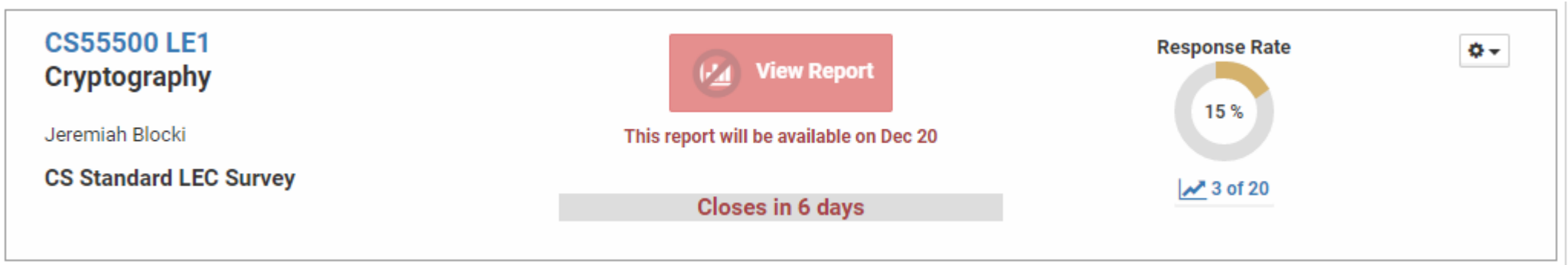
Homework 5 Statistics

Minimum Value	**40**
Maximum Value	**110.00**
Range	**110.00**
Average	**78.95**
Median	**95.00**
Standard Deviation	**31.79**

Course Business

- Please Complete Your Course Evaluations
- Your feedback is valuable!



Final Exam

- Time: Tuesday, December 12th at 1PM
- Location: LWSN 1106

- Comprehensive
  - …but heavier coverage of material covered in second half of semester
- Format
  - Multiple choice
  - Fill in the blank
  - true/false/more information

- Solutions to practice exam distributed on Thursday

Cryptography
CS 555

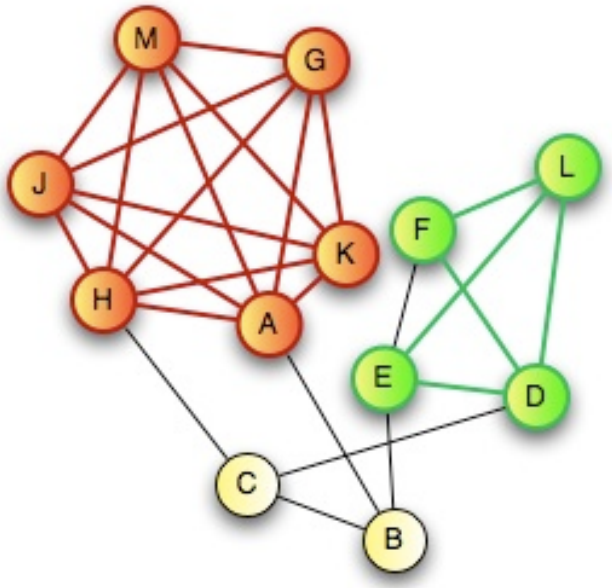
Week 16:

- Zero-Knowledge Proofs,
- Hot Topics in Cryptography
- Review for Final Exam

Readings: Katz and Lindell Chapter 10 & Chapter 11.1-11.2, 11.4

CS 555:Week 15: Zero-Knowledge Proofs

Zero-Knowledge Proof for all NP

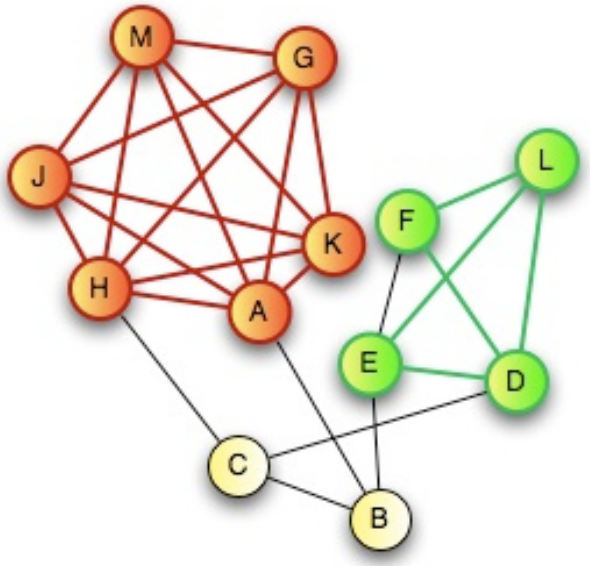


- CLIQUE
  - Input: Graph G=(V,E) and integer k>0
  - Question: Does G have a clique of size k?

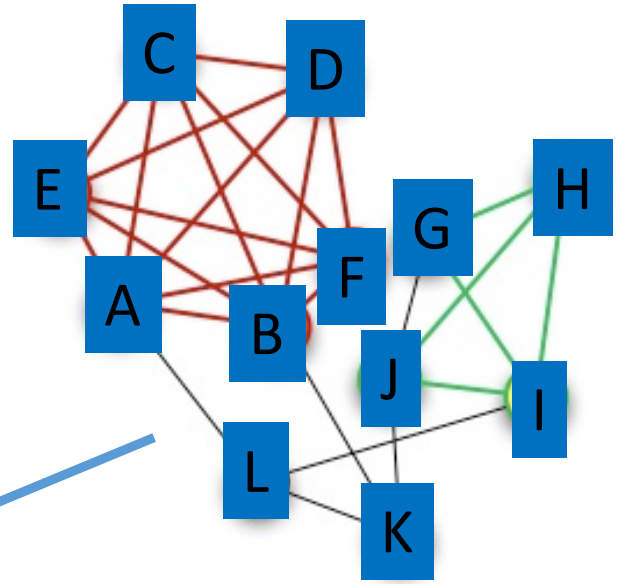
- CLIQUE is NP-Complete
  - Any problem in NP reduces to CLIQUE
  - A zero-knowledge proof for CLIQUE yields proof for all of NP via reduction

- Prover:
  - Knows k vertices $v_1,\ldots,v_k$ in G=(V,E) that form a clique

Zero-Knowledge Proof for all NP



$\sigma(G)$



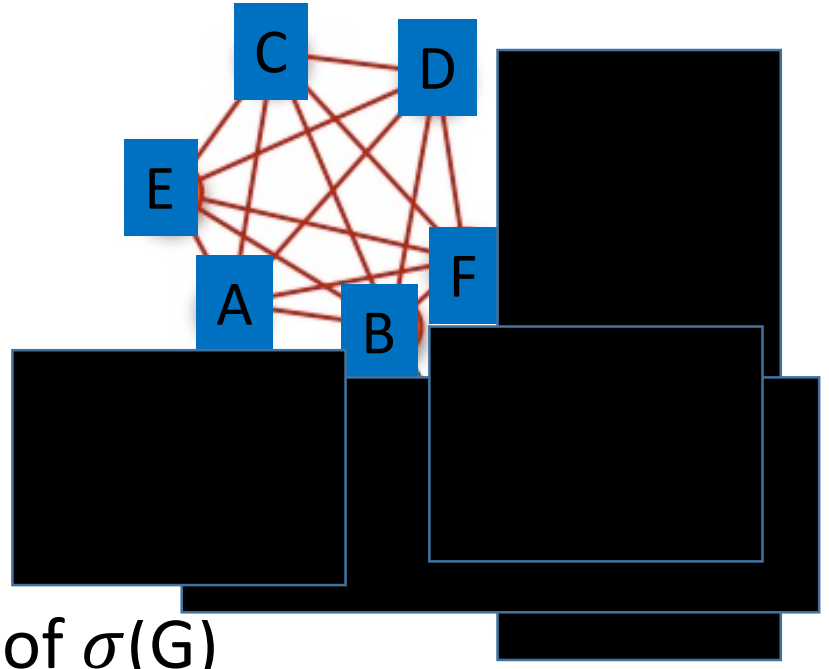
Adjacency matrix $A_{\sigma(G)}$

$$\begin{array}{c} \\ A \\ \\ L \end{array} \begin{array}{c} \begin{array}{ccc} A & \cdots & L \end{array} \\ \begin{pmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{pmatrix} \end{array}$$

Commitment to $A_{\sigma(G)}$

$$\begin{array}{c} \\ A \\ \\ L \end{array} \begin{array}{c} \begin{array}{ccc} A & \cdots & L \end{array} \\ \begin{pmatrix} Com(0, r_{A,A}) & \cdots & Com(1, r_{A,L}) \\ \vdots & \ddots & \vdots \\ Com(1, r_{L,A}) & \cdots & Com(0, r_{L,L}) \end{pmatrix} \end{array}$$

Zero-Knowledge Proof for all NP



- Prover:
  - Knows k vertices $v_1,\ldots,v_k$ in G=(V,E) that for a clique

1. Prover selects a permutation σ over V
2. Prover commits to the adjacency matrix $A_{\sigma(G)}$ of σ(G)
3. Verifier sends challenge c (either 1 or 0)
4. If c=0 then prover reveals σ and adjacency matrix $A_{\sigma(G)}$
   1. Verifier confirms that adjacency matrix is correct for σ(G)
5. If c=1 then prover reveals the submatrix formed by first rows/columns of $A_{\sigma(G)}$ corresponding to $\sigma(v_1), \ldots, \sigma(v_k)$
   1. Verifier confirms that the submatrix forms a clique.

Zero-Knowledge Proof Simulator

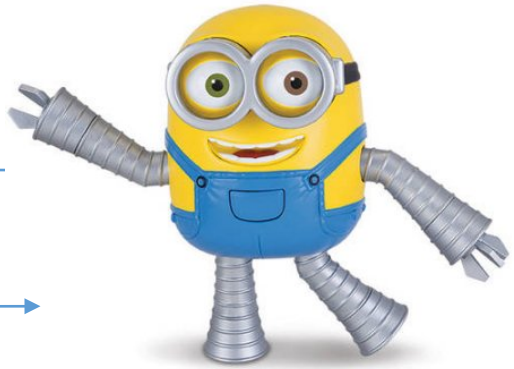
Dishonest (verifier);
$G = (V, E)$,

$$Com(A) = \begin{pmatrix} \mathrm{H}(A_{1,1}, r_{1,1}) & \cdots & \mathrm{H}(A_{1,n}, r_{1,n}) \\ \vdots & \ddots & \vdots \\ \mathrm{H}(A_{n,1}, r_{n,1}) & \cdots & \mathrm{H}(A_{n,n}, r_{n,n}) \end{pmatrix} \text{ if b=0}$$

$$\boldsymbol{challenge\ c} = \boldsymbol{V'}(\boldsymbol{G}, Com(A)) \in \{\boldsymbol{0}, \boldsymbol{1}\}$$

$$\text{Response } \boldsymbol{r} = \begin{cases} \begin{pmatrix} r_{1,1} & \cdots & r_{1,n} \\ \vdots & \ddots & \vdots \\ r_{n,1} & \cdots & r_{n,n} \end{pmatrix} \boldsymbol{and\ \sigma} & \boldsymbol{if\ c=b} \\ \perp & otherwise \end{cases}$$

$$\boldsymbol{Decision\ d} = \boldsymbol{V'}(\boldsymbol{G}, Com(A), \boldsymbol{c}, \boldsymbol{r})$$



Simulator
Cheat bit b,
$G = (V, E)$,
A= $\sigma(G)$
(random σ)

Zero-Knowledge: For all PPT V' exists PPT Sim s.t $\boldsymbol{View_{V'}} \equiv_C \mathrm{Sim}^{V'(.)}(A)$

Zero-Knowledge Proof Simulator

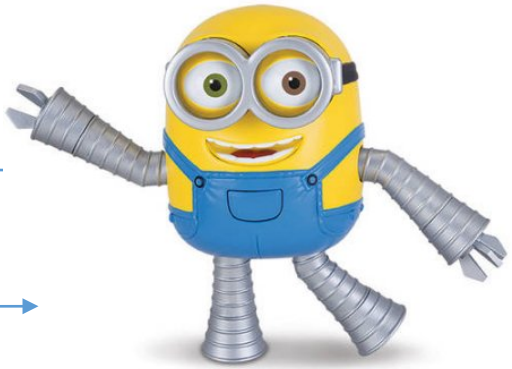
Dishonest (verifier);
$G = (V, E)$,

$$Com(K_n) = \begin{pmatrix} \mathrm{H}(0, r_{1,1}) & \cdots & \mathrm{H}(1, r_{1,n}) \\ \vdots & \ddots & \vdots \\ \mathrm{H}(1, r_{n,1}) & \cdots & \mathrm{H}(0, r_{n,n}) \end{pmatrix} \text{ if b=0}$$

$$\boldsymbol{challenge\ c} = \boldsymbol{V'}(\boldsymbol{G}, Com(A)) \in \{\boldsymbol{0}, \boldsymbol{1}\}$$

$$\boldsymbol{r} = \begin{cases} \begin{pmatrix} r_{\sigma(1),\sigma(1)} & \cdots & r_{\sigma(1),\sigma(k)} \\ \vdots & \ddots & \vdots \\ r_{\sigma(1),\sigma(k)} & \cdots & r_{\sigma(k),\sigma(k)} \end{pmatrix} & \boldsymbol{if\ c=b} \\ \perp & otherwise \end{cases}$$

$$\boldsymbol{Decision\ d} = \boldsymbol{V'}(\boldsymbol{G}, Com(A), \boldsymbol{c}, \boldsymbol{r})$$



Simulator
Cheat bit b,
$G = (V, E)$,
A= $\sigma(G)$
(random σ)

Zero-Knowledge: For all PPT V' exists PPT Sim s.t $\boldsymbol{View_{V'}} \equiv_C \mathrm{Sim}^{V'(.)}(A)$

Zero-Knowledge Proof for all NP

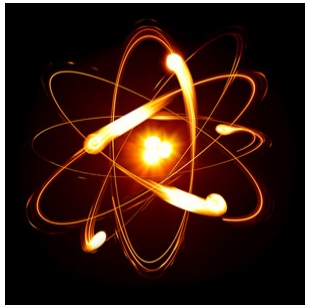
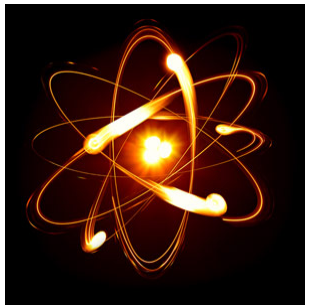
- **Completeness**: Honest prover can always make honest verifier accept
- **Soundness**: If prover commits to adjacency matrix $A_{\sigma(G)}$ of σ(G) and can reveal a clique in submatrix of $A_{\sigma(G)}$ then G itself contains a k-clique. Proof invokes binding property of commitment scheme.
- **Zero-Knowledge:** Simulator cheats and either commits to wrong adjacency matrix or cannot reveal clique. Repeat until we produce a successful transcript. Indistinguishability of transcripts follows from hiding property of commitment scheme.

Secure Multiparty Computation (Adversary Models)

- Semi-Honest ("honest, but curious")
  - All parties follow protocol instructions, but…
  - dishonest parties may be curious to violate privacy of others when possible
- Fully Malicious Model
  - Adversarial Parties may deviate from the protocol arbitrarily
    - Quit unexpectedly
    - Send different messages
  - It is much harder to achieve security in the fully malicious model
- Convert Secure Semi-Honest Protocol into Secure Protocol in Fully Malicious Mode?
  - Tool: Zero-Knowledge Proofs
  - Prove: My behavior in the protocol is consistent with honest party

CS 555:Week 15: Hot Topics

Shor's Algorithm

- Quantum Algorithm to Factor Integers

- Running Time

$$O((\log N)^2(\log\log N)(\log\log\log N))$$

- Building Quantum Circuits is challenging, but...
- RSA is broken if we build a quantum computer
  - Current record: Factor 21=3x7 with Shor's Algorithm
  - **Source:** Experimental Realisation of Shor's Quatum Factoring Algorithm Using Quibit Recycling (https://arxiv.org/pdf/1111.4147.pdf)

https://en.wikipedia.org/wiki/Shor%27s_algorithm

Quantum Resistant Crypto

- Symmetric key primitives are believed to be safe

- Integer Factoring, Discrete Log and Elliptic Curve Discrete Log are not safe
  - All public key encryption algorithms we have covered
  - RSA, RSA-OAEP, El-Gamal,….

https://en.wikipedia.org/wiki/Lattice-based_cryptography

Post Quantum Cryptography

- Symmetric key primitives are believed to be safe
- …but Grover's Algorithm does speed up brute-force attacks significantly ($2^n \; vs \; \sqrt{2^n}$)
  - Solution: Double Key Lengths
- Hashed Based Signatures
  - Lamport Signatures and extensions
- Lattice Based Cryptography is a promising approach for Quantum Resistant Public Key Crypto
  - Ring-LWE
  - NTRU

https://security.googleblog.com/2016/07/experimenting-with-post-quantum.html

Fully Homomorphic Encryption (FHE)

- Idea: Alice sends Bob $Enc_{PK_A}(x_1), \dots, Enc_{PK_A}(x_n)$

$$Enc_{PK_A}(x_i) + Enc_{PK_A}(x_j) = Enc_{PK_A}(x_i + x_j)$$

and

$$Enc_{PK_A}(x_i) \times Enc_{PK_A}(x_j) = Enc_{PK_A}(x_i \times x_j)$$

- Bob cannot decrypt messages, but given a circuit C can compute

$$Enc_{PK_A}(C(x_1, \dots, x_n))$$

- Proposed Application: Export confidential computation to cloud

https://simons.berkeley.edu/talks/shai-halevi-2015-05-18a (Lecture by Shai Halevi)

Fully Homomorphic Encryption (FHE)

- Idea: Alice sends Bob $Enc_{PK_A}(x_1), \ldots, Enc_{PK_A}(x_n)$
- Bob cannot decrypt messages, but given a circuit C can compute
$$Enc_{PK_A}(C(x_1, \ldots, x_n))$$
- We now have candidate constructions!
  - Encryption/Decryption are polynomial time
  - …but expensive in practice.
  - Proved to be CPA-Secure under plausible assumptions

- Remark 1: Partially Homomorphic Encryption schemes cannot be CCA-Secure. Why not?

https://simons.berkeley.edu/talks/shai-halevi-2015-05-18a (Lecture by Shai Halevi)

Partially Homomorphic Encryption

- Plain RSA is multiplicatively homomorphic

$$Enc_{PK_A}(x_i) \times Enc_{PK_A}(x_j) = Enc_{PK_A}(x_i \times x_j)$$

- But not additively homomorphic

- Pallier Cryptosystem

$$Enc_{PK_A}(x_i) \times Enc_{PK_A}(x_j) = Enc_{PK_A}(x_i + x_j)$$

$$\left(Enc_{PK_A}(x_i)\right)^k = Enc_{PK_A}(k \times x_j)$$

- Not same as FHE, but still useful in multiparty computation

https://en.wikipedia.org/wiki/Paillier_cryptosystem

Program Obfuscation (Theoretical Cryptography)

- Program Obfuscation
  - Idea: Alice obfuscates a circuit C and sends C to Bob
  - Bob can run C, but cannot learn "anything else"
  - Lots of applications…
- Indistinguishability Obfuscation
  - Theoretically Possible
    - In the sense that $f(n) = 2^{100000000}n^{100000}$ is technically polynomial time
- Secure Hardware Module (e.g., SGX) can be viewed as a way to accomplish this in practice
  - Must trust third party (e.g., Intel)

https://simons.berkeley.edu/talks/amit-sahai-2015-05-19a (Lecture by Amit Sahai)

Differential Privacy

Release Aggregate Statistics?

- Question 1: How many people in this room have cancer?
- Question 2: How many students in this room have cancer?
- The difference (A1-A2) exposes my answer!

Differential Privacy: Definition

- n people
- Neighboring datasets:
  - Replace x with x'

Name	CS Prof?	…	STD?
Bjork	-1	…	???

[DMNS06, DKMMN06]

$$(\epsilon, \delta)\text{-differential privacy: } \forall (D, D'),\ \forall S$$

$$\Pr[\mathsf{ALG}(D) \in S] \leq e^{\epsilon} \Pr[\mathsf{ALG}(D') \in S] + \delta$$

Differential Privacy vs Cryptography

- ε is not negligibly small.
- We are not claiming that, when D and D' are neighboring datasets,
$$\boldsymbol{Alg}(\boldsymbol{D}) \equiv_C \boldsymbol{Alg}(\boldsymbol{D'})$$
- Otherwise, we would have $\boldsymbol{Alg}(\boldsymbol{X}) \equiv_C \boldsymbol{Alg}(\boldsymbol{Y'})$ for any two data-sets X and Y.
- Why?

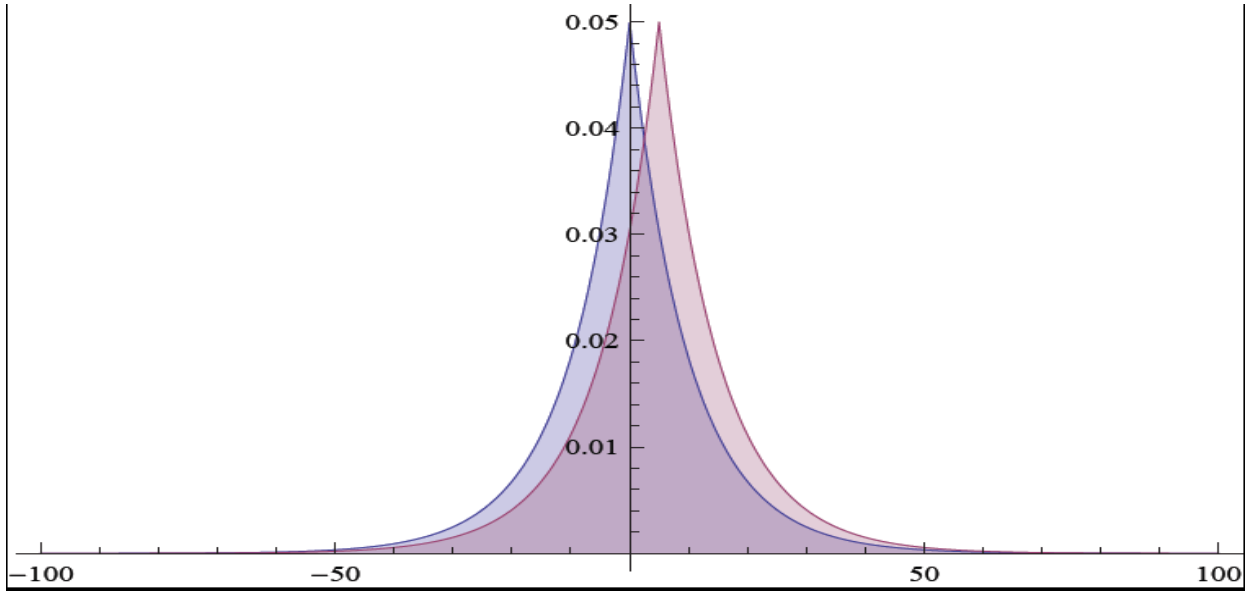
- Cryptography
  - Insiders/Outsiders
  - Only those with decryption key(s) can reveal secret
  - Multiparty Computation: Alice and Bob learn nothing other than f(x,y)

Traditional Differential Privacy Mechanism

Theorem: Let $\mathrm{D} = (x_1, \ldots, x_n) \in \{0,1\}^n$

$$\mathrm{A}(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i + \mathrm{Lap}\left(\frac{1}{\varepsilon}\right),$$

satisfies $(\varepsilon, 0)$-differential privacy. (True Answer, Noise)



Google Scholar

differential privacy

About 3,000,000 results (**0.06** sec)

Articles
Case law
My library

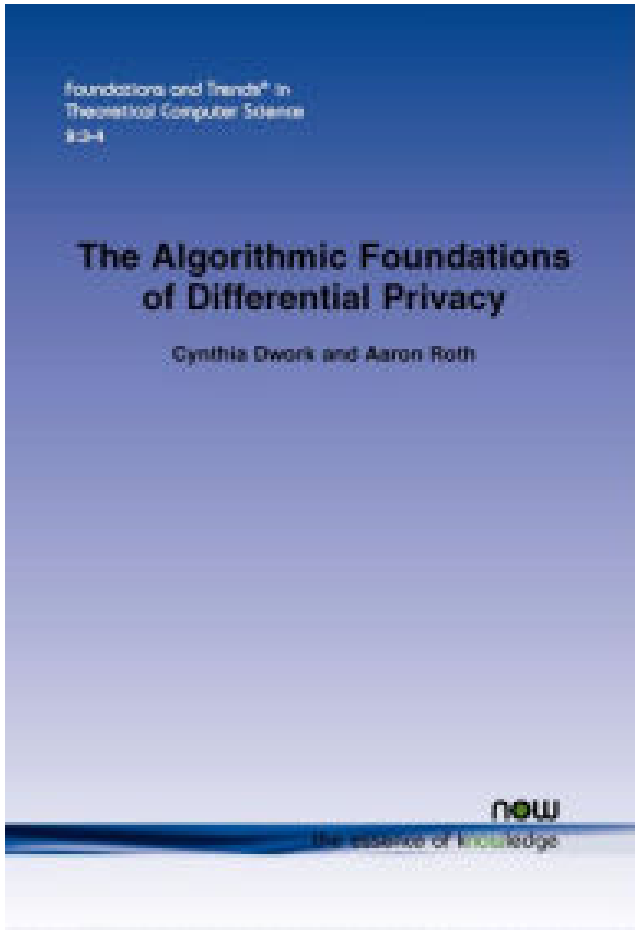
Any time
Since 2016
Since 2015
Since 2012
Custom range...

Differential privacy: A survey of results
C Dwork - International Conference on Theory and Applications of …, 2008 - Springer
Abstract Over the past five years a new approach to **privacy**-preserving data analysis has born fruit [13, 18, 7, 19, 5, 37, 35, 8, 32]. This approach differs from much (but not all!) of the related literature in the statistics, databases, theory, and cryptography communities, in that **...**
Cited by 2557 Related articles All 32 versions Web of Science: 365 Cite Save More

Mechanism design via **differential privacy**
F McSherry, K Talwar - … of Computer Science, 2007. FOCS'07. …, 2007 - ieeexplore.ieee.org
Abstract We study the role that **privacy**-preserving algorithms, which prevent the leakage of specific information about participants, can play in the design of mechanisms for strategic agents, which must encourage players to honestly report information. Specifically, we **...**
Cited by 708 Related articles All 25 versions Cite Save

Microsoft® Research

Resources

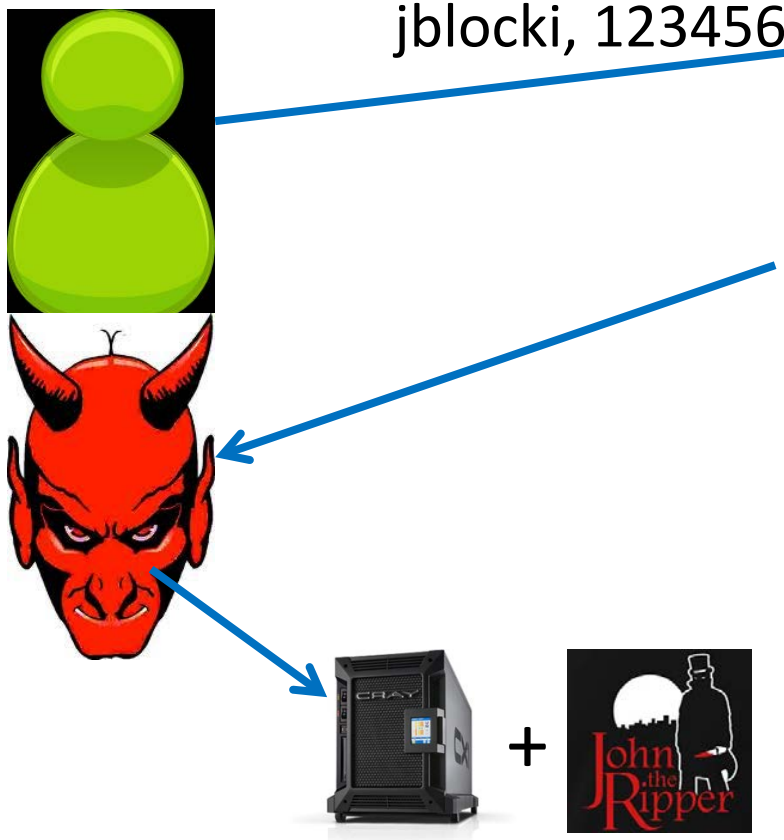


BARNES & NOBLE

- $99

Free PDF:
https://www.cis.upenn.edu/~aaroth/Papers/privacybook.pdf

Password Storage and Key Derivation Functions



Username	Salt	Hash
jblocki	89d978034a3f6	85e23cfe0021f584e3db87aa72630a9a2345c062

SHA1(12345689d978034a3f6)=85e23cfe0021f584e3db87aa72630a9a2345c062

Offline Attacks: A Common Problem

- Password breaches at major companies have affected ~~millions~~ **billions** of user accounts.

Offline Attacks: A Common Problem

- Password breaches at major companies have affected ~~millions~~ **billions**

TECH

Yahoo Triples Estimate of Breached Accounts to 3 Billion

Company disclosed late last year that 2013 hack exposed private information of over 1 billion users

By *Robert McMillan* and *Ryan Knutson*

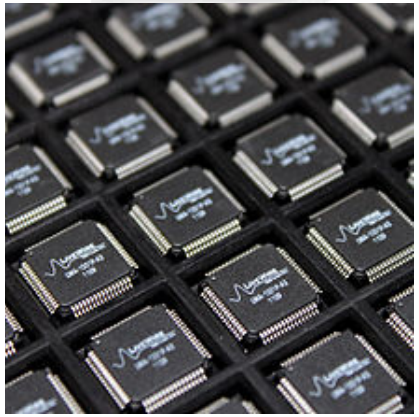
Updated Oct. 3, 2017 9:23 p.m. ET

A massive data breach at Yahoo in 2013 was far more extensive than previously disclosed, affecting all of its 3 billion user accounts, new parent company Verizon Communications Inc. said on Tuesday.

The figure, which Verizon said was based on new information, is three times the 1 billion accounts Yahoo said were affected when it first disclosed the breach in December 2016. The new disclosure, four months after Verizon completed its acquisition of Yahoo, shows that executives are still coming to grips with the extent of the...

Goal: Moderately Expensive Hash Function

Fast on PC and
Expensive on ASIC?



Attempt 1: Hash Iteration

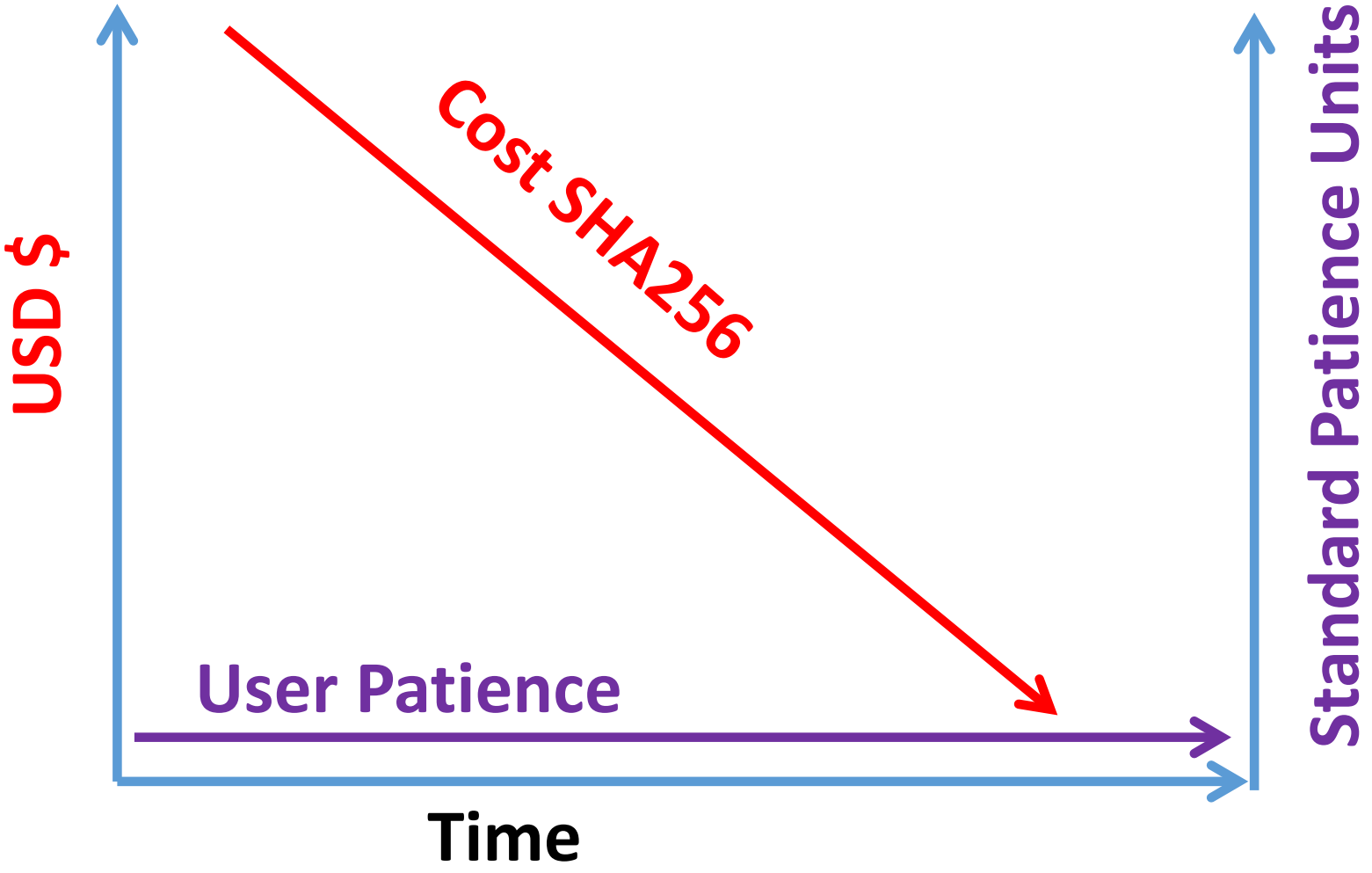
- BCRYPT

- PBKDF2

100,000 SHA256 computations (iterative)

Estimated Cost on ASIC: $1 per billion password guesses [BS14]

The Challenge

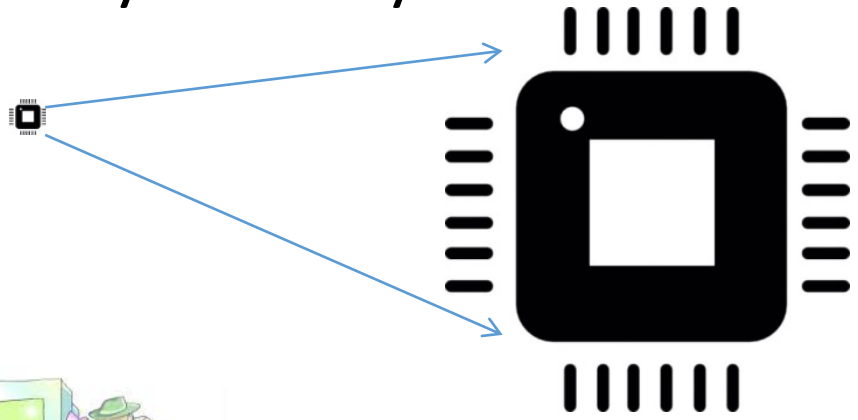


Disclaimer: This slide is entirely for humorous effect.

Memory Hard Function (MHF)

- Intuition: computation costs dominated by memory costs

vs.



sCrypt

- Data Independent Memory Hard Function (iMHF)
  - Memory access pattern should not depend on input

password hashing competition

(2013-2015)

https://password-hashing.net/

password hashing competition

(2013-2015)

We recommend that you use Argon2…

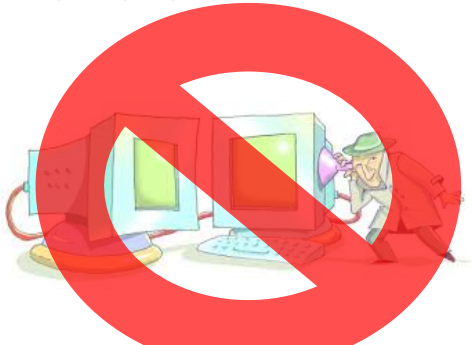
https://password-hashing.net/

password hashing competition

(2013-2015)

We recommend that you use Argon2…

There are two main versions of Argon2, **Argon2i** and Argon2d. **Argon2i** is the safest against side-channel attacks



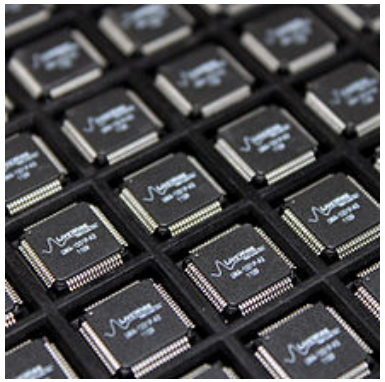
https://password-hashing.net/

Depth-Robustness: The Key Property

Necessary [AB16] and sufficient [ABP16] for secure iMHFs

Question

Are existing iMHF candidates based on depth-robust DAGs?



Answer: No

On the Depth-Robustness and Cumulative Pebbling Cost of Argon2i

Jeremiah Blocki* Samson Zhou†

August 4, 2017

Abstract

Argon2i is a data-independent memory hard function that won the password hashing competition. The password hashing algorithm has already been incorporated into several open source crypto libraries such as libsodium. In this paper we analyze the cumulative memory cost of computing Argon2i. On the positive side we provide a lower bound for Argon2i. On the negative side we exhibit an improved attack against Argon2i which demonstrates that our lower bound is nearly tight. In particular, we show that

(1) An Argon2i DAG is $(c, O(n^3/c^3))$-reducible.

(2) The cumulative pebbling cost for Argon2i is at most $O(n^{1.768})$. This improves upon the previous best upper bound of $O(n^{1.8})$ [AB17].

(3) Argon2i DAG is $\left(e, \tilde{\Omega}(n^3/e^3)\right)$-depth robust. By contrast, analysis of [ABP17a] only established that Argon2i was $\left(e, \tilde{\Omega}(n^2/e^2)\right)$-depth robust.

(4) The cumulative pebbling complexity of Argon2i is at least $\tilde{\Omega}(n^{1.75})$. This improves on the previous best bound of $\Omega(n^{1.66})$ [ABP17a] and demonstrates that Argon2i has higher cumulative memory cost than competing proposals such as Catena or Balloon Hashing.

...n2i and Balloon Hashing

Jeremiah Blocki
Purdue University

For the Alwen-Blocki attack to fail against practical memory parameters, Argon2i-B must be instantiated with more than 10 passes on memory. The current IRTF proposal calls even just 6 passes as the recommended "paranoid" setting.
More generally, the parameter selection process in the proposal is flawed in that it tends towards producing parameters for which the attack is successful (even under realistic constraints on parallelism).

...cted acyclic graph (DAG) G on $n = \Theta(\sigma * \tau)$ nodes representing

...analyzing iMHFs. First we define and motivate a new complexity ...(i.e. electricity) required to compute a function. We argue that, ...portant as the more traditional AT-complexity. Next we describe ...an iMHF based on an arbitrary DAG G. We upperbound both ...nce evaluated in terms of a certain combinatorial property of G. ...everal general classes of DAGs which include those underlying ...didates in the literature. In particular, we obtain the following ...meters σ and τ (and thread-count) such that $n = \sigma * \tau$.

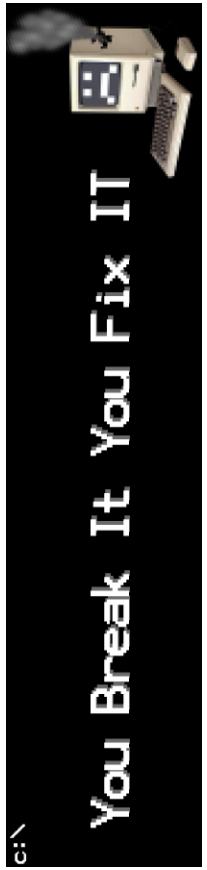
...[FLW13] has AT and energy complexities $O(n^{1.67})$.

...FLW13] has complexities is $O(n^{1.67})$.

...functions of [CGBS16] both have complexities in $O(n^{1.67})$.

- The Argon2i function of [BDK15] (winner of the Password Hashing Competition [PHC]) has complexities $O(n^{7/4}\log(n))$.

Can we build a secure iMHF?



Practical Graphs for Optimal Side-Channel Resistant Memory-Hard Functions

Joël Alwen*
IST Austria
jalwen@ist.ac.at

Jeremiah Blocki
Purdue University
jblocki@purdue.edu

Ben Harsha†
Purdue University
bharsha@purdue.edu

ABSTRACT

A memory-hard function (MHF) f_n with parameter n can be computed in sequential time and space n. Simultaneously, a high *amortized parallel* area-time complexity (aAT) is incurred per evaluation. In practice, MHFs are used to limit the rate at which an adversary (using a custom computational device) can evaluate a security sensitive function that still occasionally needs to be evaluated by honest users (using an off-the-shelf general purpose device). The most prevalent examples of such sensitive functions are Key Derivation Functions (KDFs) and password hashing algorithms where rate limits help mitigate off-line dictionary attacks. As the honest users' inputs to these functions are often (low-entropy) passwords special attention is given to a class of side-channel resistant MHFs called iMHFs.

Experimental benchmarks on a standard off-the-shelf CPU show that the new modifications do not adversely affect the impressive throughput of Argon2i (despite seemingly enjoying significantly higher aAT).

CCS CONCEPTS

• **Security and privacy** → **Hash functions and message authentication codes**;

KEYWORDS

hash functions; key stretching; depth-robust graphs; memory hard functions

1 INTRODUCTION

Github: https://github.com/Practical-Graphs/Argon2-Practical-Graph