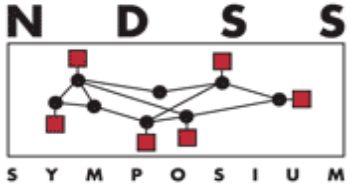
Tangled Web of Password Reuse

Anupam Das (UIUC),
Joseph Bonneau (Princeton University),
Matthew Caesar (UIUC),
Nikita Borisov (UIUC),
XiaoFeng Wang (Indiana University at Bloomington)

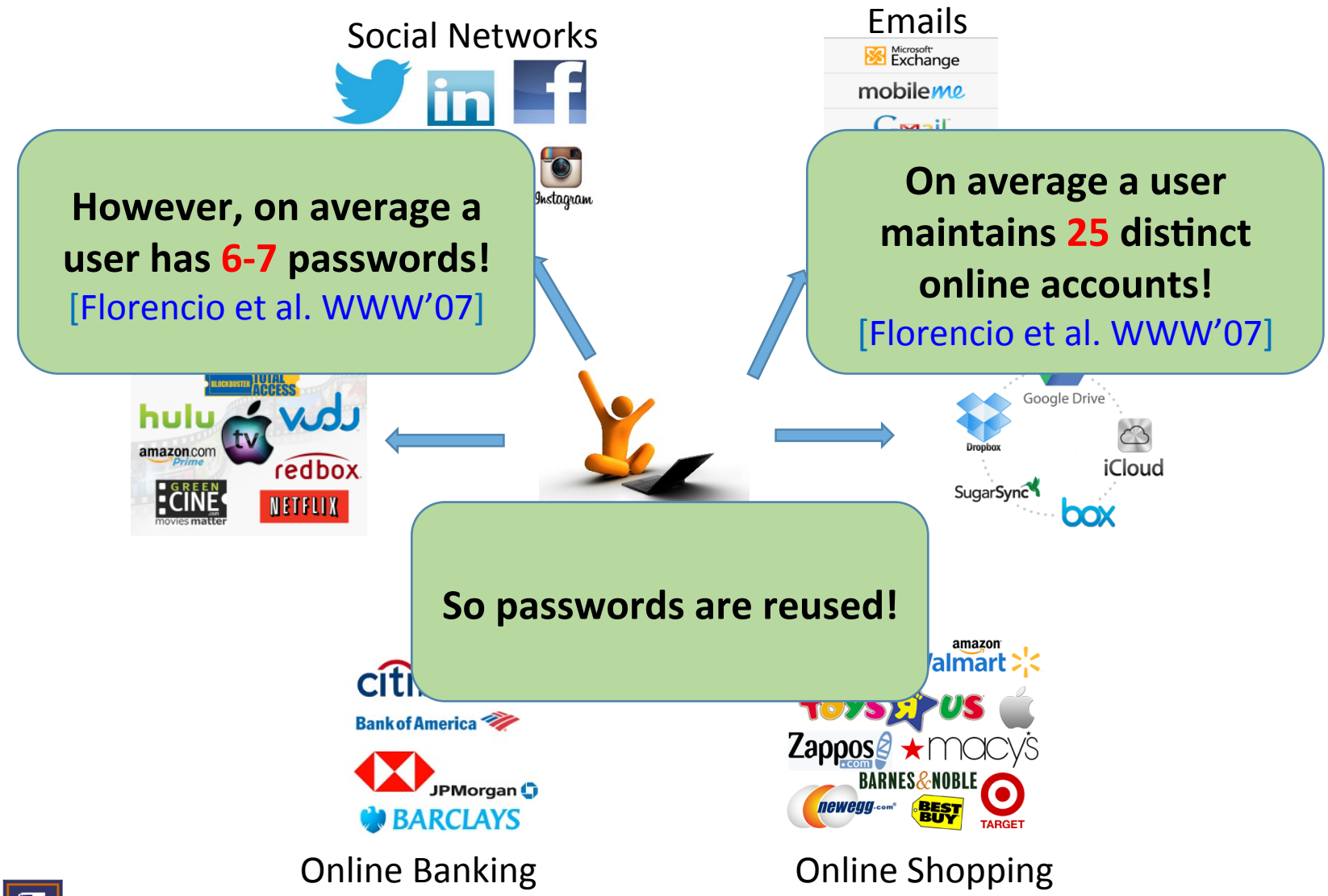


2014

Audience Poll

- **How many of you use the web?**
- **How many of you use web sites with accounts/ passwords?**
- **How many of you use different passwords on all the sites you use?**

The Problem


Social Networks
Emails
However, on average a user has 6-7 passwords!
[Florencio et al. WWW'07]
On average a user maintains 25 distinct online accounts!
[Florencio et al. WWW'07]
So passwords are reused!
Online Banking
Online Shopping

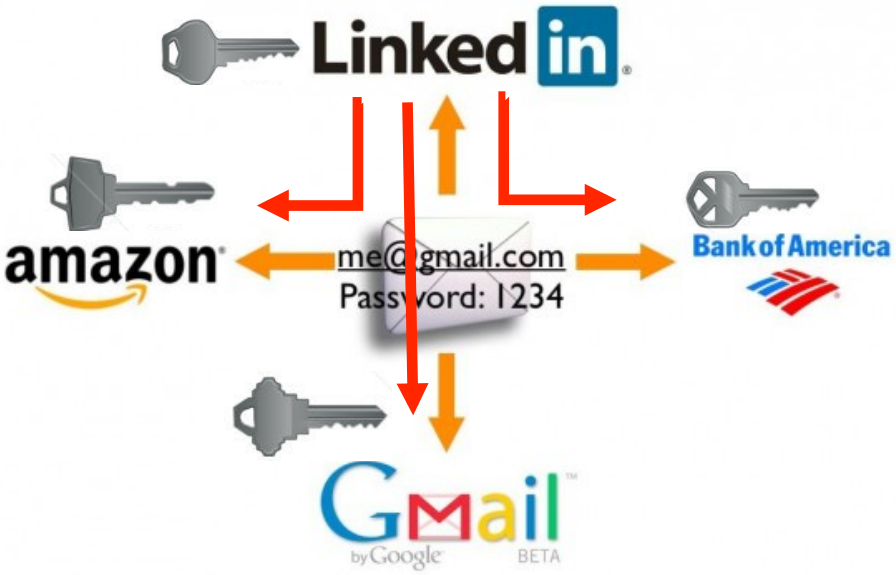
What's wrong with reuse?

Some High Profile Leaks:

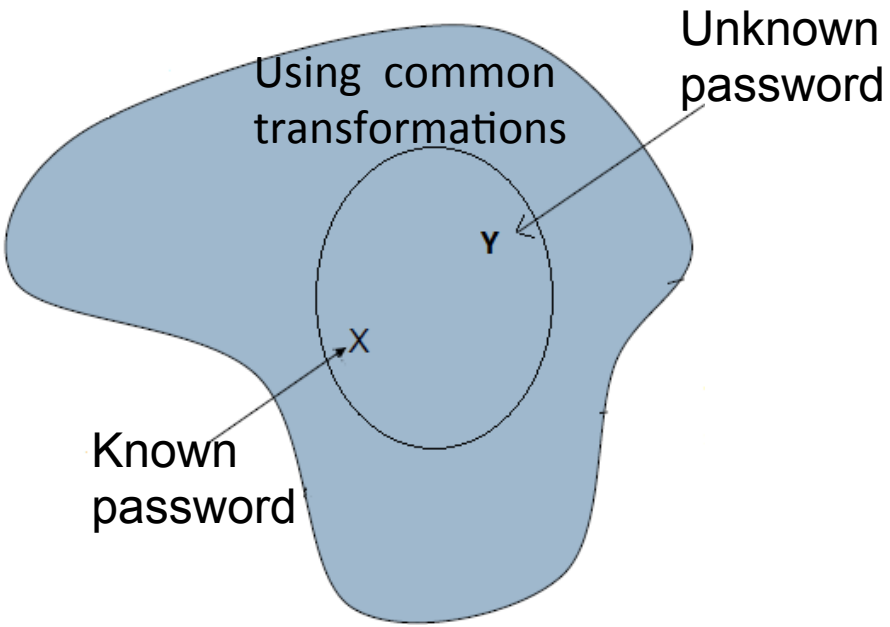
- Linkedin : 6.5 million
- Yahoo Voice mail: 450,000
- Twitter: 56,000
- RockYou: 32 million
- Adobe: 150 million

Enables cross-site password guessing.

What if passwords are not same?



Say modify passwords based on web site policy.



Search Space

- Use leaked password to guess unknown password
- Potentially reduce the search space.

Our work

We study the problem of password reuse across different web sites-

- Characterize extent of the problem
  - Conduct measurement and user study

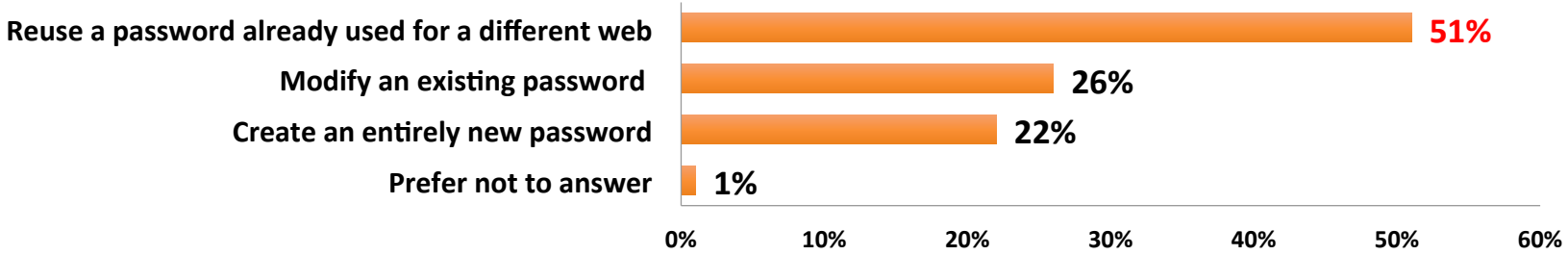
- Characterize severity of the problem
  - Develop and evaluate cross-site guessing algorithm

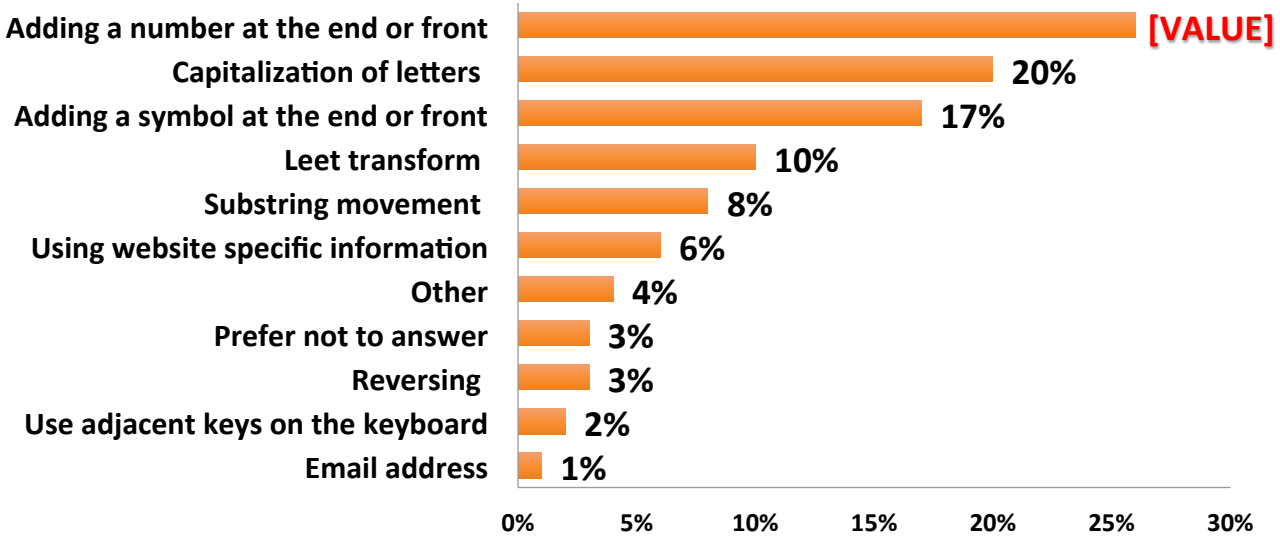
Challenges and approaches

- How you obtain data to evaluate on?
  - User study, collect publicly leaked passwords
- How do we analyze data?
  - Derive typical “transformations”
- What’s a good algorithm to guess passwords?
  - Parameterize and derive ordering of transformations that minimizes the number of guesses

Getting data with a user study

To gain insight into users' behavior and thought processes when creating passwords for different websites, we conducted an anonymous survey. We had a total of 224 participants.





Getting more data from leaked passwords

Publicly leaked email and password pairs from 11 different web sites:

Site	#	Year
csdn.net	6428630	2011
gawker.com	748559	2010
voices.yahoo.com	442837	2012
militarysingles.com	163482	2012
rootkit.com	81450	2011
myspace.com	49711	2006
porn.com	25934	2011
hotmail.com	8504	2009
facebook.com	8183	2011
youporn.com	5388	2012

Total 6077 unique users

Passwords Per user	Percentage
2	97.75%
3	1.82%
4	0.26%
5	0.15%
6	0.02%

Snapshot of leaked passwords

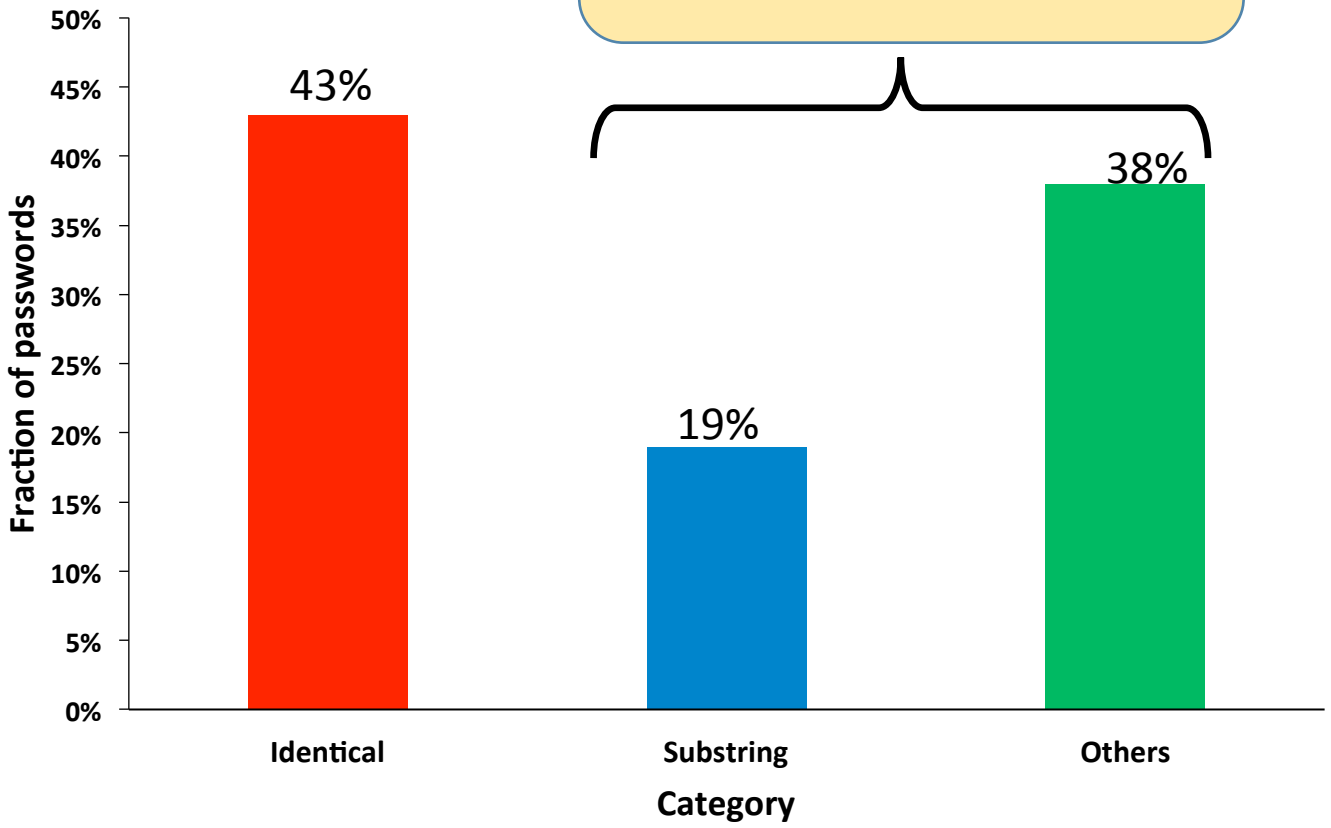
Email ID	Passwords	
116	iloveyou	iloveyou
117	loving	loving1
118	naughty	NAUGHTY
119	password	pa55wOrd
120	logout0616	logout
121	butcher05	Butcher05
122	joey1992	joey92
123	123456	12345678
124	gzwz0204	0204gzwz
125	mike04	jade1979
126	lucky777	lucky7

- **Identical**
- **Substring**
- **Others (more complex transformations)**

Categorizing passwords

We categorize the passw gories

We focus on trying to guess these passwords



Characterizing similarity of passwords

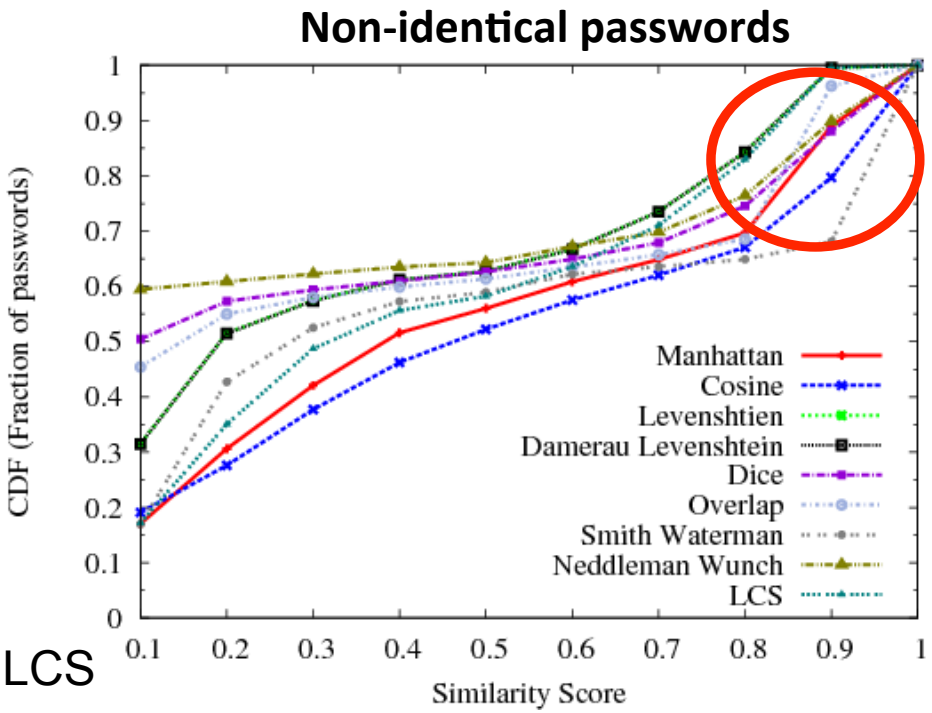
To get a better understanding of the similarity of the non-identical passwords, we look at different similarity metrics.

Distance-like functions
Manhattan, Cosine

Edit distance-like functions
Levenshtein, Damerau Levenshtein

Token-based distance functions
Dice, Overlap

Alignment-like functions
Smith-Waterman, Neddleman-Wunsch, LCS



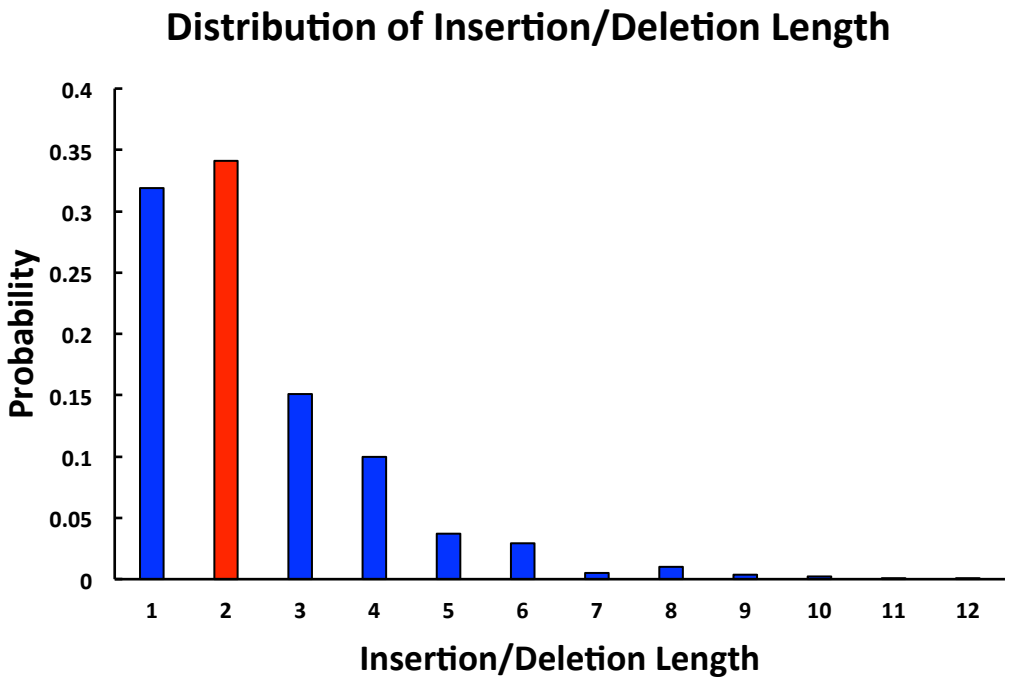
Deriving transformation operations

Lets start with passwords that are substring of each other as they require only insertions or deletions.

Insertion/Deletion Location

Location	Percentage
At start	10%
At end	**88**%
At both ends	2%

Similar to what we found in Our user study.



Most insertions/deletions are of length= 2

Deriving more complex transformations

- Sequential key:
  qwerasdf →1234qwer
- Sequential alternative key:
  12345 → !@#$%
- Sequential alphabet:
  abcde → 12345
- Capitalization:
  naughty → NAUGHTY
- Reverse:
  123456 →654321
- LeetSpeak Transformation:
  password → pa$$w0rd
- Substring Movement:
  gzwz0204 → 0204gzwz

Other less common transformations also exist

- repeating character sequences
- swap of positions
- using email address
- Etc.

Automating password guessing

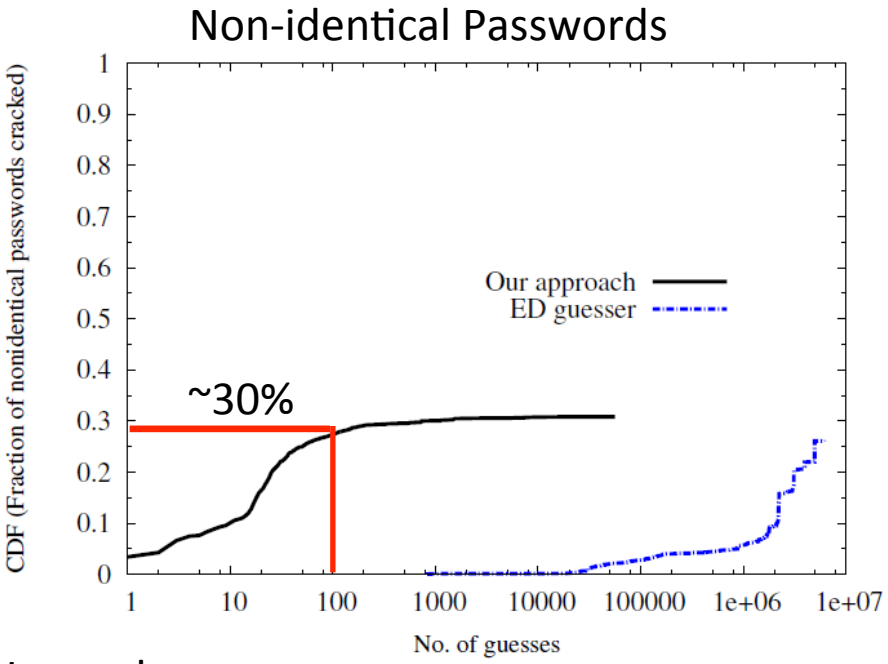
The question to ask:
Given a leaked password (as seed), can we design an algorithm to automatically guess other passwords?

Answer:
Yes, we can!

Goals:

1. Compute the orderings and parameterizations that require minimum number of guesses
   - We obtain the ordering using 40% of our data
2. Make the design applicable for online attack scenario.

Number of guesses



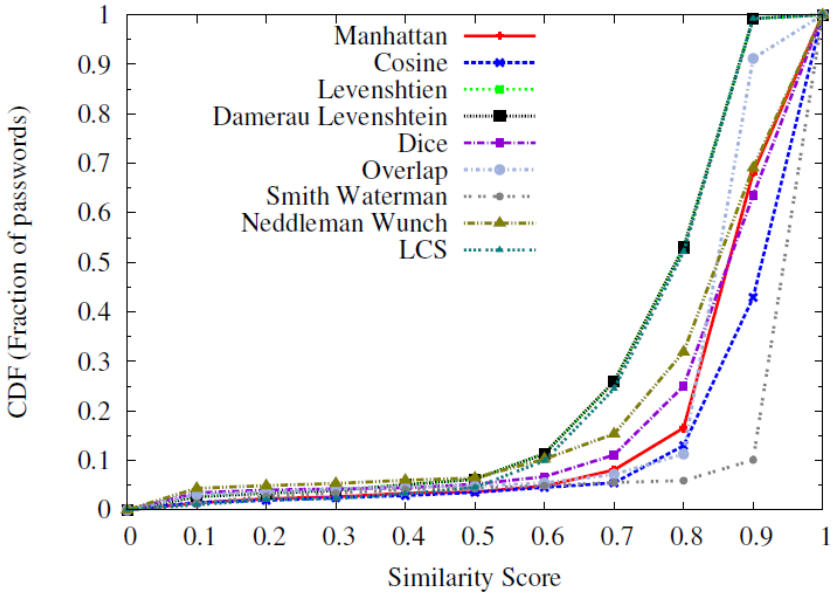
We were able to guess **75**% of the passwords in the 'Substring' category within 100 attempts

Legend:
ED guesser -- Edit Distance based Guesser

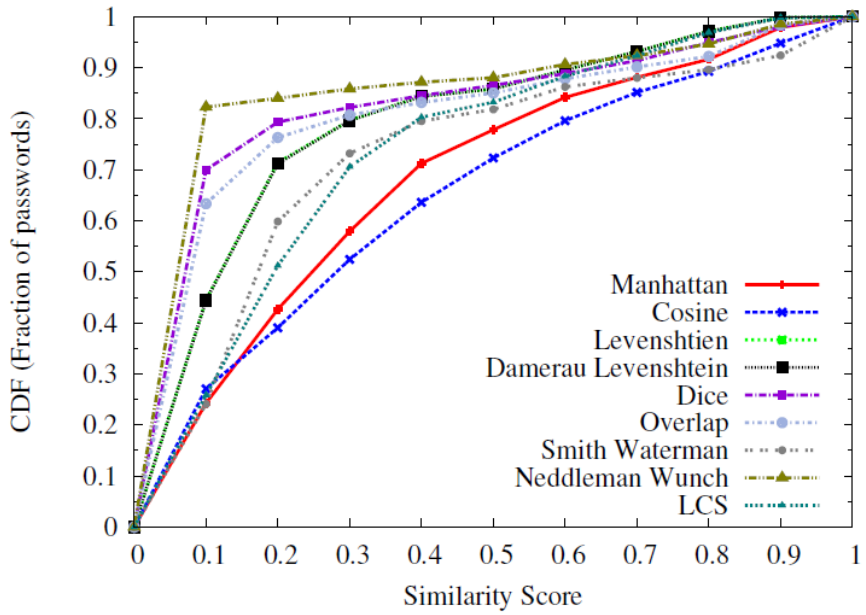
- Able to guess ~30% passwords within 100 attempts.
- Our approach is therefore more suitable for online attack scenarios.

Similarity of guessed passwords

Cracked Passwords



Non-Cracked Passwords



- Majority of the correctly guessed passwords had high similarity score.
- Majority of the non-cracked passwords had small similarity score

Conclusion

- Password reuse is common
  - We found 43-51% of users reused their passwords
- Password reuse is harmful
  - Makes cross-site guessing easier.
  - We were able to guess 30% of the non-identical passwords within 100 attempts.

Even a "low-value" website compromise can be serious. A hack of your Zynga (Farmville) account can potentially compromise your Gmail account!

Details about the project is available at-
http://web.engr.illinois.edu/~das17/passwordreuse.html

END

Related Works

- Guess again (and again and again) [IEEE S&P 2012]
  - Perform comparative strength of different composition policy

- Password cracking using Probabilistic Context Free Grammars[IEEE S&P 2009]
  - Uses Probabilistic Context Free Grammar to generate new word mingling rules.

- Adaptive Password Strength Meters from Markov Models [NDSS 2012]
  - Uses Markov models to guess passwords

- Password Strength: An Empirical Analysis [InfoCom 2010]
  - Compare PCFG, Markov model, Dictionary attack

- Security of modern password expiration [CCS 2010]
  - Offline 41% and online 17% (5 guesses) password cracked.

- How does your Password Measure Up? [USENIX 2012]
  - Studies user behavior for different password meter