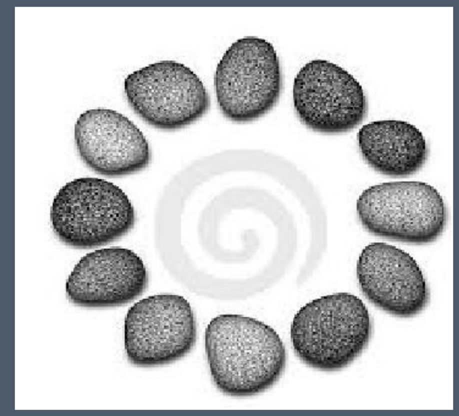
Password Hashing and Graph Pebbling*



Samson Zhou

Department of Computer Science, Purdue University

ABSTRACT

Although the passwords of users are no longer being stored, we show an offline attacker is compelled to crack all stolen passwords under current security recommendations. Memory hard functions have been proposed as moderately expensive cryptographic tools for password hashing. The cryptanalysis of these functions has focused on the cumulative memory complexity and the energy complexity of the function. The first metric measures how much memory users must commit to evaluating a function, while the second metric measures how much energy users must commit. We prove these evaluations reduce to pebbling games on graphs but show that a tool for exact cryptanalysis of functions is unlikely to exist. Nevertheless, we provide asymptotic upper and lower bounds on several families of functions, including Argon2i, the winner of the password hashing competition that is currently being considered for standardization by the Cryptography Form Research Group of the Internet Research Task Force.

BACKGROUND

- Data compromise is *inevitable*
- Recent corporations with leaked passwords:

OBJECTIVES

- Assuming password files are leaked, how can we protect against offline attackers?

User	Password
Stephen	auhsoJ
Lisa	hsifdrowS
James	1010NO1Z
Harry	sinocarD tupaC
Sarah	auhsoJ

User	Password Hash
Stephen	39e717cd3f5c4be78d97090c69f4e655
Lisa	f567c40623df407ba980bfad6dff5982
James	711f1f88006a48859616c3a5cbcc0377
Harry	fb74376102a049b9a7c5529784763c53
Sarah	39e717cd3f5c4be78d97090c69f4e655

- Make computation of hashes difficult for attackers!

METHODS

Economics of Password Cracking

- Develop a new game theoretic framework to quantify the damage of an offline attack
- Show that Yahoo! leaked passwords (over 70 million users) follow Zipfian distribution
- Analysis on a Zipfian distribution with estimated black market password costs
- Compared key-stretching vs. memory-hard function performance
- Model independent analysis, removing the assumption for Zipfian distribution

Models of Function Cost

- Formalized the bandwidth cost model
- Bandwidth-hard vs Memory-hard

Analysis of Password Hash Functions

- Showed NP-Hardness of computing bandwidth cost and cumulative memory cost
- Provided upper and lower bounds for cumulative memory cost for several functions
  - Argon2i, winner of the Password Hashing Competition, is currently being considered for standardization by the Internet Research Task Force (IRTF)
- Provided lower bounds for bandwidth cost for several functions
- Showed relationship between bandwidth cost and cumulative memory cost. Thus the goals of memory hardness are well-aligned.3

RESULTS

Sample Size (Millions)	y	r	R^2
15	0.00949	0.2843	0.9542
30	0.01321	0.2544	0.9531
45	0.01592	0.2384	0.9529
60	0.01810	0.2277	0.9530
Full	0.02112	0.2166	0.9544

TABLE 2: Yahoo! CDF-Zipf with Sub-sampling

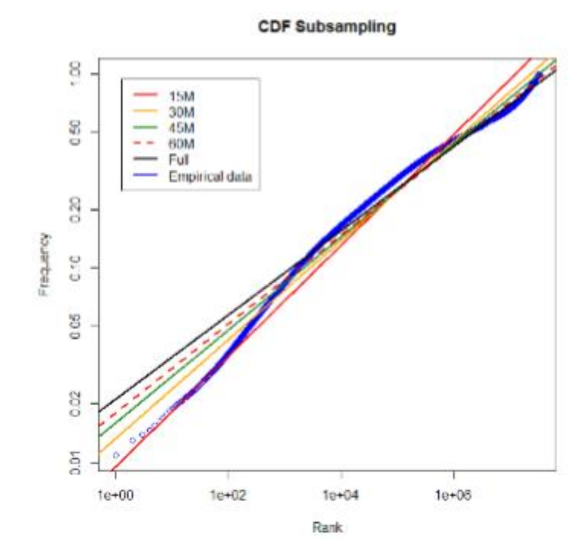


Fig. 1: Yahoo! CDF-Zipf Subsampling

Method	y	r	R^2	KS
LLS	0.0211	0.2166	0.9544	0.0094328
GSS	0.03315	0.1811	0.9498	0.022282

TABLE 3: Yahoo! CDF-Zipf Test Results

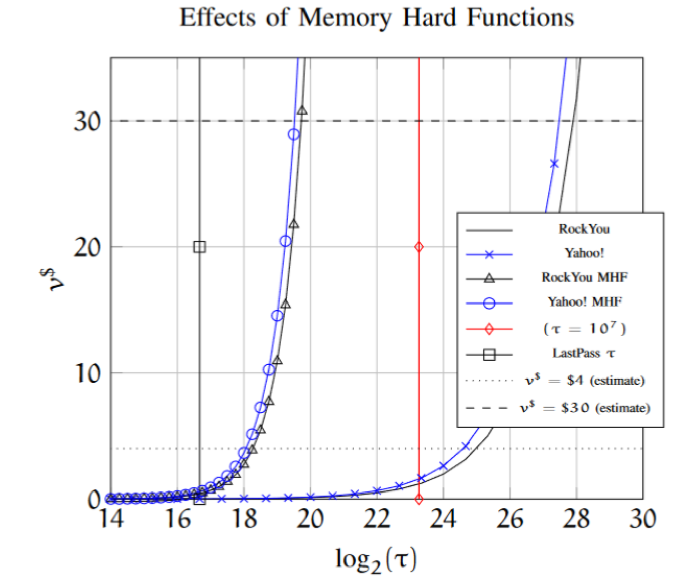


Fig. 3: Memory Hard Functions: $v^\$$ vs τ when $v = k \times T(y, r, 1)$ using thresholds $T(y, r, 1)$ for RockYou and Yahoo! $k = \tau C_H + \tau^2 C_{mem}$ for MHFs and $k = C_H \times \tau$ otherwise.

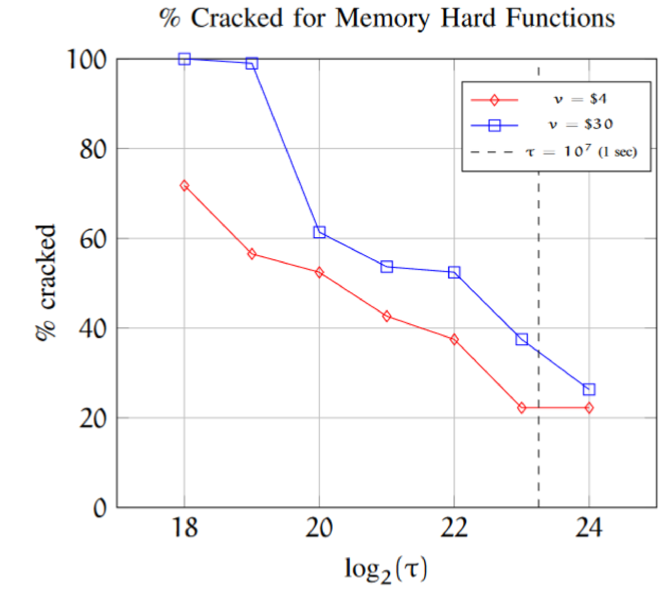
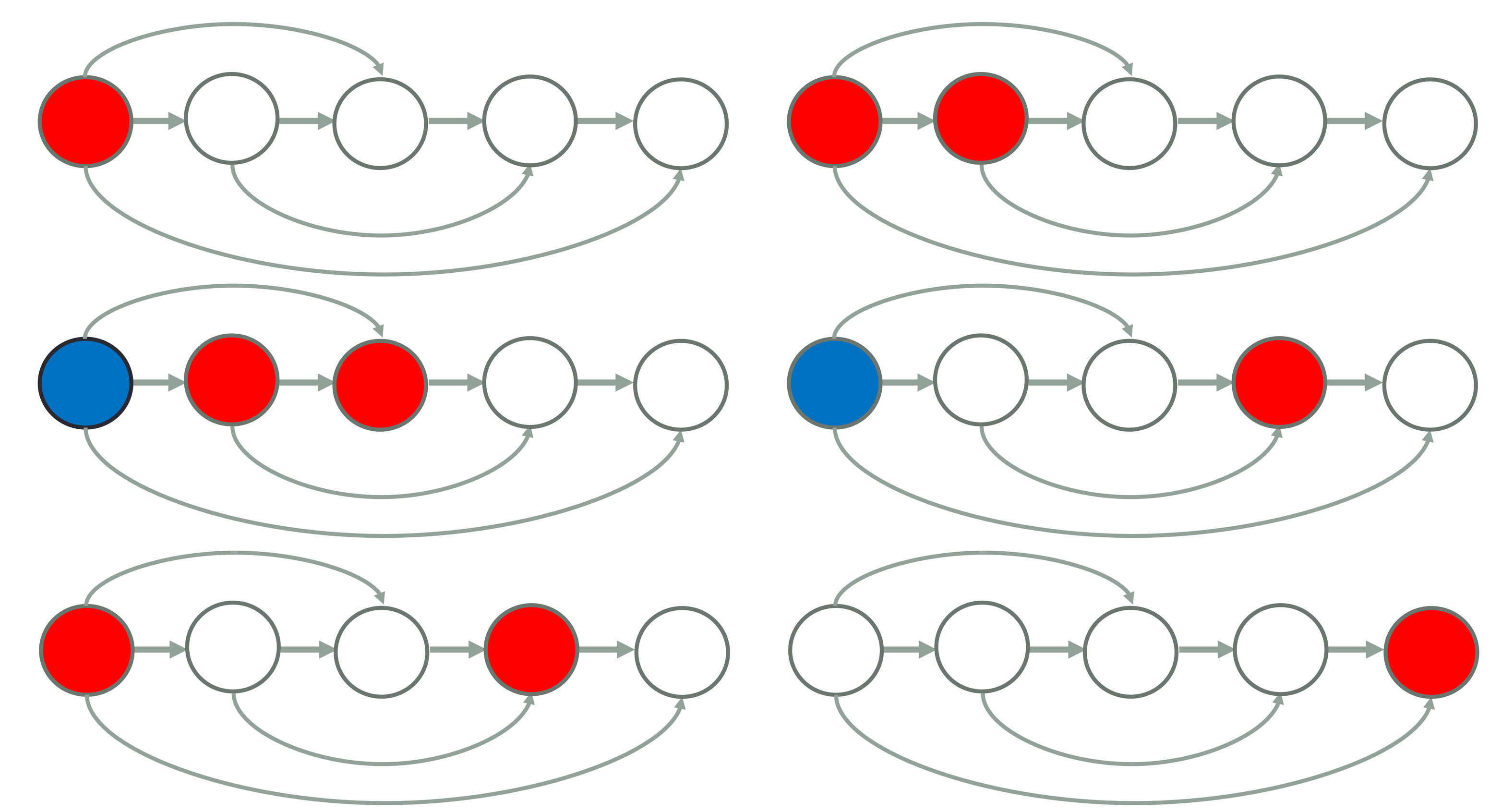


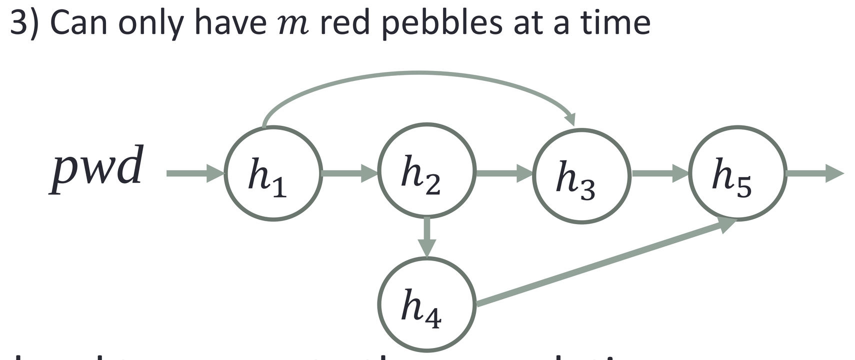
Fig. 5: Memory Hard Functions: % cracked by value $v^\$ \in \{\$4, \$30\}$ adversary against an ideal MHF with running time parameter τ.

RESULTS

- Bandwidth-hardness, which measures the amount of energy needed to compute a function, can be measured as red-blue graph pebbling



- Pebbling game goal is to place a pebble at the last node. Rules:
  - 1) Can only place red node if all parent nodes contain red nodes
  - 2) Can swap between red and blue pebbles at a node
  - 3) Can only have m red pebbles at a time



$$h_1 = H(pwd, salt)$$
$$h_2 = H(h_1)$$
$$h_3 = H(h_1, h_2)$$
$$h_4 = H(h_2)$$
$$h_5 = H(h_3, h_4)$$

- NP-hard to compute the cumulative memory or bandwidth cost of a function.
- The cumulative memory cost of Argon2i is $\Omega(n^{1.75})$ but $O(n^{1.768})$.
- The bandwidth cost of Argon2i is $\Omega(n^{5/3}c_r + nc_b)$.
- $\mathrm{BWC}(f) = \Omega\left(\sqrt{c_b c_r \mathrm{CMC}(f)} - c_b m\right)$, where BWC is the bandwidth cost and CMC is the cumulative memory cost of evaluating a function f.

*CITATIONS

- **Jeremiah Blocki, Ben Harsha, Samson Zhou.** *On the Economics of Offline Password Cracking.* IEEE Security and Privacy (S&P, Oakland) 2018
- **Jeremiah Blocki, Ling Ren, Samson Zhou.** *Bandwidth-Hard Functions: Reductions and Lower Bounds.* Manuscript
- **Jeremiah Blocki, Samson Zhou.** *On the Computational Complexity of Minimal Cumulative Cost Graph Pebbling.* Financial Cryptography and Data Security (FC) 2018
- **Jeremiah Blocki, Samson Zhou.** *On the Depth-Robustness and Cumulative Pebbling Cost of Argon2i.* 15th IACR Theory of Cryptography Conference (TCC) 2017