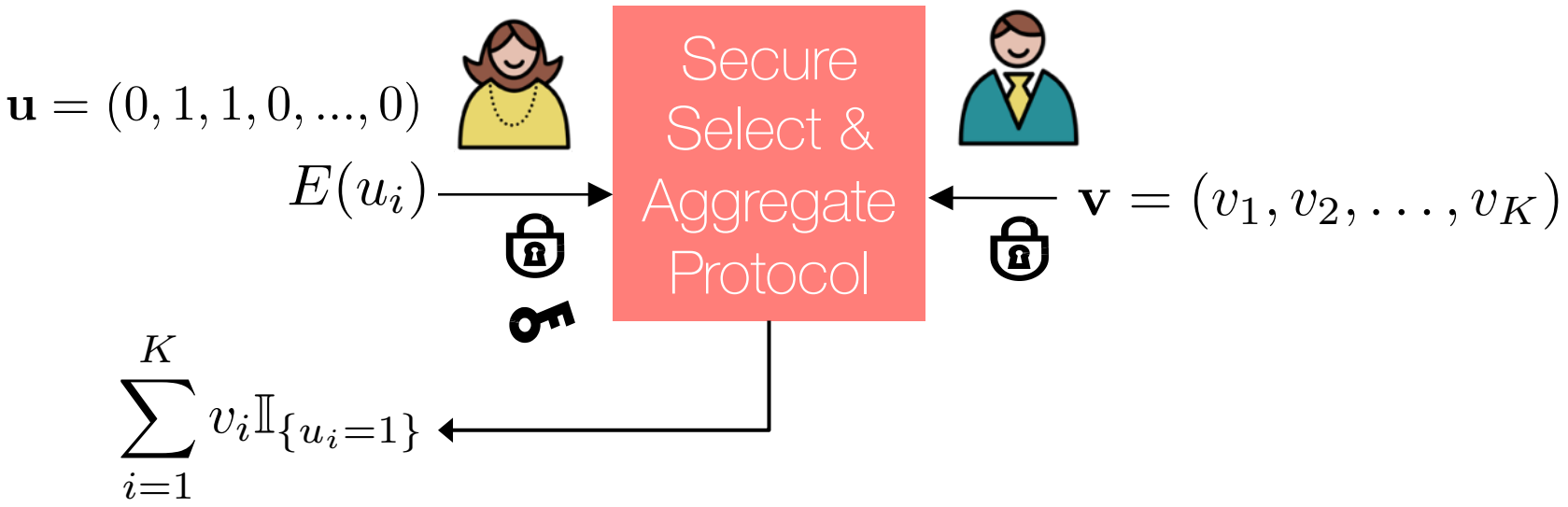


Goal: Alice has a binary selector $\mathbf{u}$, Bob has data vector $\mathbf{v}$. Alice should discover the sum of selected elements from $\mathbf{v}$.

Query Privacy: Bob should not find the selector vector.

Data Privacy: Alice should not discover any information other than the selected aggregate.

Select & Aggregate Protocol



1. Alice sends element-wise encryptions of $\mathbf{u}$ to Bob.
2. Bob computes the dot product of $\mathbf{u}$ and $\mathbf{v}$ using additive homomorphic property, and sends it to Alice.

$$\prod_{i=1}^{K} E(u_i)^{v_i} = \prod_{i=1}^{K} E(u_i v_i) = E\left(\sum_{i=1}^{K} v_i u_i\right) = E\left(\sum_{i=1}^{K} v_i \mathbb{I}_{\{u_i\}}\right)$$

3. Alice decrypts the dot product.

Select & Aggregate Complexity

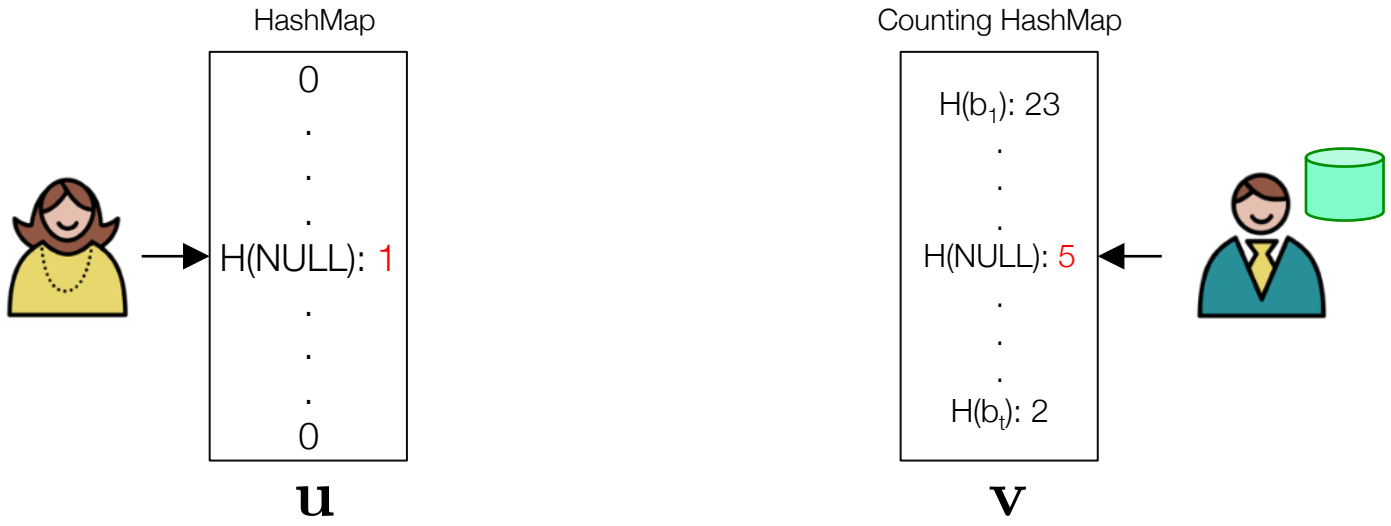
		
# Encryptions	K	0
# Decryptions	1	0
# Multiplications	0	K
# Exponentiations	0	K
# Transmissions	K	1

Cannot afford O(#tuples) complexity for large databases.

Key Idea

1. Find a suitable low-dimensional representation.
2. Use Select & Aggregate to evaluate quality metric.

Completeness Setup



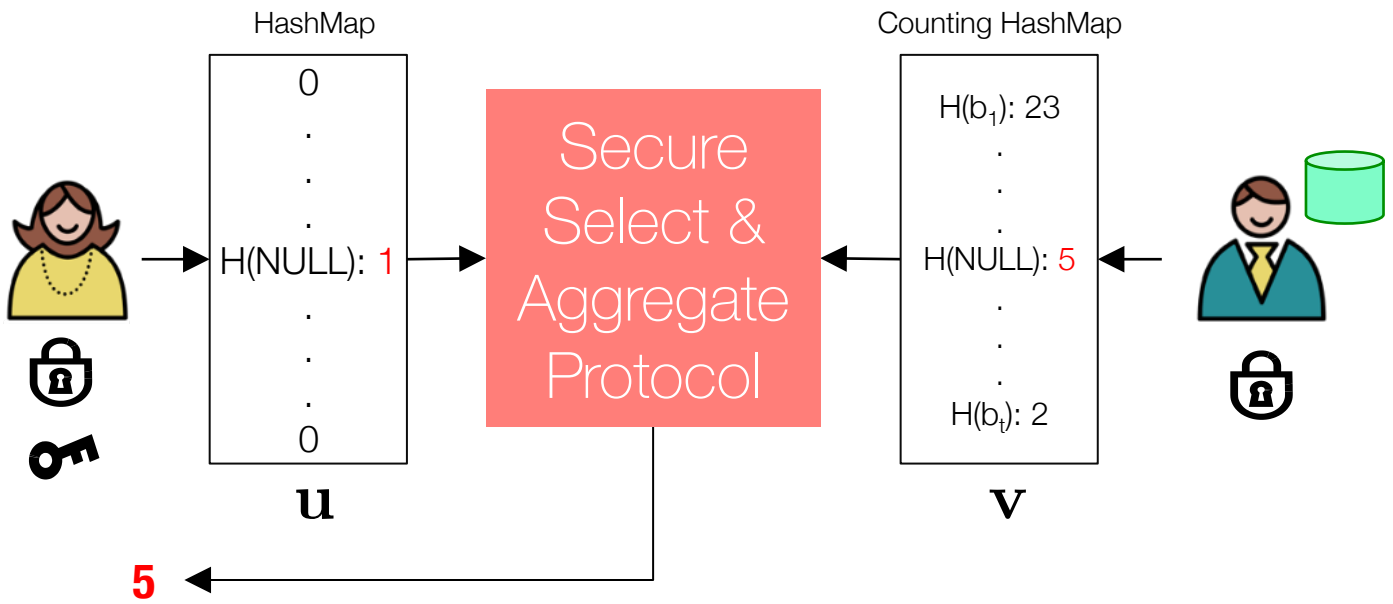
Example: Alice wants to find the number of NULL values in Bob's data.

Query Privacy: Bob does not discover that Alice is searching for the number of NULLs.

Data Privacy: Alice discovers nothing else about Bob's data.

Trick: Alice generates a Hashmap, Bob generates a Counting Hashmap.

Completeness Protocol

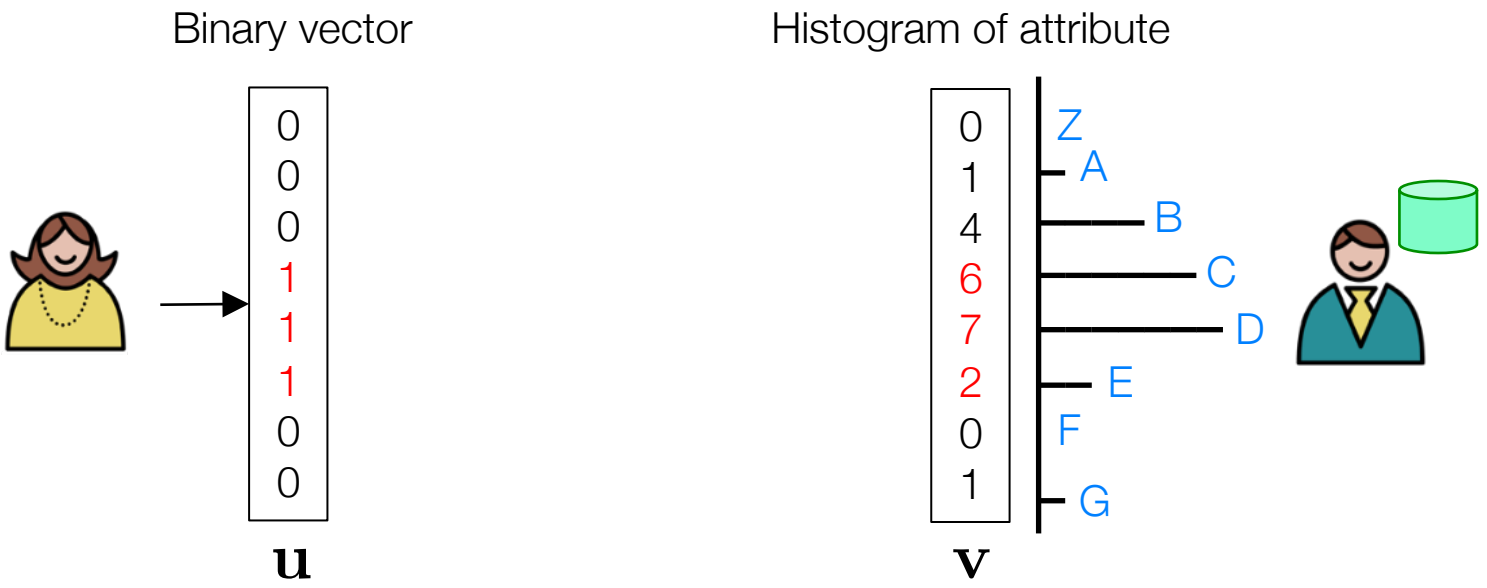


Alice generates public encryption key and private decryption key for additively homomorphic cryptosystem.

The parties evaluate Select & Aggregate on Alice's Hashmap and Bob's Counting Hashmap.

By construction, protocol reveals number of NULLs to Alice.

Validity Evaluation Setup



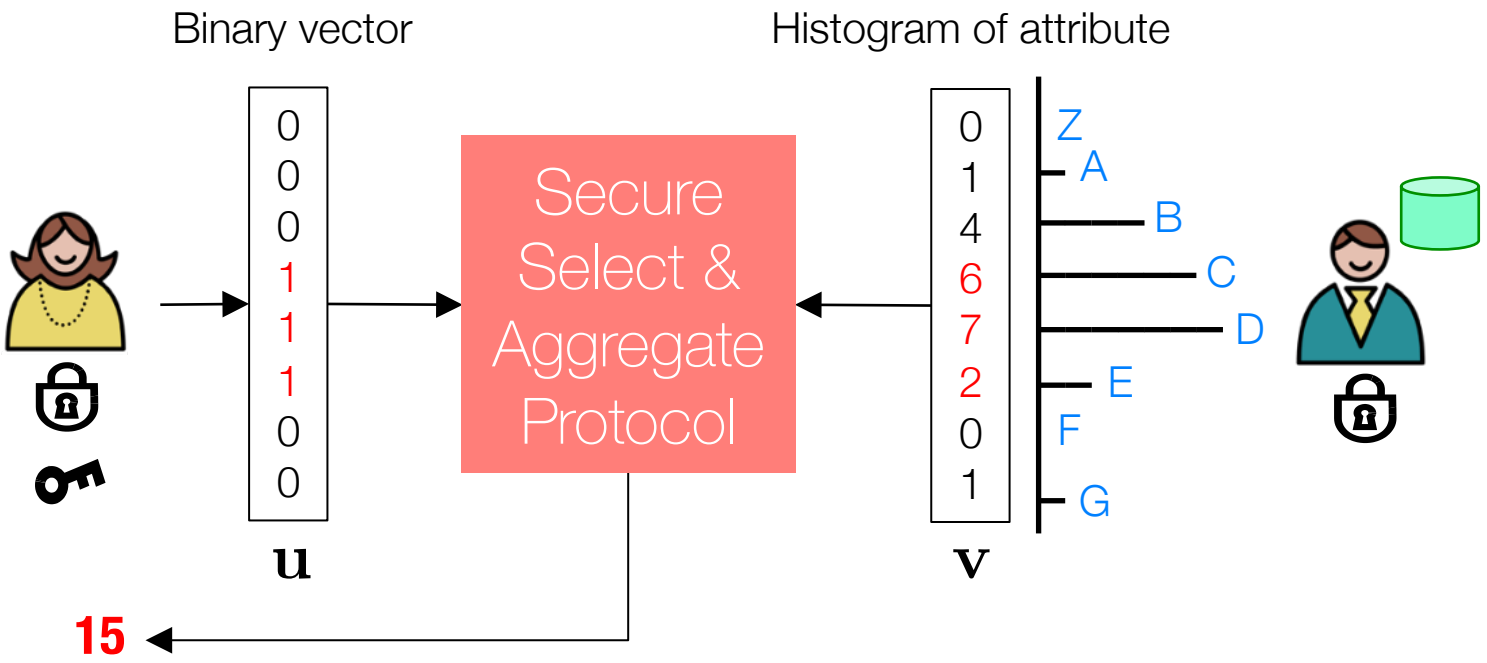
Example: Alice wants to know how many of Bob's entries are in the range [C,E].

Query Privacy: Bob does not discover the range of Alice's searches.

Data Privacy: Alice discovers nothing else about Bob's data.

Trick: Bob generates a histogram vector, Alice generates a binary selector vector on the support of the histogram.

Validity Evaluation Protocol

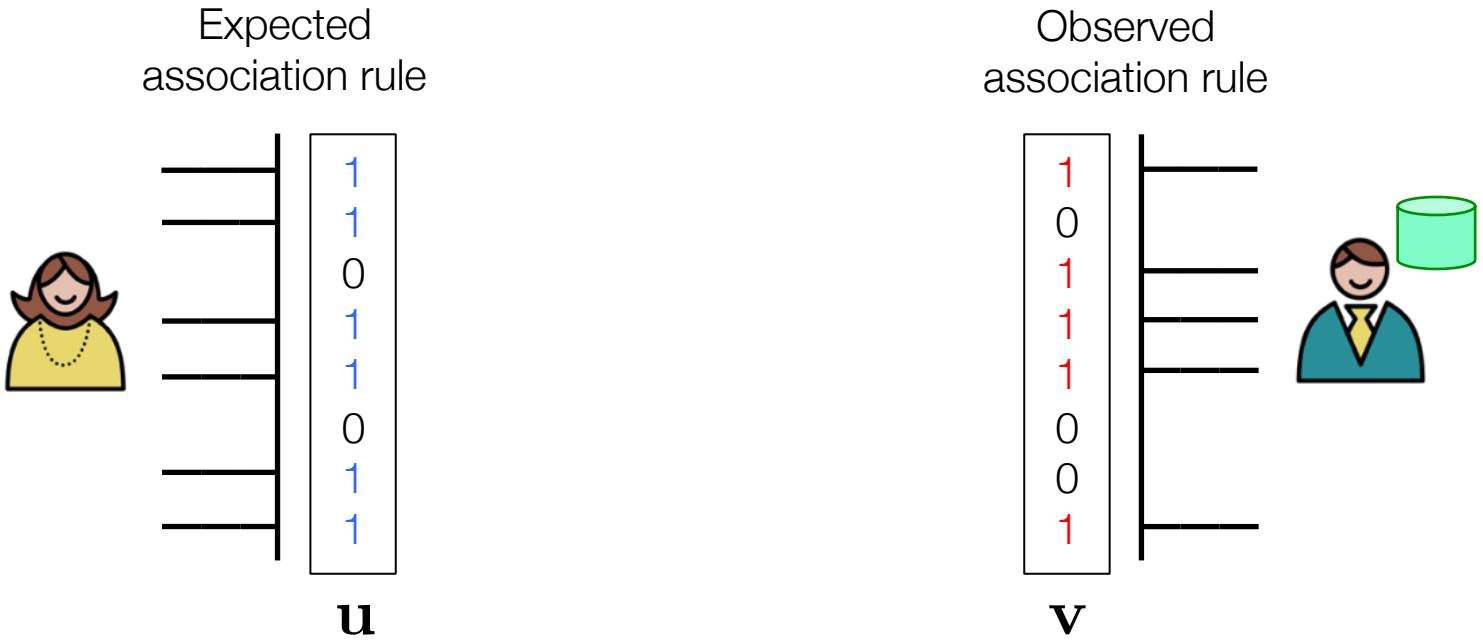


As before, Alice and Bob run the Select & Aggregate protocol on Alice's selector vector and Bob's histogram.

By construction, protocol reveals number of "valid" values to Alice.

Protocol works for arbitrary range queries.

Consistency Evaluation Setup



Example: Alice wants to know how many of Bob's entries follow correct dependencies among attributes, e.g., State – Zipcode.

Query Privacy: Bob doesn't discover which dependencies Alice is checking.

Data Privacy: Alice discovers nothing else about Bob's data.

Trick: Bob generates a vector of observed associations, Alice generates a

	CA	MA	MN	…
94304	1	0	0	0
55414	0	0	1	0
02139	0	1	0	0
94305	1	0	0	0
…				

Desired Dependencies

	CA	MA	MN	…
94304	0	0	1	0
55414	0	0	1	0
02139	0	1	0	0
94305	1	0	0	0
…				

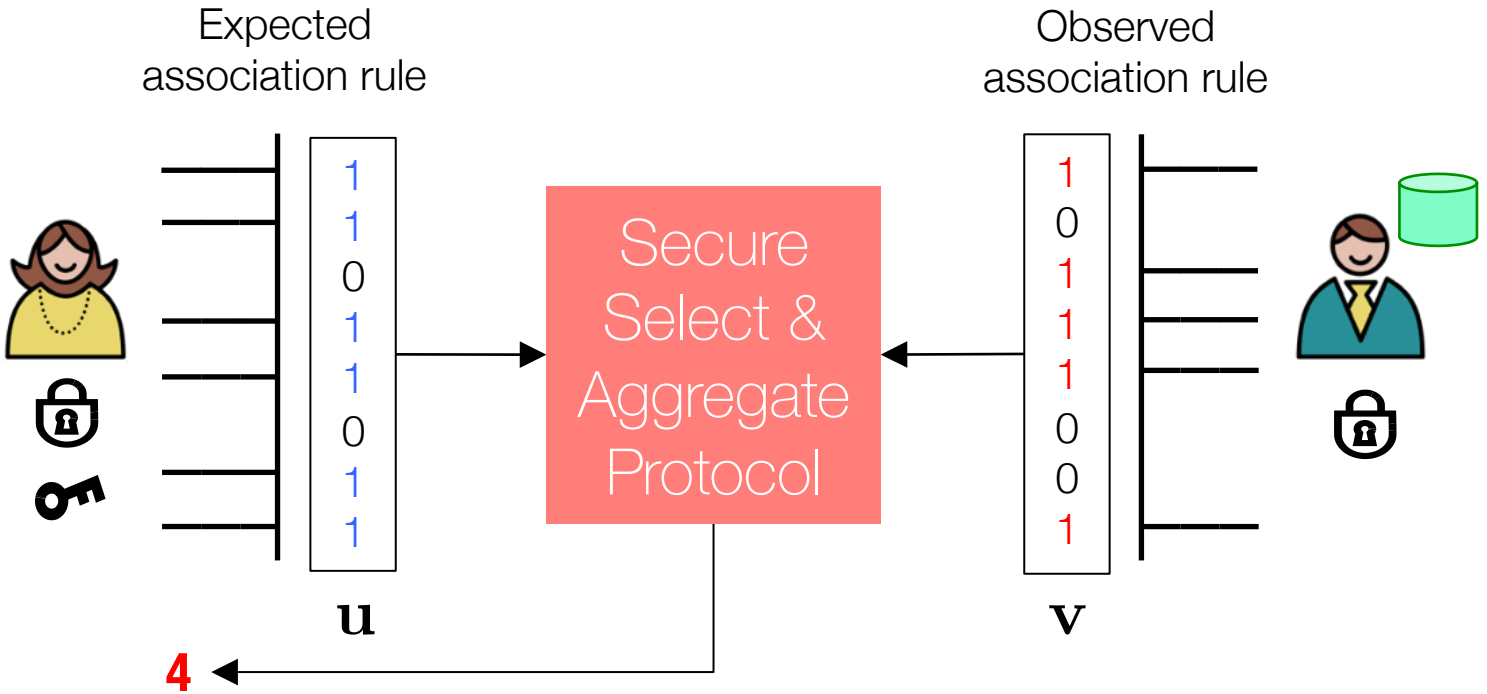
Observed Dependencies

Alice and Bob agree upon an ordering of attribute values.

They also agree on a vectorization (flattening) pattern.

Need to securely compute how many of Bob's dependencies are consistent with Alice's rules.

Consistency Evaluation Protocol



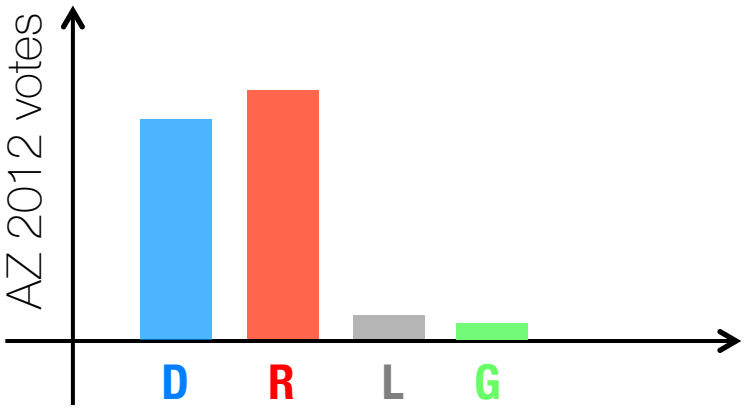
Alice and Bob run the Select & Aggregate protocol on Alice's desired rule vector and Bob's observed rule vector.

Protocol reveals number of "valid" dependencies to Alice.

Works for dependencies among arbitrary attribute combinations.

Computational Complexity

Metrics	Proposed Protocols	Using PSI-CA
Completeness	O(# uniques)	O(# tuples)
Validity Timeliness Uniqueness	O(# histogram bins)	O(# tuples)
Consistency	O((# histogram bins)^m)	O((# tuples)^m)



uniques = # bins = 4

tuples = 2,306,559

7

Research Directions

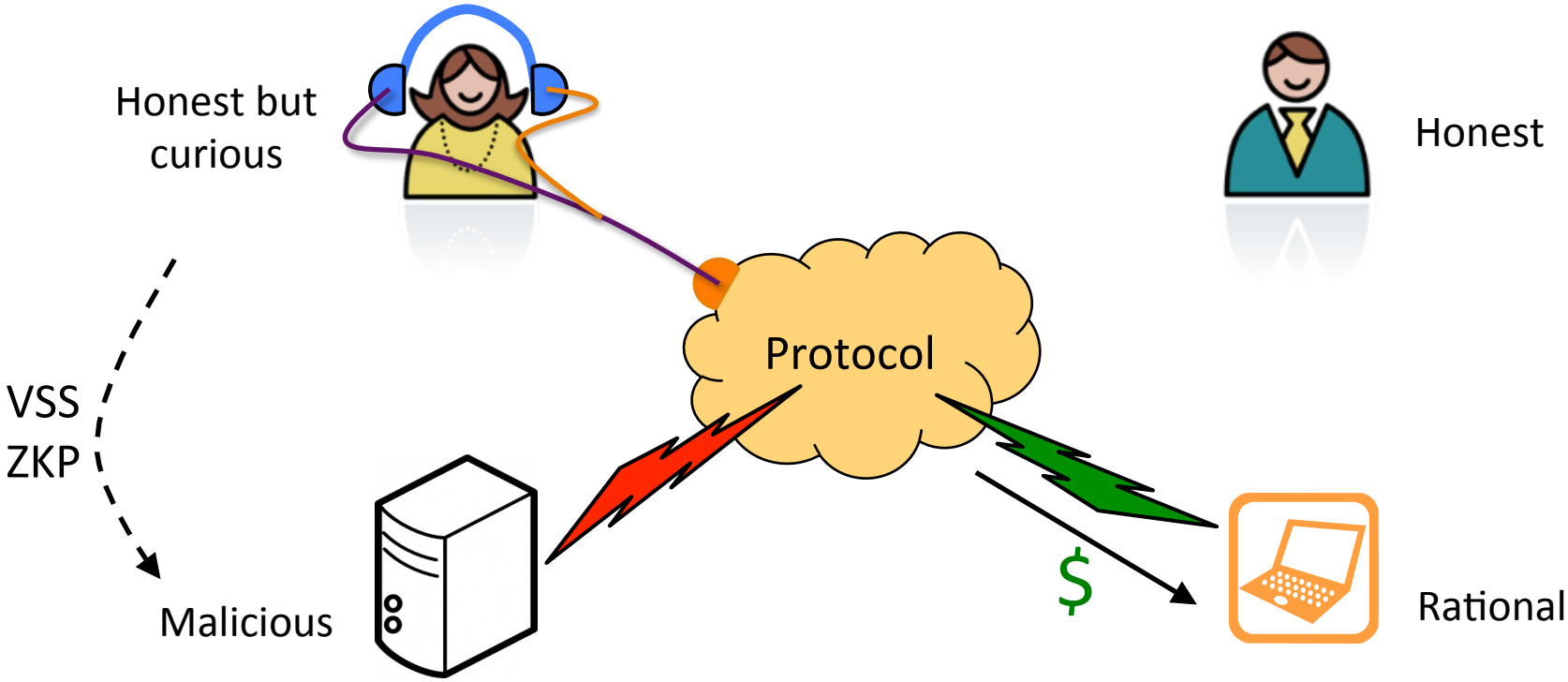
Expert systems

Privacy policy automation

Interdisciplinary perspectives

Verifiable computing

Establishing good adversarial models



Semi-honest assumption is often too lenient.

Malicious assumption is often intractable and unnecessary.

Is there something reasonable in between?

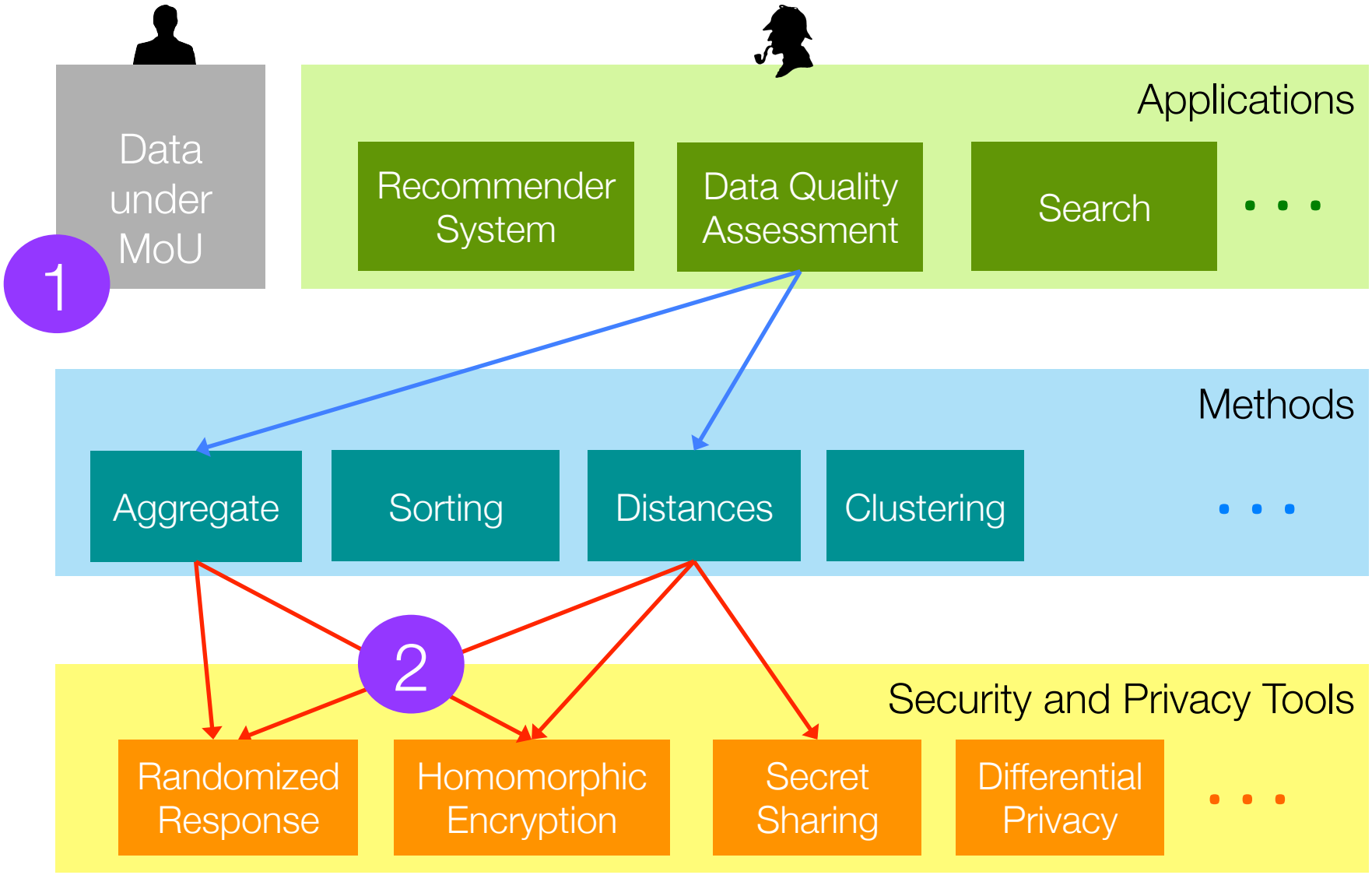
Establishing good adversarial models

Rational Adversaries

Adversarial Signal Processing

Combinations of differential privacy and crypto

How We Achieve Privacy Today



Two Main Directions

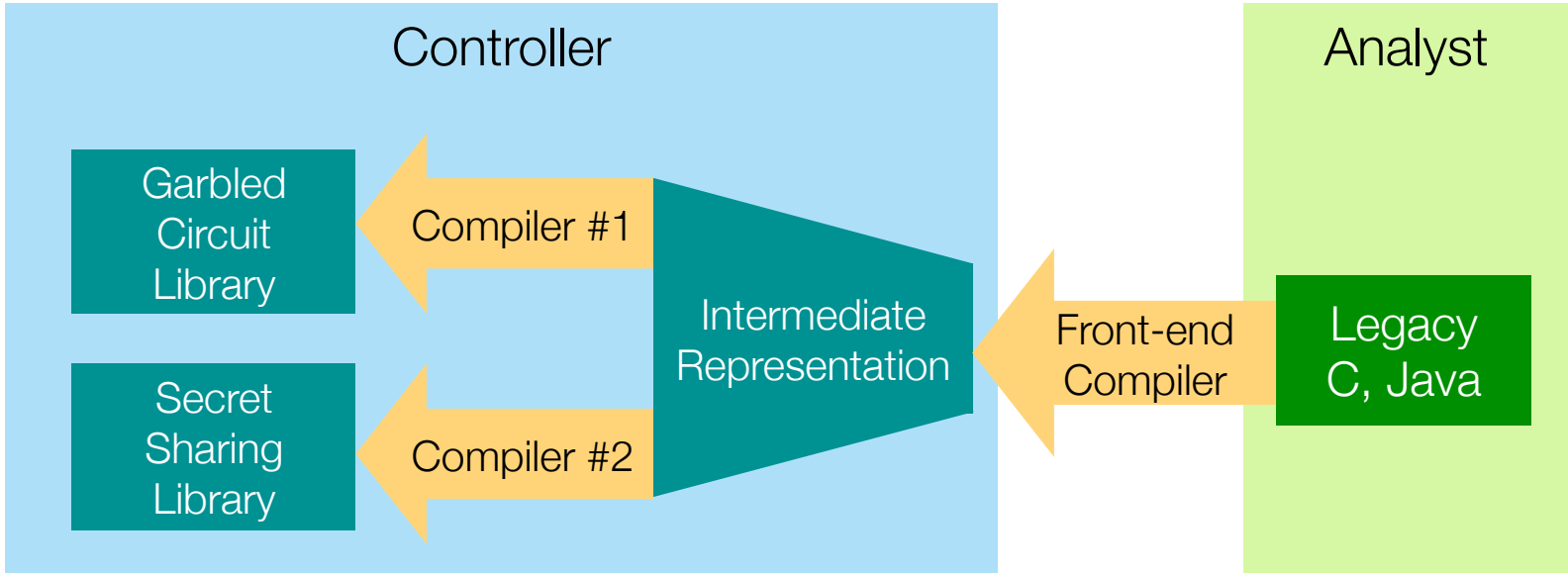
People-facing perspective

- Help people understand privacy implications
- Help them express privacy preferences
- (Really) Give people more control over their privacy

Analyst-facing perspective

- Automate selection & composition of privacy primitives
- Support legacy machine learning approaches
- Improve privacy-utility tradeoffs (by a lot!)

Support for Legacy Analytics

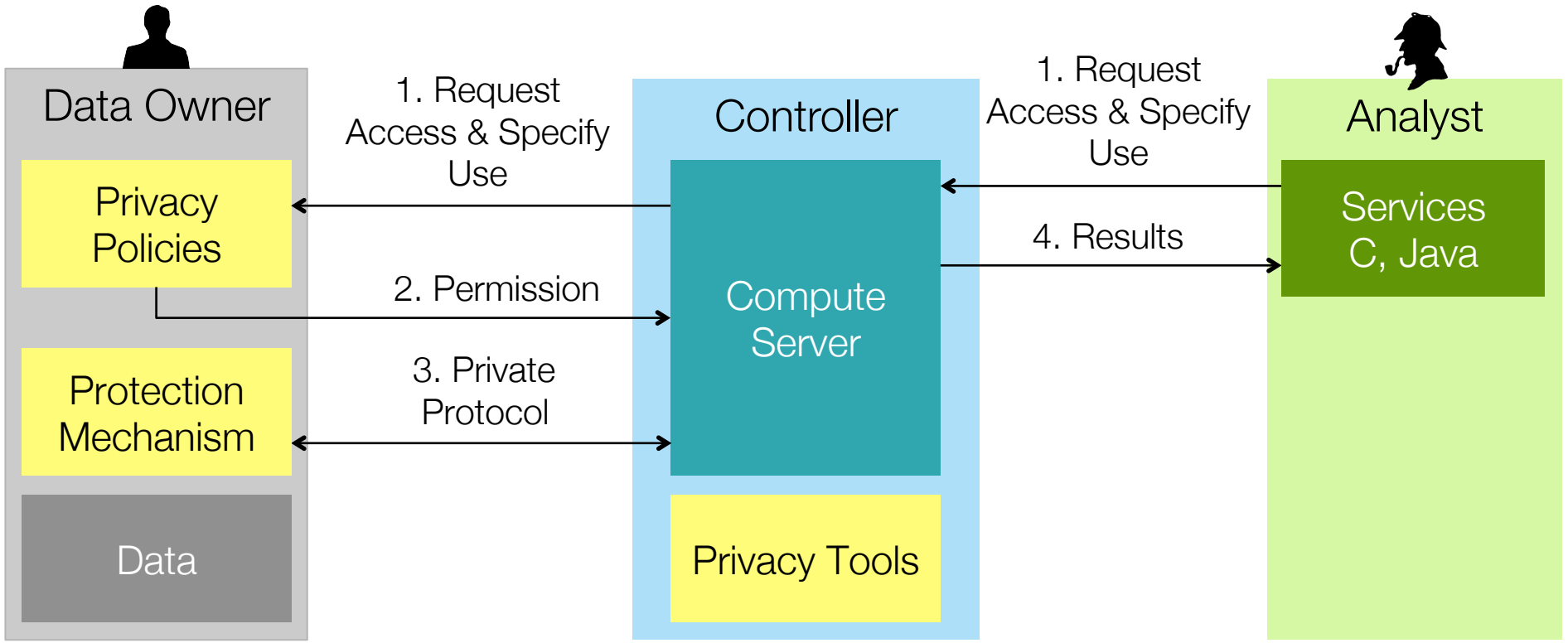


Advantages

- Plug-and-play interoperability for privacy-preserving primitives.
- Leverage large body of existing code.
- Support for non-specialist programmers.

Examples: ObliVM [Liu, Hicks, Shi, 2013-14]

Data Transaction: Personal Privacy



Match users' requests for data against owners' privacy policies.

Rewrite analytics programs using one or more privacy tools.

Update policies using feedback from previous computations.

New (or Repurposed) Combinations

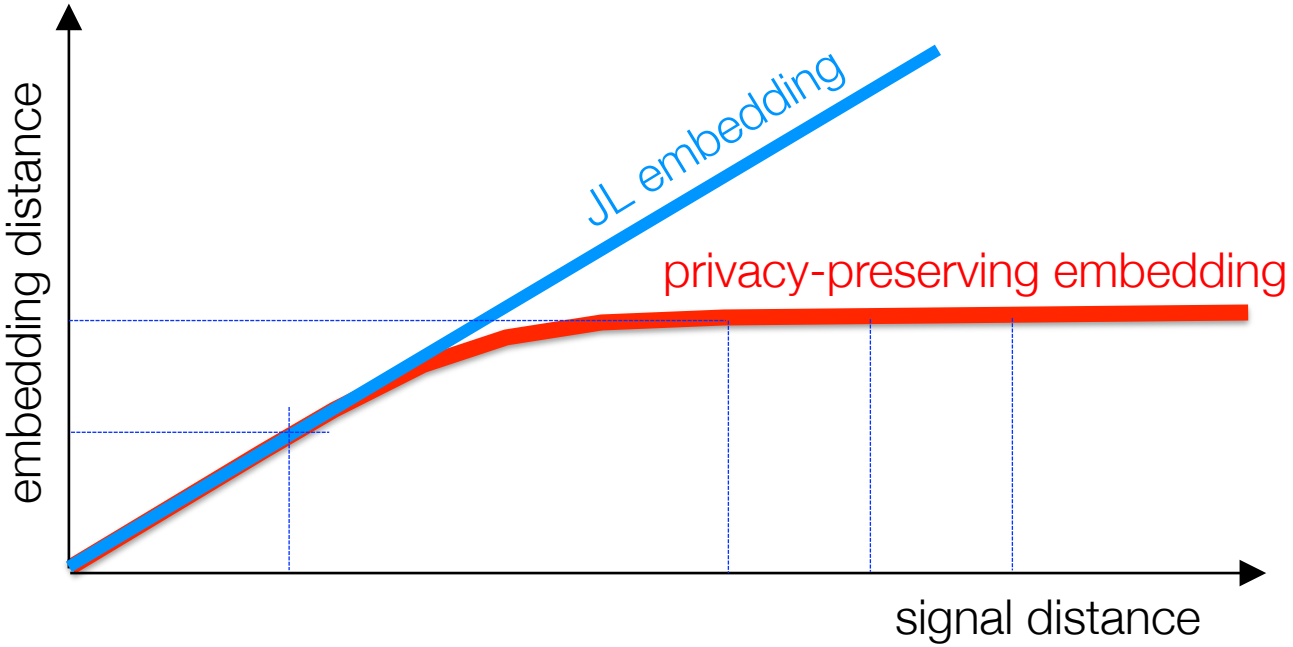
Use statistical or cryptographic primitives with machine learning and signal processing techniques.

Must be better than just combining the gains and removing the flaws of each technique?

Examples:

- Low-dimensionality embeddings + encrypted-domain clustering
- Sampling + differentially private mechanisms
- ECC + cryptographic hashes (Fuzzy Vaults)

Embeddings + Encryption



Restrict computations to pairwise distances.

Reduced dimensionality → reduced ciphertext complexity.

Some embeddings are themselves privacy preserving. Eliminate or reduce need for encrypted-domain interaction. [Boufounos, Rane, 11]

Sampling + Randomization

Not private, more samples reveal more information

Need too much noise for privacy, so utility is poor.

Fewer samples, so need less noise for sufficient privacy.
[Li, Qardaji, Su, 2011] [Lin, Wang, Rane, 2013]

Some Perspectives

An interesting research area, with growing awareness and investment from industry and governments alike.

We need ways to automatically select privacy primitives according to their privacy-utility-efficiency tradeoffs.

It's not just all about crypto. Significant contributions to this field could be via:

- Statistics (information theory)
- Signal Processing and Digital Communications
- Game Theory
- Domain-specific languages and formal methods
- Human, ethnographic studies

Thanks!

Acknowledgments

Julien Freudiger
Alejandro Brito
Ersin Uzun — PARC

Petros Boufounos — MERL

References (Secure Aggregation)

Z. Erkin, J. R. Troncoso-Pastoriza, R. Lagendijk, and F. Perez-Gonzalez. Privacy-preserving data aggregation in smart metering systems: An overview. *Signal Processing Magazine, IEEE*, 30(2):75–86, 2013.

E. Shi, T-H. H. Chan, E. Rieffel, R. Chow, and D. Song. Privacy- preserving aggregation of time-series data. In *NDSS*, volume 2, page 4, 2011.

T-H. H. Chan, E. Shi, and D. Song. Privacy-preserving stream aggregation with fault tolerance. In *Financial Cryptography and Data Security*, pages 200–214. 2012.

I. Bilogrevic, J. Freudiger, E. De Cristofaro, and E. Uzun. What's the gist? privacy-preserving aggregation of user profiles. In *Computer Security-ESORICS 2014*, pages 128–145. 2014.

M. Jawurek and F. Kerschbaum. Fault-tolerant privacy-preserving statistics. In *Privacy Enhancing Technologies*, pages 221–238, 2012

I. Leontiadis, R. Molva, and M. Onen. Privacy preserving statistics in the smart grid. In *Distributed Computing Systems Workshops (ICDCSW), 2014 IEEE 34th International Conference on*, pages 182– 187,2014.

M. Joye and B. Libert. A scalable scheme for privacy-preserving aggregation of time-series data. In *Financial Cryptography and Data Security*, pages 111–125. 2013.

References (Secure Aggregation)

Z. Erkin and G. Tsudik. Private computation of spatial and temporal power consumption with smart meters. In *Applied Cryptography and Network Security*, pages 561–577, 2012.

G.Acs and C.Castelluccia. I have a DREAM! (differentially private smart metering). In *Information Hiding*, pages 118–132, 2011.

K. Kursawe, G. Danezis, and M. Kohlweiss. Privacy-friendly aggre- gation for the smart-grid. In *Privacy Enhancing Technologies*, pages 175–191, 2011.

F. Garcia and B. Jacobs. Privacy-friendly energy-metering via homo- morphic encryption. In *Security and Trust Management*, pages 226–238. 2011.

References (Data Quality Assessment and CryptDB)

Thomas C Redman. The impact of poor data quality on the typical enterprise. *Communications of the ACM*, 41(2), 1998.

Wayne W Eckerson. Data quality and the bottom line. *TDWI Report, The Data Warehouse Institute*, 2002.

Yang W Lee, Diane M Strong, Beverly K Kahn,and Richard Y Wang. AIMQ: a methodology for information quality assessment. *Information & management*, 40(2), 2002.

Health insurance portability and accountability act of 1996. *Public Law*, 104:191, 1996.

Phillip Cykana, Alta Paul, and Miranda Stern. DoD Guidelines on Data Quality Management. In *IQ*, pages 154–171, 1996

Freudiger, J., Rane, S., Brito, A. E., & Uzun, E. (2014, November). Privacy Preserving Data Quality Assessment for High-Fidelity Data Sharing. In *Proceedings of the 2014 ACM Workshop on Information Sharing & Collaborative Security* (pp. 21-29). ACM.

Popa, R. A., Zeldovich, N., & Balakrishnan, H. (2011). CryptDB: A practical encrypted relational DBMS.

References (Intermediate Representations)

Memory Trace Oblivious Program Execution. Chang Liu, Mike Hicks, and Elaine Shi. Computer Security Foundations Symposium (CSF), 2013.

Automating Efficient RAM-Model Secure Computation. Chang Liu, Yan Huang, Elaine Shi, Jonathan Katz, and Michael Hicks. IEEE Symposium on Security and Privacy (S&P), 2014.

Oblivious Data Structures. Xiao Shaun Wang, Kartik Nayak, Chang Liu, T-H. Hubert Chan, Elaine Shi, Emil Stefanov, and Yan Huang. In ACM Conference on Computer and Communications Security (CCS), 2014.

Liu, Chang, Xiao Shaun Wang, Kartik Nayak, Yan Huang, and Elaine Shi. "Oblivm: A programming framework for secure computation." (2015).