ECE750: Usable Security and Privacy

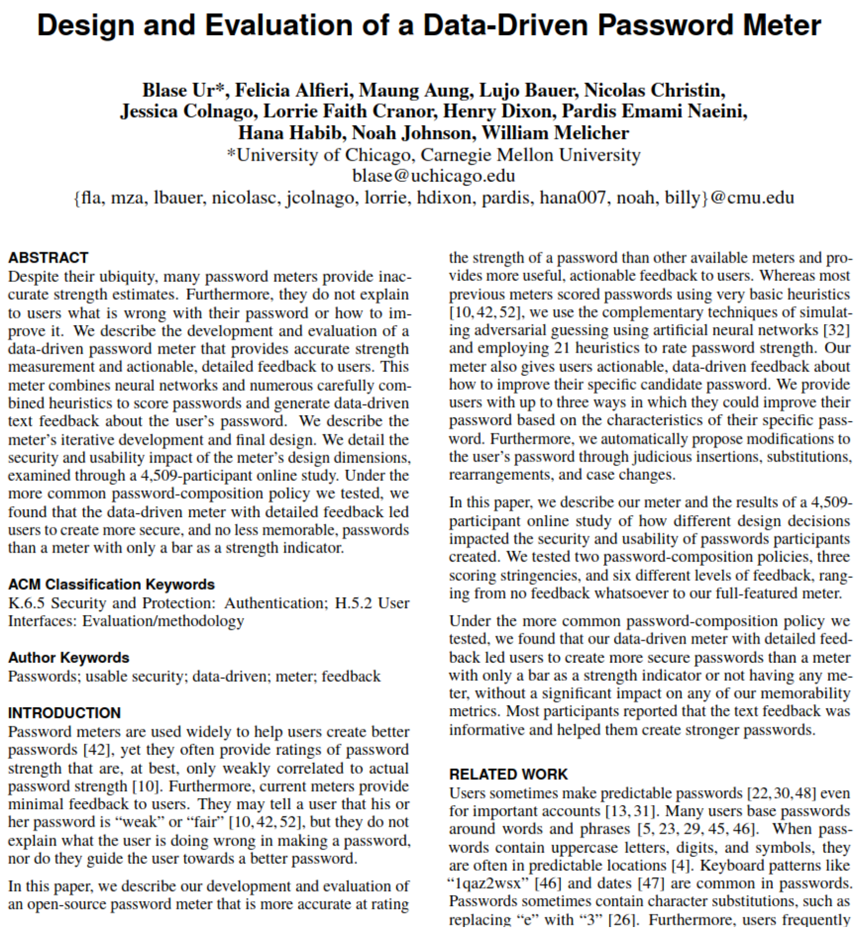
Authentication

Dr. Kami Vaniea
Electrical and Computer Engineering
kami.vaniea@uwaterloo.ca

First, the news…

- First 5 minutes we talk about something interesting and recent
- You will not be tested on the news part of lecture
- You may use news as an example on tests
- Why do this?
    1. Some students show up late for various good reasons
    2. Reward students who show up on time
    3. Important to see real world examples

Two possible papers today:



Design and Evaluation of a Data-Driven Password Meter

Blase Ur*, Felicia Alfieri, Maung Aung, Lujo Bauer, Nicolas Christin,
Jessica Colnago, Lorrie Faith Cranor, Henry Dixon, Pardis Emami Naeini,
Hana Habib, Noah Johnson, William Melicher
*University of Chicago, Carnegie Mellon University
blase@uchicago.edu
{fla, mza, lbauer, nicolasc, jcolnago, lorrie, hdixon, pardis, hana007, noah, billy}@cmu.edu

ABSTRACT

Despite their ubiquity, many password meters provide inaccurate strength estimates. Furthermore, they do not explain to users what is wrong with their password or how to improve it. We describe the development and evaluation of a data-driven password meter that provides accurate strength measurement and actionable, detailed feedback to users. This meter combines neural networks and numerous carefully combined heuristics to score passwords and generate data-driven text feedback about the user's password. We describe the meter's iterative development and final design. We detail the security and usability impact of the meter's design dimensions, examined through a 4,509-participant online study. Under the more common password-composition policy we tested, we found that the data-driven meter with detailed feedback led users to create more secure, and no less memorable, passwords than a meter with only a bar as a strength indicator.

ACM Classification Keywords

K.6.5 Security and Protection: Authentication; H.5.2 User Interfaces: Evaluation/methodology

Author Keywords

Passwords; usable security; data-driven; meter; feedback

INTRODUCTION

Password meters are used widely to help users create better passwords [42], yet they often provide ratings of password strength that are, at best, only weakly correlated to actual password strength [10]. Furthermore, current meters provide minimal feedback to users. They may tell a user that his or her password is "weak" or "fair" [10, 42, 52], but they do not explain what the user is doing wrong in making a password, nor do they guide the user towards a better password.

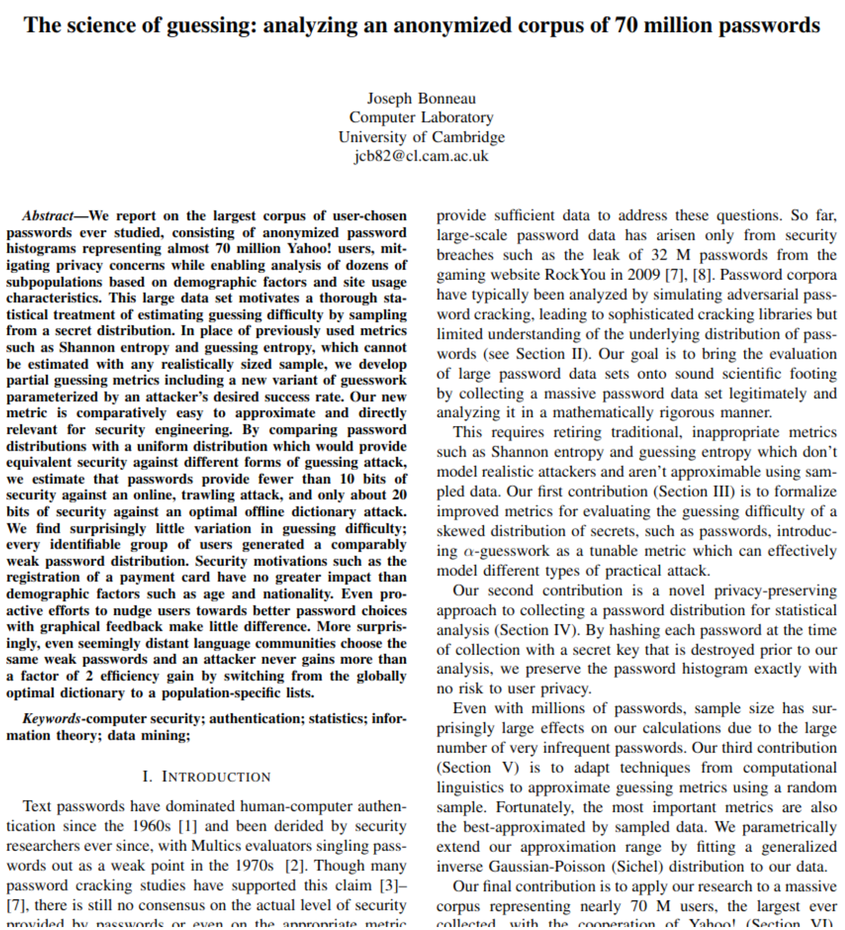
In this paper, we describe our development and evaluation of an open-source password meter that is more accurate at rating the strength of a password than other available meters and provides more useful, actionable feedback to users. Whereas most previous meters scored passwords using very basic heuristics [10, 42, 52], we use the complementary techniques of simulating adversarial guessing using artificial neural networks [32] and employing 21 heuristics to rate password strength. Our meter also gives users actionable, data-driven feedback about how to improve their specific candidate password. We provide users with up to three ways in which they could improve their password based on the characteristics of their specific password. Furthermore, we automatically propose modifications to the user's password through judicious insertions, substitutions, rearrangements, and case changes.

In this paper, we describe our meter and the results of a 4,509-participant online study of how different design decisions impacted the security and usability of passwords participants created. We tested two password-composition policies, three scoring stringencies, and six different levels of feedback, ranging from no feedback whatsoever to our full-featured meter.

Under the more common password-composition policy we tested, we found that our data-driven meter with detailed feedback led users to create more secure passwords than a meter with only a bar as a strength indicator or not having any meter, without a significant impact on any of our memorability metrics. Most participants reported that the text feedback was informative and helped them create stronger passwords.

RELATED WORK

Users sometimes make predictable passwords [22, 30, 48] even for important accounts [13, 31]. Many users base passwords around words and phrases [5, 23, 29, 45, 46]. When passwords contain uppercase letters, digits, and symbols, they are often in predictable locations [4]. Keyboard patterns like "1qaz2wsx" [46] and dates [47] are common in passwords. Passwords sometimes contain character substitutions, such as replacing "e" with "3" [26]. Furthermore, users frequently



The science of guessing: analyzing an anonymized corpus of 70 million passwords

Joseph Bonneau
Computer Laboratory
University of Cambridge
jcb82@cl.cam.ac.uk

***Abstract*—We report on the largest corpus of user-chosen passwords ever studied, consisting of anonymized password histograms representing almost 70 million Yahoo! users, mitigating privacy concerns while enabling analysis of dozens of subpopulations based on demographic factors and site usage characteristics. This large data set motivates a thorough statistical treatment of estimating guessing difficulty by sampling from a secret distribution. In place of previously used metrics such as Shannon entropy and guessing entropy, which cannot be estimated with any realistically sized sample, we develop partial guessing metrics including a new variant of guesswork parameterized by an attacker's desired success rate. Our new metric is comparatively easy to approximate and directly relevant for security engineering. By comparing password distributions with a uniform distribution which would provide equivalent security against different forms of guessing attack, we estimate that passwords provide fewer than 10 bits of security against an online, trawling attack, and only about 20 bits of security against an optimal offline dictionary attack. We find surprisingly little variation in guessing difficulty; every identifiable group of users generated a comparably weak password distribution. Security motivations such as the registration of a payment card have no greater impact than demographic factors such as age and nationality. Even proactive efforts to nudge users towards better password choices with graphical feedback make little difference. More surprisingly, even seemingly distant language communities choose the same weak passwords and an attacker never gains more than a factor of 2 efficiency gain by switching from the globally optimal dictionary to a population-specific lists.**

***Keywords*-computer security; authentication; statistics; information theory; data mining;**

I. Introduction

Text passwords have dominated human-computer authentication since the 1960s [1] and been derided by security researchers ever since, with Multics evaluators singling passwords out as a weak point in the 1970s [2]. Though many password cracking studies have supported this claim [3]–[7], there is still no consensus on the actual level of security provided by passwords or even on the appropriate metric provide sufficient data to address these questions. So far, large-scale password data has arisen only from security breaches such as the leak of 32 M passwords from the gaming website RockYou in 2009 [7], [8]. Password corpora have typically been analyzed by simulating adversarial password cracking, leading to sophisticated cracking libraries but limited understanding of the underlying distribution of passwords (see Section II). Our goal is to bring the evaluation of large password data sets onto sound scientific footing by collecting a massive password data set legitimately and analyzing it in a mathematically rigorous manner.

This requires retiring traditional, inappropriate metrics such as Shannon entropy and guessing entropy which don't model realistic attackers and aren't approximable using sampled data. Our first contribution (Section III) is to formalize improved metrics for evaluating the guessing difficulty of a skewed distribution of secrets, such as passwords, introducing α-guesswork as a tunable metric which can effectively model different types of practical attack.

Our second contribution is a novel privacy-preserving approach to collecting a password distribution for statistical analysis (Section IV). By hashing each password at the time of collection with a secret key that is destroyed prior to our analysis, we preserve the password histogram exactly with no risk to user privacy.

Even with millions of passwords, sample size has surprisingly large effects on our calculations due to the large number of very infrequent passwords. Our third contribution (Section V) is to adapt techniques from computational linguistics to approximate guessing metrics using a random sample. Fortunately, the most important metrics are also the best-approximated by sampled data. We parametrically extend our approximation range by fitting a generalized inverse Gaussian-Poisson (Sichel) distribution to our data.

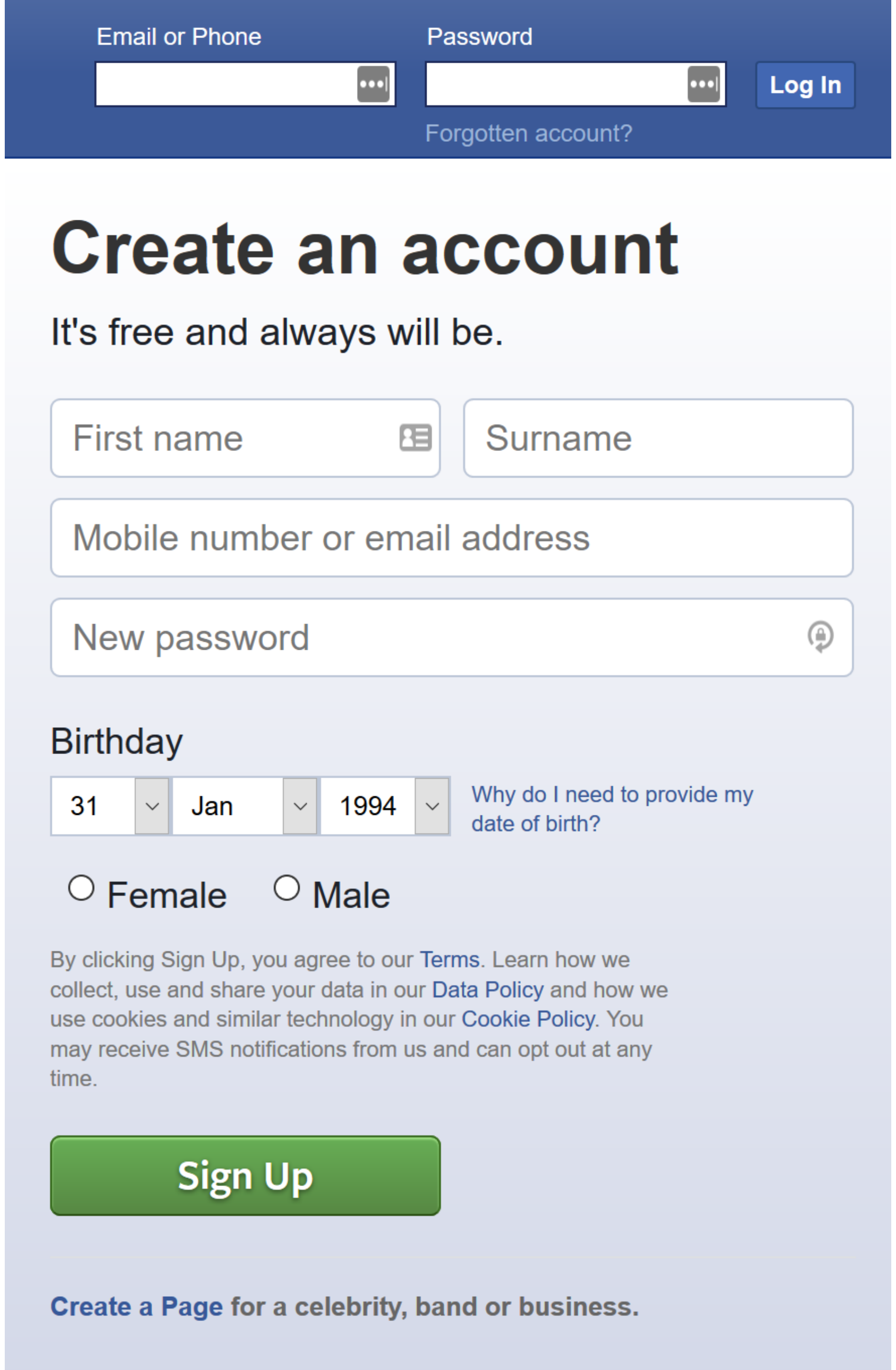
Our final contribution is to apply our research to a massive corpus representing nearly 70 M users, the largest ever collected, with the cooperation of Yahoo! (Section VI)

AUTHENTICATION

Authentication

- Verifying a fact about an entity before allowing it/them to perform an action
  - Entity could be a person or a computer or even an animal (i.e. dog doors)
  - Actions include: viewing, reading, writing, or interacting in any way
- Authentication should happen every time an action is taken and there is no way to be certain that the authenticated entity has not changed.
  - Authentications do not have to be the same
  - Initial authentication can be:
    - person -> computer
    - person -> web server
    - computer -> computer

When you think of authentication you probably envision a password login like this one.



Email or Phone

Password

Log In

Forgotten account?

Create an account

It's free and always will be.

First name

Surname

Mobile number or email address

New password

Birthday

31 Jan 1994

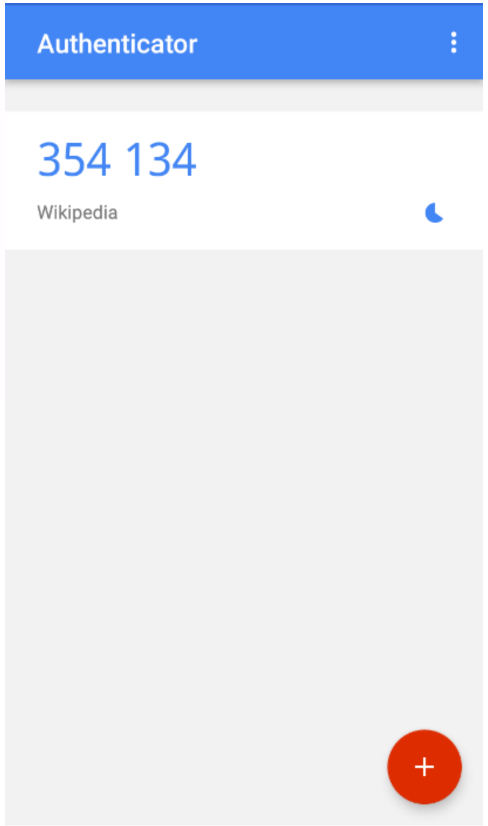
Why do I need to provide my date of birth?

○ Female ○ Male

By clicking Sign Up, you agree to our Terms. Learn how we collect, use and share your data in our Data Policy and how we use cookies and similar technology in our Cookie Policy. You may receive SMS notifications from us and can opt out at any time.

Sign Up

Create a Page for a celebrity, band or business.



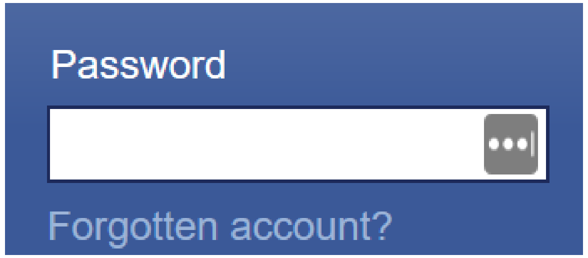
Authenticator

354 134

Wikipedia

There are many forms of authentication

Authentication factors (for humans)



Password

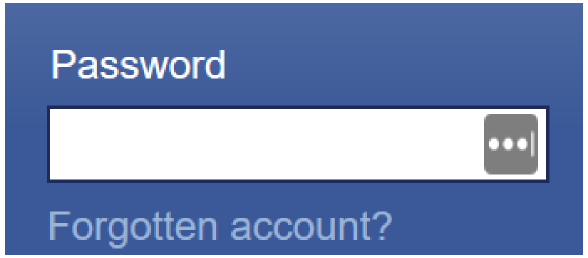
Forgotten account?

- Something you **know**
  - Password, mother's maiden name, your address

- Something you **have**
  - Student ID card, credit card chip, RSA key fob, Yubikey

- Something you **are**
  - Fingerprints, voice tones, iris, typing patterns

Also jokingly known as:



- Something you **can forget**
  - Password, mother’s maiden name, your address

- Something you **can loose**
  - Student ID card, credit card chip, RSA key fob, Yubikey

- Something you **cannot change**
  - Fingerprints, voice tones, iris, typing patterns

Something you know

Something you know

- Passwords
- Birthdate
- Last ATM visited
- Last purchase made
- Where you lived in 2012
- Drivers license number
- SIN number
- Favorite song
- Make and model of first car

Something you have

Physical keys

- Simplest and one of the most common examples of something you have
- Each key contains a “code” in the form of notches on the key
- Having one allows you to open physical locks
- Single factor authentication

RSA key fob

- When a button is pushed the fob prints out a number
- The number is generated securely using methods we will talk about later
- The number must be typed in along with a password
- Two factor authentication

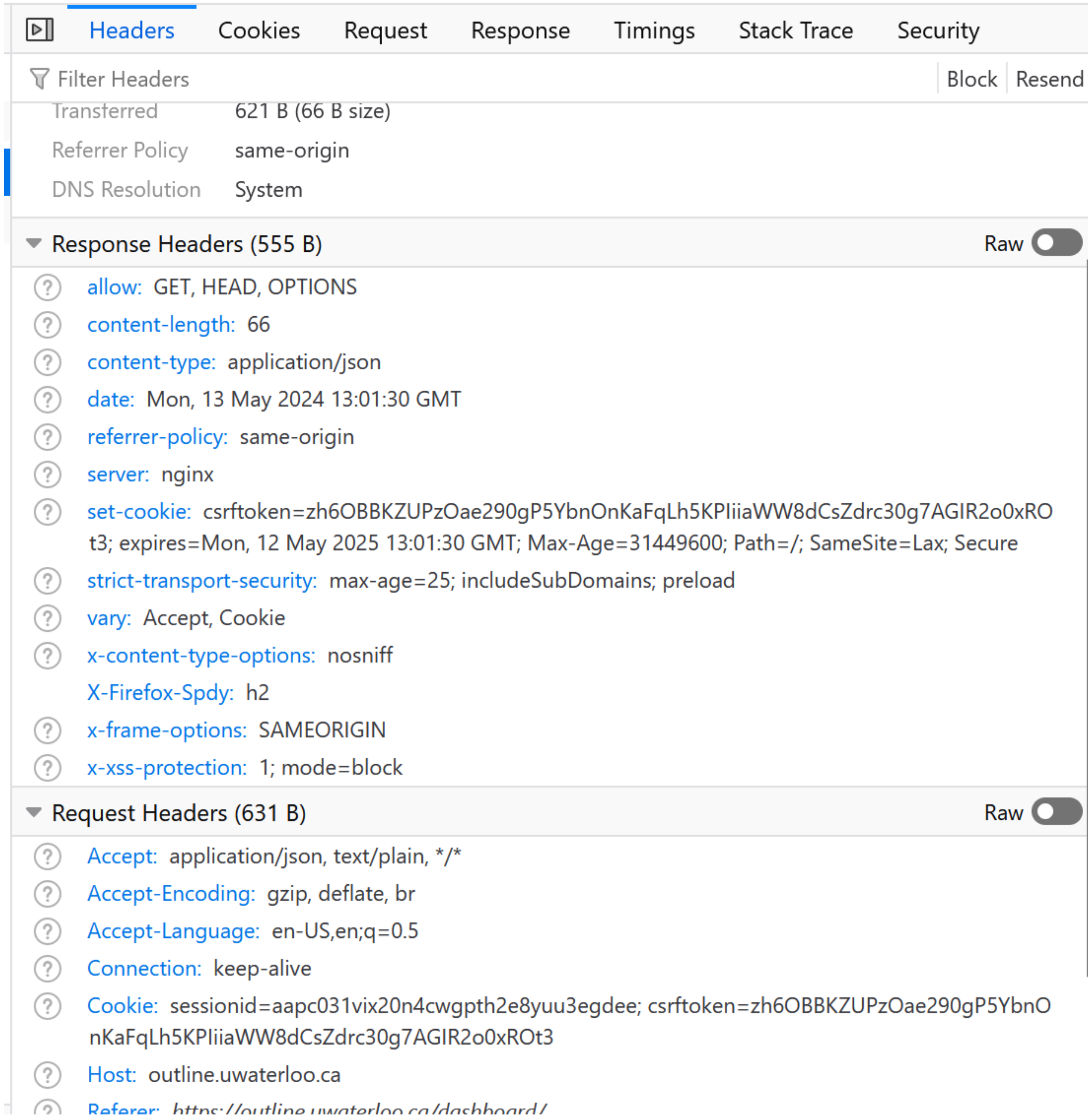
Chip in a credit card

- Similar to RSA fob, the chip generates a unique code
- The user

Access to information sent to your phone number or email

- Having access to something else can be proof of something you have
- Messages sent to your phone number
- Messages sent to your email
- Information in your bank account (how much was deposited)

Your browser or computer may have "something you have" on your behalf



Transferred 621 B (66 B size)
Referrer Policy same-origin
DNS Resolution System

Response Headers (555 B)

```
allow: GET, HEAD, OPTIONS
content-length: 66
content-type: application/json
date: Mon, 13 May 2024 13:01:30 GMT
referrer-policy: same-origin
server: nginx
set-cookie: csrftoken=zh6OBBKZUPzOae290gP5YbnOnKaFqLh5KPIiiaWW8dCsZdrc30g7AGIR2o0xRO
t3; expires=Mon, 12 May 2025 13:01:30 GMT; Max-Age=31449600; Path=/; SameSite=Lax; Secure
strict-transport-security: max-age=25; includeSubDomains; preload
vary: Accept, Cookie
x-content-type-options: nosniff
X-Firefox-Spdy: h2
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Request Headers (631 B)

```
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Cookie: sessionid=aapc031vix20n4cwgpth2e8yuu3egdee; csrftoken=zh6OBBKZUPzOae290gP5YbnO
nKaFqLh5KPIiiaWW8dCsZdrc30g7AGIR2o0xROt3
Host: outline.uwaterloo.ca
Referer: https://outline.uwaterloo.ca/dashboard/
```

A private digital key

- A private key can be "something you have"
- A PGP key is something you have which authenticates you
- For example, if a file is encrypted using the key on the right only I can decrypt it using my matching private key which only I possess

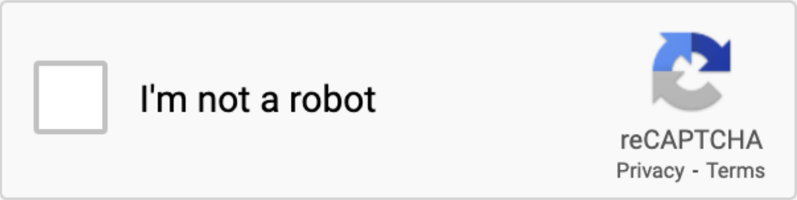
My public key

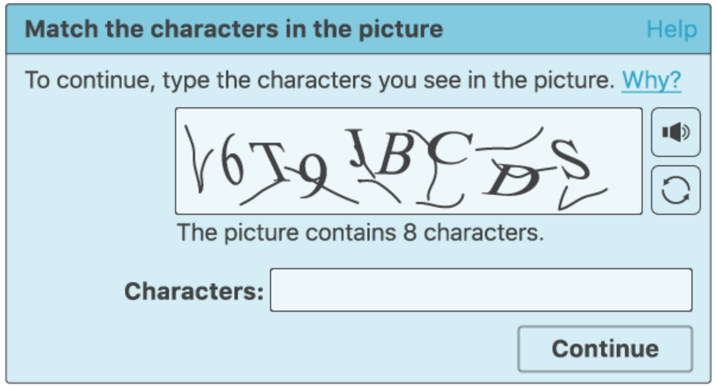
```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2

mQENBFHMcgABCAC9WrYDO6K2L3VHyi4eHN6suHLqMpJ+SO+IUTuLEVnUzIoXAUXH
KozHejfV/9XoG8j933ZtszXKCog3aMESeoEoz6fNGfolvaCe5B4jwqoJt8NHwb5L
B2dnq0CplgXcN2GJxfEHHUaf27COSobCJxPMeshUh4ZHke+g6DatmiEtBpVp41Ot
1zgxdMQkgb2H2xw28RYfYkdDouetelkOrFLrCy9ZF9KdMhA1eBH94KnwIQshdiZR
QYEX25+M8cKCb++Rc9H6an7EG9WHOFRW40UsY52OfveOyfQPzkkRto7u2339hvHo
B/h+7xLM6FQbOUZQ9BD5w7IQHgYtXJVsUjodABEBAAG0IkthbWkgVmFuaWVhIDxr
dmFuaWVhQGluZi5lZC5hYy51az6JAT8EEwEIACkFAlYKYvECGyMFCQlmAYAHCwkI
BwMCAQYVCAIJCgsEFgIDAQIeAQIXgAAKCRCTdsxl9/HZffG+CACShuKxje3QAqew
GWh8K4gCdiYoxDqJwq3PHxmyhZmQeN/1a1KcOrIjI2b+Q75/5t+EgXOHpRoPIxfG
lZ6zOEpf6A18iFXx3JgQZdwPDojtBiWNpOyMeBGTgIvEYG3so2VueQoeXcq3dbYp
5vstVxtD+TKHQ5CioIT75P2bzYq/XLT5aIbNQhQDPcTooDgbRH+FvqsRXr7yeaef
JaPnxXo+1L33t2QY9zctiGyebwrvHMrIPBJ2VYCDzQkJ7uQ5eFh4ZhsMgOmzLQD4
YiGr5weIMFwAvxZOaRxEa9Vf48jiWvrxuJ8YfHWSohEScNOcYC2P8q2olJwwE26T
lpdtrwCqtB1LYW1pIFZhbmllYSA8a2FtaUB2YW5pZWEuY29tPokBQgQTAQIALAIb
IwUJCWYBgAcLCQgHAwIBBhUIAgkKCwQWAgMBAh4BAheABQJWCmMeAhkBAAoJEJN2
zGX38dl9JJAIAIWorxrlYsrmKS6CbW8MgTxxTDOXaCt1b7FoWoQZHskIUQhEcE+a
XBYib1A5uHaatLfyjeXaD3qMEoZnQHoYMGEoGKuoowWsbhfoQzHPgwzRLkD1i75M
BIbawwoKWoVB9e4AkMakXJCnF5BXeo6AHRL2v15V205DikVnlCRXocKtu8b7LnkM
cLn7oLobr1de1uyKoNzbSnO/vpKDJpo/EY5yUeV9oIypZy/6wFQBehg1sXye6znO
9wb9uUsu9+/P8pz4JILMDSevjfT7zSRSl/YP3fOfZ6N4bc+KOdwPM7u5Iyoeu9zh
pzibv3ge7VhH2xIWz8vYZ/2xT1345tWRRMOJAhwEEwECAAYFAlTnSpEACgkQjyxM
p99tBt2B8A/+OpIzOsQbQJB8yxti4I7PpD1weJDf3a81Vhm7JyXE/Xy66ypfdt3w
XmFRUuIrwezY1NebWNCRQHzQvRv/VJwjbTUx+Q3HsjIkKlHbE7iCiQXXtTRkoEny
2nudcjGI2vo3C3B2JCucEw6esF1x79PI/lPv2+6tgUBKmDfOpsB2vbtqrHnmAYKL
4lQBFH1YSJgnzwo2JkhohcHdF9oZem1eMeiDEeVkH63893N8Swk5fBKdTj+SKZ/L
rQElBBlpMR9BmeY6bPvWRuycVKonIMR8oG9iFABxjTpWBL8aGk6EeVK5EqYDGvkd
ZIarK84r+KU1KD5IfgOCN7nhwgy7VImE68caZHSRiPWZP1fVVMhydiRJv8WsoUs6
INfVU3nxH+ZYthPbYoT86leGSchBT5K/fBQvbjhrRTbTFwvjzSifb9efWylDi994
nzP6cNorir3GIpsT8gPgBB2/NjxaWiM6y3X1az1vRnsunQHuyKkFWPZwnEvDJYaC
NN/3jWcbhLFwKBDsaHps2+1meFPooJFvNetzp2bjT9a9pXaQ6KhOmo5DnhLcaV97
bFBpsUuBGaYZTSSo5x1RdXHqpEbgap8dtuHhVvJw9QYDQBJroK4aKyG9qqMD8cta
Pl/FAdyAqwH8Nw9efqAK+RQxSVUaue9BYEnbIRpsDK6MkP3YMFmu5ki5AQoEUcxy
AAEIALyXYy8G2ZaTDJpdGcRhmIqOOSUlzPV7/5E5BbYKBNu4KU3nX+JLVcF5jxPQ
42c7i/WRVxE1BJTiarKGsEvCi94TTXSIUKAt3T1oGBtXmGvqbGBq8ljSGl1UTwdF
5yu5oJyRSf2fqRND6P/2eHNXejDUtdvhUXIUt8h9MuUO/ipDoDnwIvMnAATJHA+R
Zqw6oNpyjRGzvr3iuWUwe4PtyJDI3ELAFkbp/NAc5TIuVHRHNOWNplcIJhM5zHuB
QQb3G/EsCn2PQZ5w5SDzavF2SpvQfDqxYpDaTLAXtF+wsJL5iaUjxwRgJPOdbCZf
2Tozd7h9MXtGJDlPKJ8eLG8ogcMAEQEAAYkBJQQYAQIADwUCUcxyAAIbDAUJCWYB
gAAKCRCTdsxl9/HZfS+hB/9BJqSmIgcoHFXnb1PVIKxekzL8+WVm5Pk/EgMQSLZ2
HX4p3ial5PEPcYgUw9YnaG4iOodwJGw5/daTWRrTzcnKd8YqoP+DUot96HZDSu3m
mCzE9NVAQYboFbVmGOxoeo627UBSvFqaXvAxBDYkoR8BoTnKhrQFwXkZVb3ohKwD
TgAFjOGlZiE6uAdST231tFaqobizYfe5AVXRqro2oxBqNbaJNqs3SWoD831Syvdv
llOBx83/Rogg7hUkI6F2vzXicWmUwFSXRrggCSbLosHsP6isBWwvlHeRmna/aQab
YKG3gbV9iyczAS31gbogVLAZqNSWhp8vVIEE28Fyf/Ed
=x5FK
-----END PGP PUBLIC KEY BLOCK-----
```

Something you are

Something you are

- A property about the person (or device)
- Fingerprints
- Iris scan
- Voice recognition
- Facial recognition
- The way you move your mouse



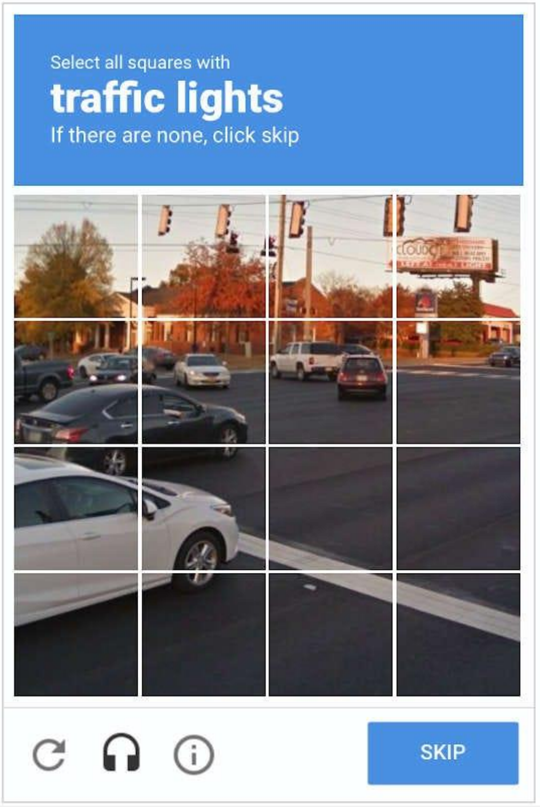


Match the characters in the picture Help

To continue, type the characters you see in the picture. Why?

The picture contains 8 characters.

Characters:



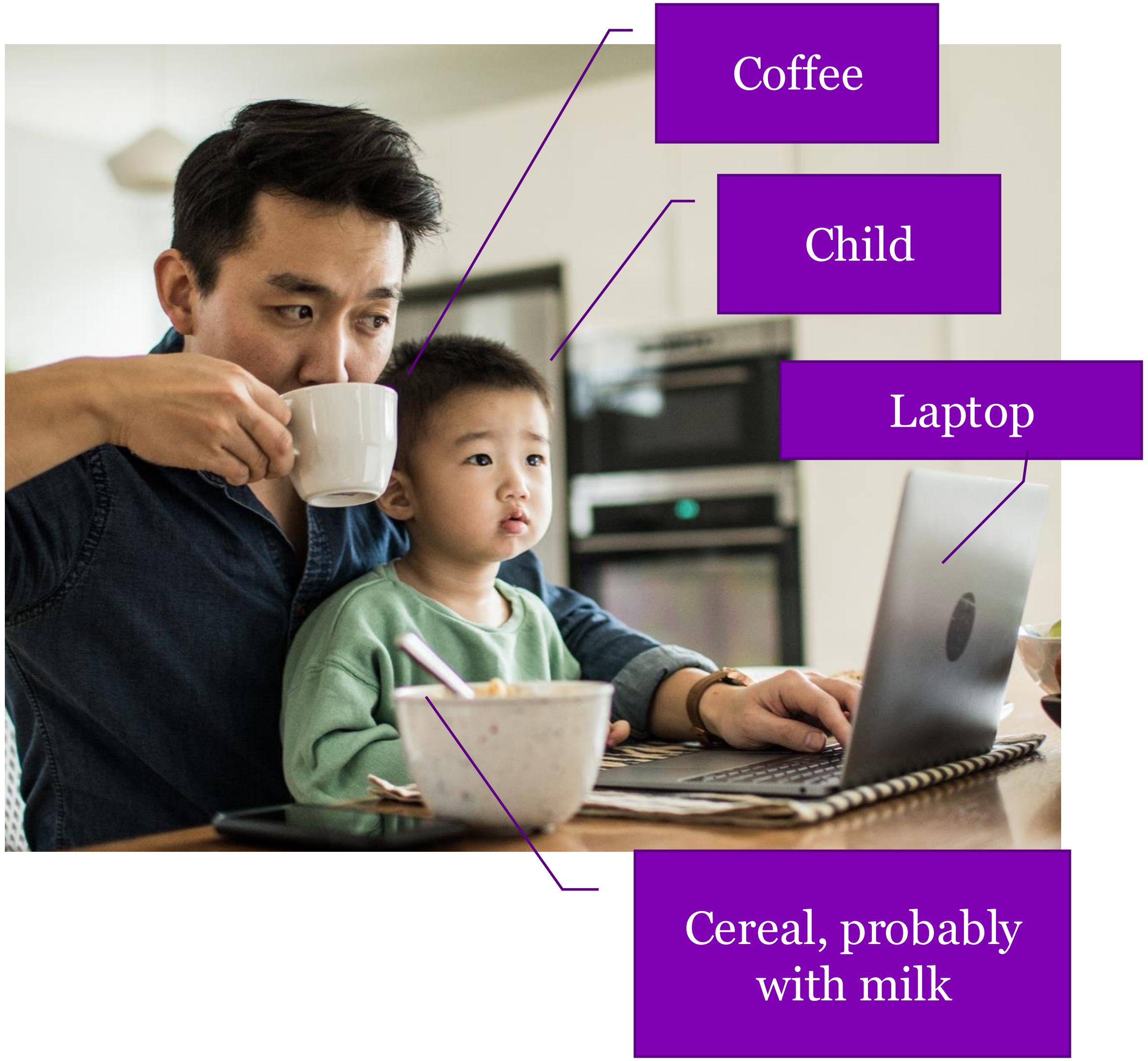
Select all squares with **traffic lights** If there are none, click skip

Fingerprint readers

- Fingerprints are nearly unique so they seem like a good authenticator
- Not all people have fingerprints
  - Some professions destroy fingerprints
  - Some fingerprints are too faint to read
  - Dehydration (from say flying) reduces fingerprint ridges
- Fingerprints can never be changed
- You leave fingerprints everywhere

Continuous authentication

- Your interaction with a computer is unique and we can measure it
  - Mouse movements
  - Keyboard typing patterns
- Nearly impossible to duplicate a real user’s typing patterns
- Easy to lose access if the user hurts their hand, or is doing something non-standard
- Repetitive Stress Injury (RSI) patients trigger continuous authentication warnings regularly while healing



Coffee

Child

Laptop

Cereal, probably with milk

Privacy

- Users have a right to privacy, that is, a right to keep aspects of themselves hidden that are not necessary to expose
- Authentication mechanisms need to take privacy into account and not ask for more than they need
- Identifying a user using a Facebook, Google, or Apple account may be easy, but it gives away large amounts of data
- Similarly, requiring a validated ID such as drivers or passport information also exposes quite a bit of information

Multi-factor authentication

- Combine two of the earlier factors. For example:
  - **Having** a credit card and **knowing** the pin

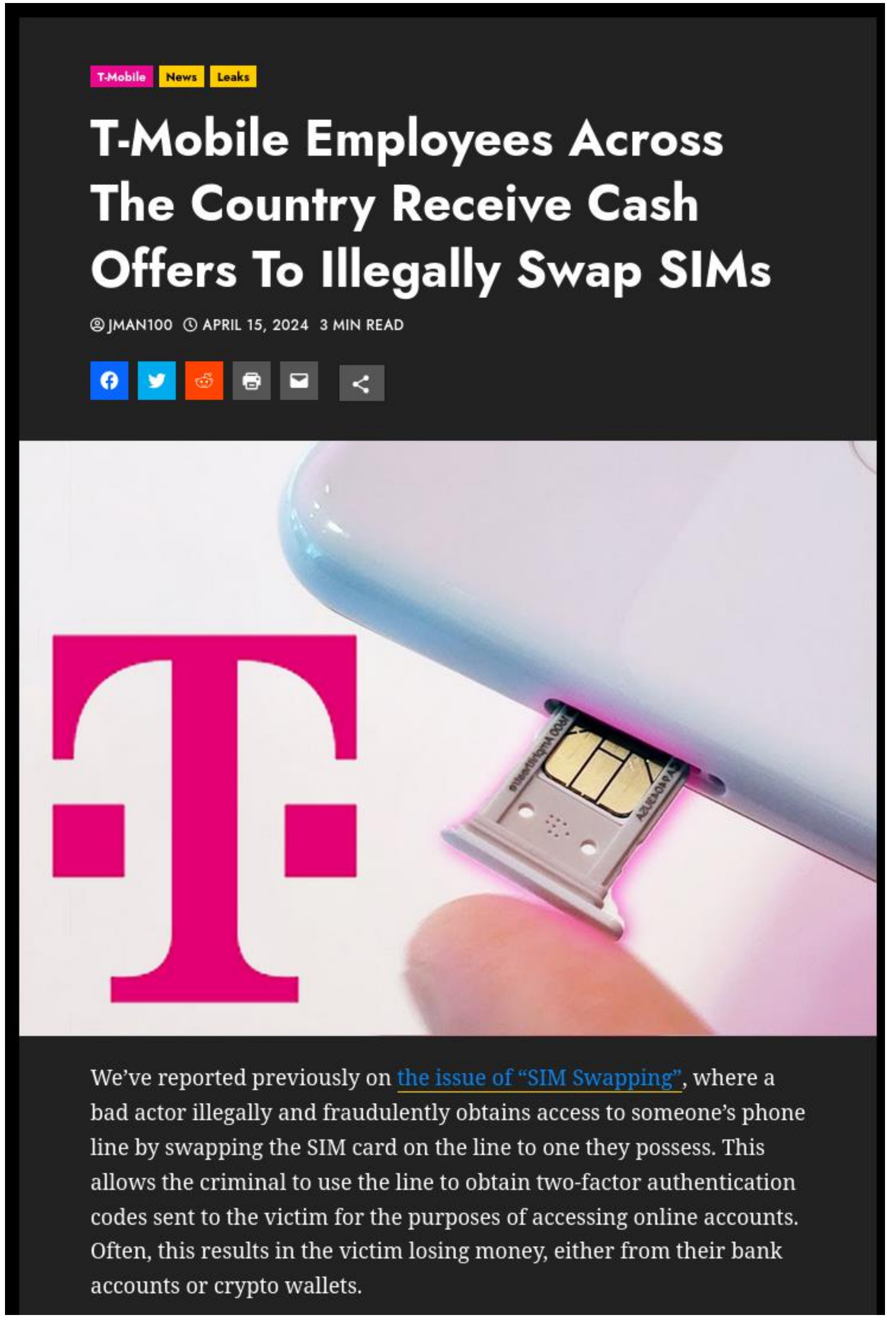
  - **Knowing** a passcode and **being** the person with the correct fingerprints

Multi factor authentication

- Authentication that requires two or more of the factors.
- Two-factor
  - Chip and pin in a credit card. Something you have (chip) something you know (pin).
  - Chip and signature credit card. Something you have (chip) something you are (signature pattern).
- Three-factor
  - Security guard that check's your ID against what you look like and then requires a code.
  - Secure finger print reading fob that gives you a code after it reads your fingerprint, then you use the code and a password to log in.

Think about who/what verifies the second factor

- Phone number destinations can be altered by a large number of people



T-Mobile News Leaks

T-Mobile Employees Across The Country Receive Cash Offers To Illegally Swap SIMs

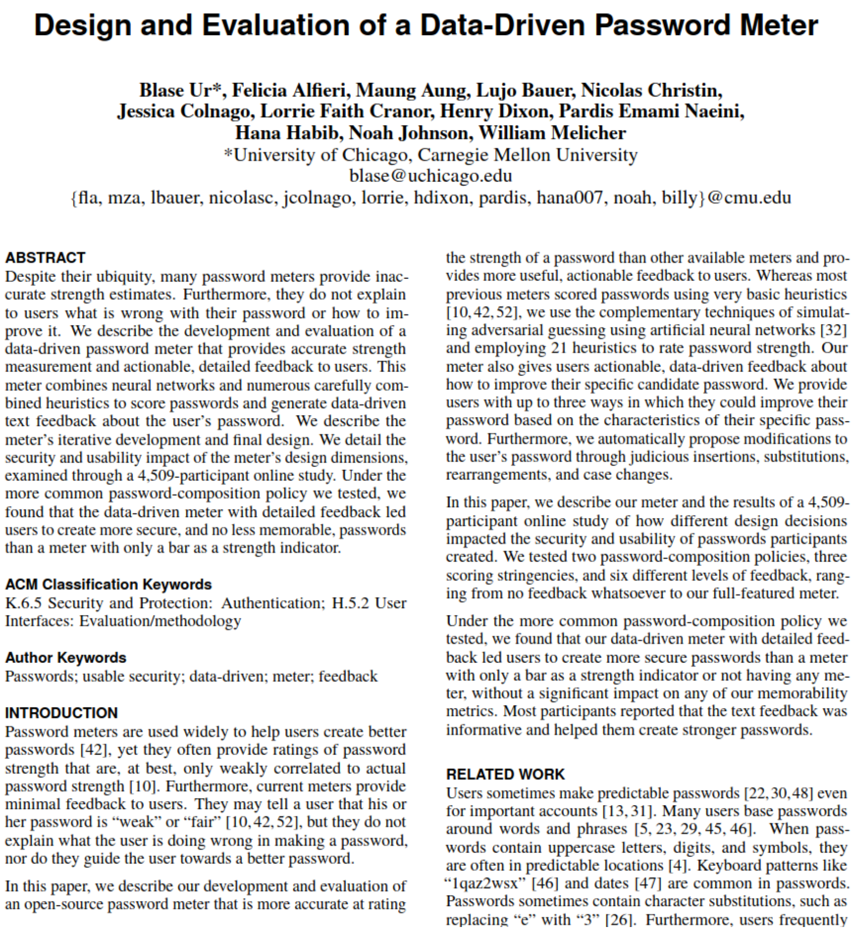
JMAN100 APRIL 15, 2024 3 MIN READ

We've reported previously on the issue of "SIM Swapping", where a bad actor illegally and fraudulently obtains access to someone's phone line by swapping the SIM card on the line to one they possess. This allows the criminal to use the line to obtain two-factor authentication codes sent to the victim for the purposes of accessing online accounts. Often, this results in the victim losing money, either from their bank accounts or crypto wallets.

Authentication is not about verifying your identity.

Authentication verifies that you possess a property.

Two possible papers today:



Design and Evaluation of a Data-Driven Password Meter

Blase Ur*, Felicia Alfieri, Maung Aung, Lujo Bauer, Nicolas Christin,
Jessica Colnago, Lorrie Faith Cranor, Henry Dixon, Pardis Emami Naeini,
Hana Habib, Noah Johnson, William Melicher
*University of Chicago, Carnegie Mellon University
blase@uchicago.edu
{fla, mza, lbauer, nicolasc, jcolnago, lorrie, hdixon, pardis, hana007, noah, billy}@cmu.edu

ABSTRACT

Despite their ubiquity, many password meters provide inaccurate strength estimates. Furthermore, they do not explain to users what is wrong with their password or how to improve it. We describe the development and evaluation of a data-driven password meter that provides accurate strength measurement and actionable, detailed feedback to users. This meter combines neural networks and numerous carefully combined heuristics to score passwords and generate data-driven text feedback about the user's password. We describe the meter's iterative development and final design. We detail the security and usability impact of the meter's design dimensions, examined through a 4,509-participant online study. Under the more common password-composition policy we tested, we found that the data-driven meter with detailed feedback led users to create more secure, and no less memorable, passwords than a meter with only a bar as a strength indicator.

ACM Classification Keywords
K.6.5 Security and Protection: Authentication; H.5.2 User Interfaces: Evaluation/methodology

Author Keywords
Passwords; usable security; data-driven; meter; feedback

INTRODUCTION

Password meters are used widely to help users create better passwords [42], yet they often provide ratings of password strength that are, at best, only weakly correlated to actual password strength [10]. Furthermore, current meters provide minimal feedback to users. They may tell a user that his or her password is "weak" or "fair" [10, 42, 52], but they do not explain what the user is doing wrong in making a password, nor do they guide the user towards a better password.

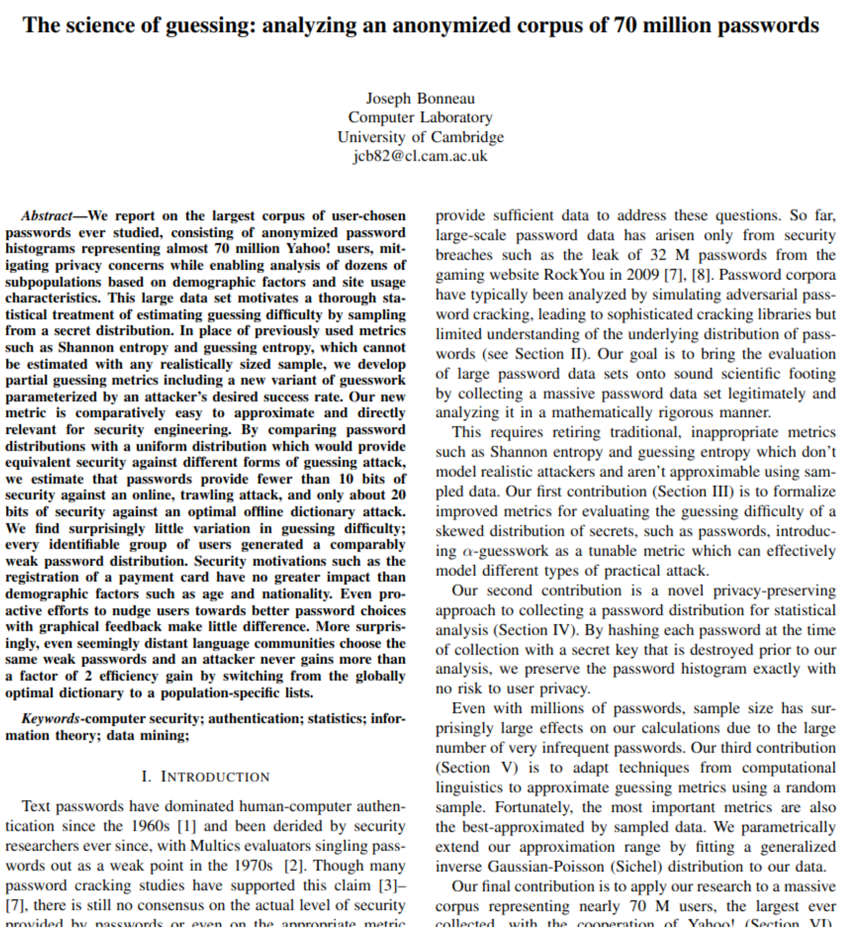
In this paper, we describe our development and evaluation of an open-source password meter that is more accurate at rating the strength of a password than other available meters and provides more useful, actionable feedback to users. Whereas most previous meters scored passwords using very basic heuristics [10, 42, 52], we use the complementary techniques of simulating adversarial guessing using artificial neural networks [32] and employing 21 heuristics to rate password strength. Our meter also gives users actionable, data-driven feedback about how to improve their specific candidate password. We provide users with up to three ways in which they could improve their password based on the characteristics of their specific password. Furthermore, we automatically propose modifications to the user's password through judicious insertions, substitutions, rearrangements, and case changes.

In this paper, we describe our meter and the results of a 4,509-participant online study of how different design decisions impacted the security and usability of passwords participants created. We tested two password-composition policies, three scoring stringencies, and six different levels of feedback, ranging from no feedback whatsoever to our full-featured meter.

Under the more common password-composition policy we tested, we found that our data-driven meter with detailed feedback led users to create more secure passwords than a meter with only a bar as a strength indicator or not having any meter, without a significant impact on any of our memorability metrics. Most participants reported that the text feedback was informative and helped them create stronger passwords.

RELATED WORK

Users sometimes make predictable passwords [22, 30, 48] even for important accounts [13, 31]. Many users base passwords around words and phrases [5, 23, 29, 45, 46]. When passwords contain uppercase letters, digits, and symbols, they are often in predictable locations [4]. Keyboard patterns like "1qaz2wsx" [46] and dates [47] are common in passwords. Passwords sometimes contain character substitutions, such as replacing "e" with "3" [26]. Furthermore, users frequently



The science of guessing: analyzing an anonymized corpus of 70 million passwords

Joseph Bonneau
Computer Laboratory
University of Cambridge
jcb82@cl.cam.ac.uk

***Abstract*—We report on the largest corpus of user-chosen passwords ever studied, consisting of anonymized password histograms representing almost 70 million Yahoo! users, mitigating privacy concerns while enabling analysis of dozens of subpopulations based on demographic factors and site usage characteristics. This large data set motivates a thorough statistical treatment of estimating guessing difficulty by sampling from a secret distribution. In place of previously used metrics such as Shannon entropy and guessing entropy, which cannot be estimated with any realistically sized sample, we develop partial guessing metrics including a new variant of guesswork parameterized by an attacker's desired success rate. Our new metric is comparatively easy to approximate and directly relevant for security engineering. By comparing password distributions with a uniform distribution which would provide equivalent security against different forms of guessing attack, we estimate that passwords provide fewer than 10 bits of security against an online, trawling attack, and only about 20 bits of security against an optimal offline dictionary attack. We find surprisingly little variation in guessing difficulty; every identifiable group of users generated a comparably weak password distribution. Security motivations such as the registration of a payment card have no greater impact than demographic factors such as age and nationality. Even proactive efforts to nudge users towards better password choices with graphical feedback make little difference. More surprisingly, even seemingly distant language communities choose the same weak passwords and an attacker never gains more than a factor of 2 efficiency gain by switching from the globally optimal dictionary to a population-specific lists.**

***Keywords*-computer security; authentication; statistics; information theory; data mining;**

I. Introduction

Text passwords have dominated human-computer authentication since the 1960s [1] and been derided by security researchers ever since, with Multics evaluators singling passwords out as a weak point in the 1970s [2]. Though many password cracking studies have supported this claim [3]–[7], there is still no consensus on the actual level of security provided by passwords or even on the appropriate metric

provide sufficient data to address these questions. So far, large-scale password data has arisen only from security breaches such as the leak of 32 M passwords from the gaming website RockYou in 2009 [7], [8]. Password corpora have typically been analyzed by simulating adversarial password cracking, leading to sophisticated cracking libraries but limited understanding of the underlying distribution of passwords (see Section II). Our goal is to bring the evaluation of large password data sets onto sound scientific footing by collecting a massive password data set legitimately and analyzing it in a mathematically rigorous manner.

This requires retiring traditional, inappropriate metrics such as Shannon entropy and guessing entropy which don't model realistic attackers and aren't approximable using sampled data. Our first contribution (Section III) is to formalize improved metrics for evaluating the guessing difficulty of a skewed distribution of secrets, such as passwords, introducing α-guesswork as a tunable metric which can effectively model different types of practical attack.

Our second contribution is a novel privacy-preserving approach to collecting a password distribution for statistical analysis (Section IV). By hashing each password at the time of collection with a secret key that is destroyed prior to our analysis, we preserve the password histogram exactly with no risk to user privacy.

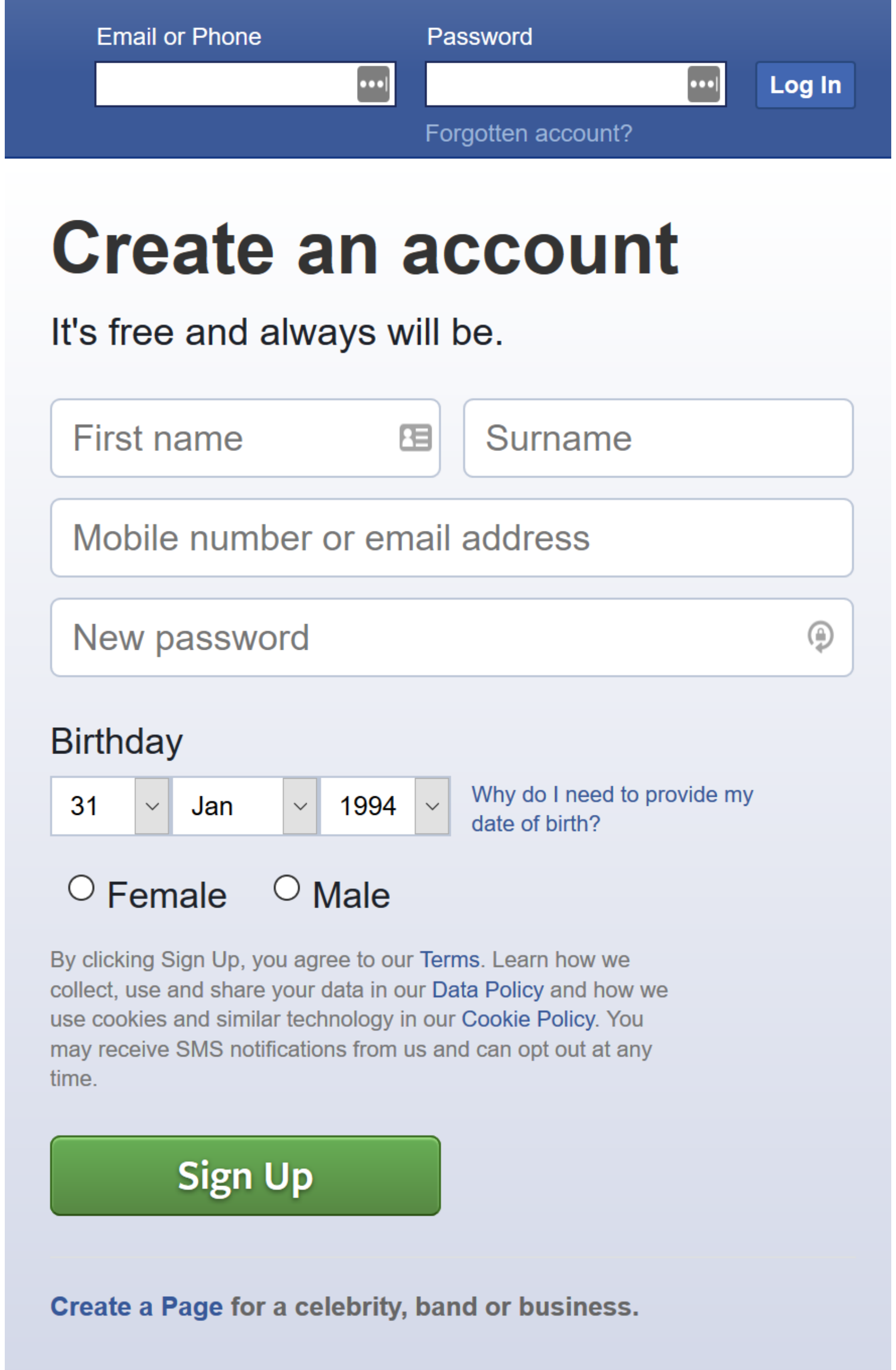
Even with millions of passwords, sample size has surprisingly large effects on our calculations due to the large number of very infrequent passwords. Our third contribution (Section V) is to adapt techniques from computational linguistics to approximate guessing metrics using a random sample. Fortunately, the most important metrics are also the best-approximated by sampled data. We parametrically extend our approximation range by fitting a generalized inverse Gaussian-Poisson (Sichel) distribution to our data.

Our final contribution is to apply our research to a massive corpus representing nearly 70 M users, the largest ever collected, with the cooperation of Yahoo! (Section VI)

PASSWORDS

The “create an account” is collecting information on the first interaction.

Then using the password to verify that you are the same person next time you log in.



Email or Phone

Password

Log In

Forgotten account?

Create an account

It's free and always will be.

First name

Surname

Mobile number or email address

New password

Birthday

31 | Jan | 1994

Why do I need to provide my date of birth?

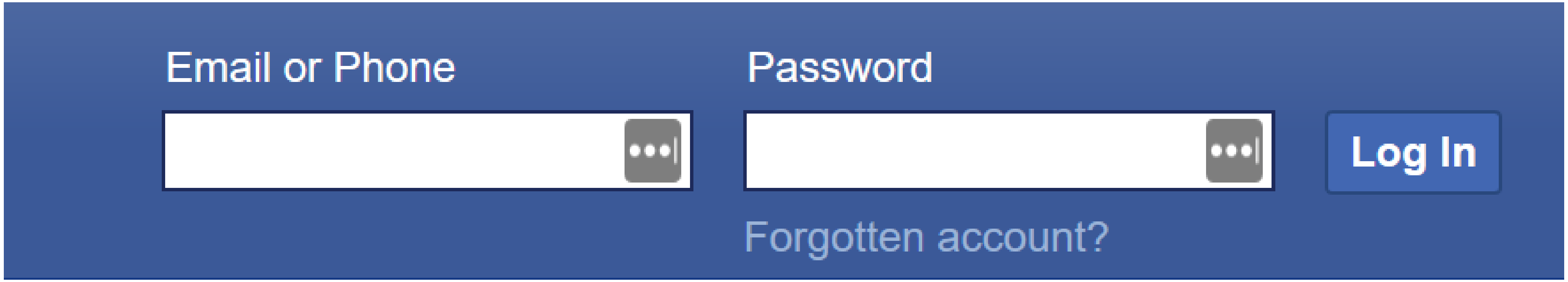
○ Female ○ Male

By clicking Sign Up, you agree to our Terms. Learn how we collect, use and share your data in our Data Policy and how we use cookies and similar technology in our Cookie Policy. You may receive SMS notifications from us and can opt out at any time.

Sign Up

Create a Page for a celebrity, band or business.

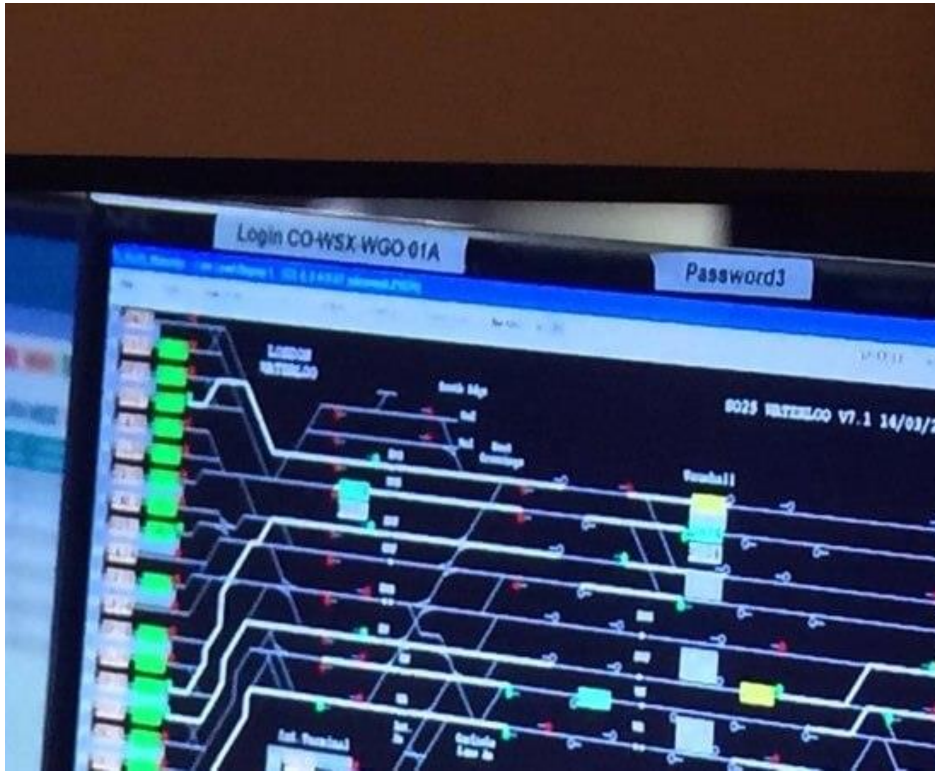
Passwords



- User enters userID and password
- Client sends userID and password to server
- Server hashes the password
- If hashed password matches the hashed password on file
  - Send a unique code back to the client as proof of authentication

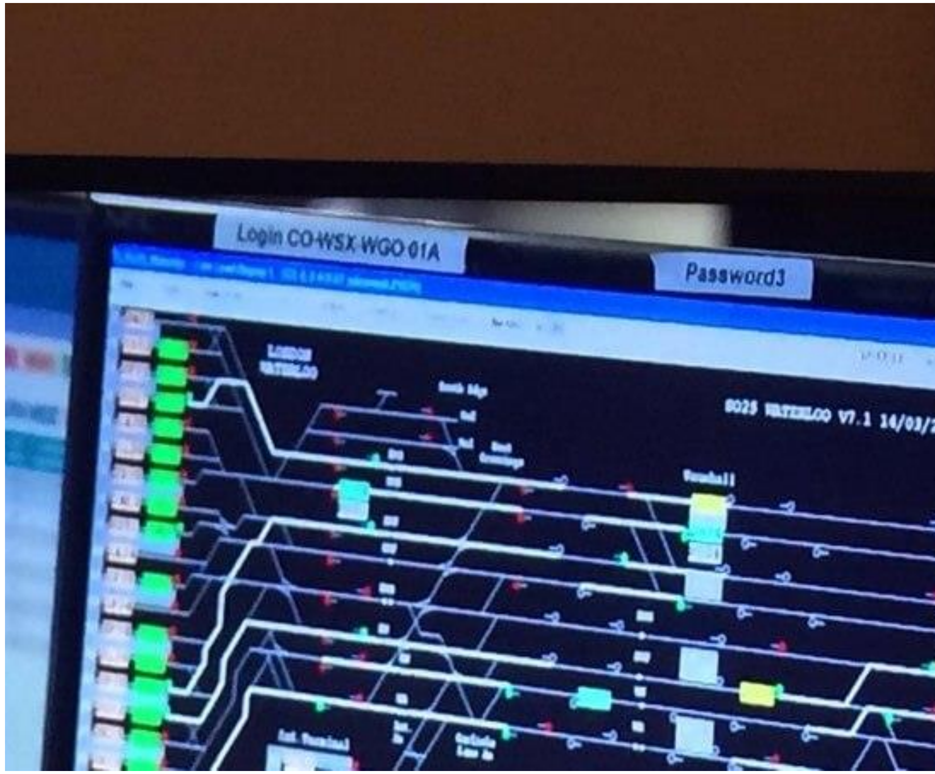
How can an attacker get a password?

- **Trick** someone into giving it to the attacker (i.e. phishing)
- **Steal** it from an unsecured place (i.e. sticky note on monitor)
- **Guess** it by entering userIDs and passwords into the login box till they login – Online attack
- **Steal hashes** + **guess** by compromising a computer, stealing the password hash file, and then hashing guessed passwords + salts till the resulting hash matches



Guessing

- **Online attack:** guess password on a live website by entering userIDs and passwords into the login box till they login – Online attack
- **Offline attack:** steal the password hash file, and then guess by hashing passwords + salts till they get a match



"Stronger" passwords are better.

What does "strong" mean?

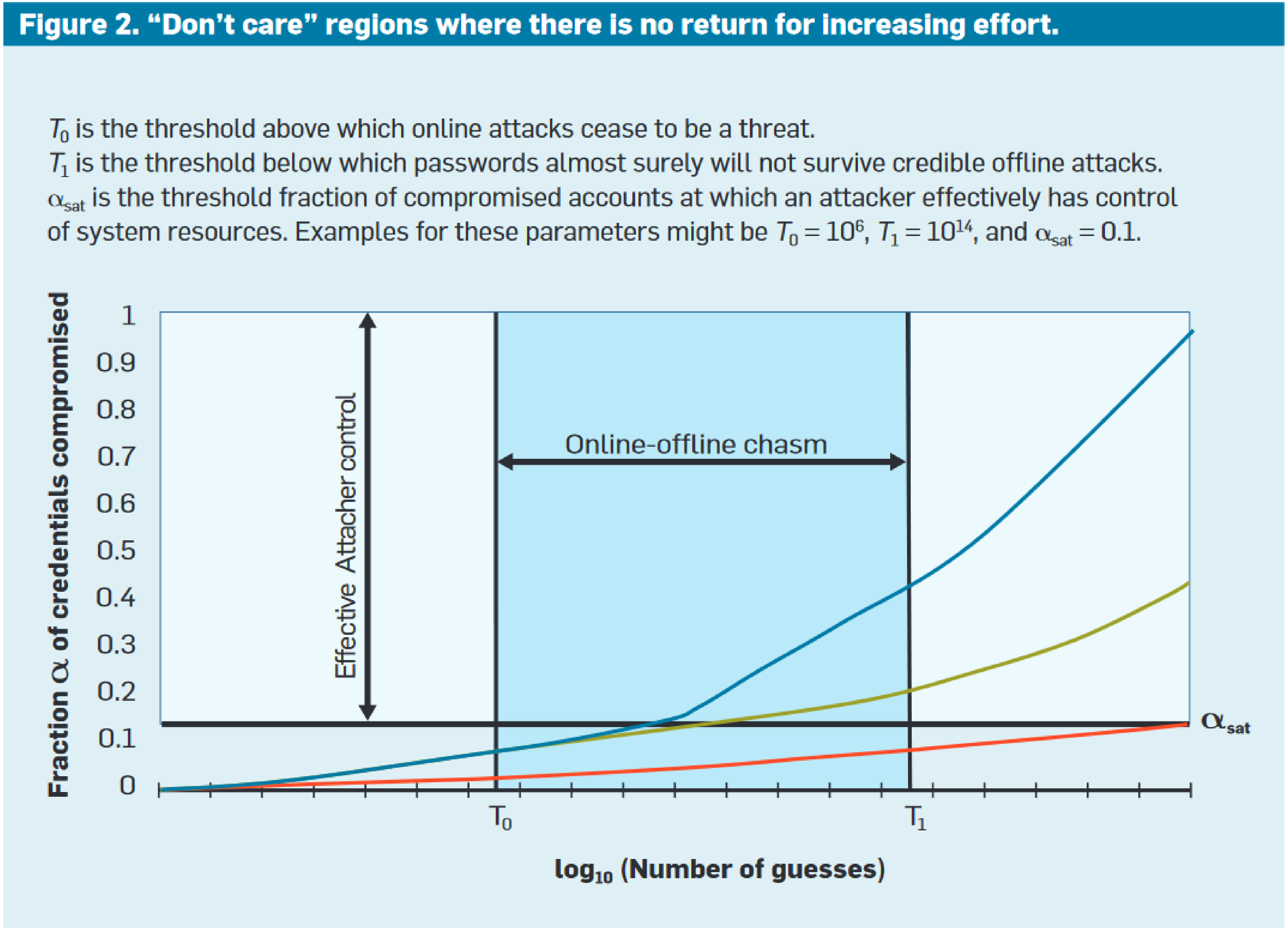


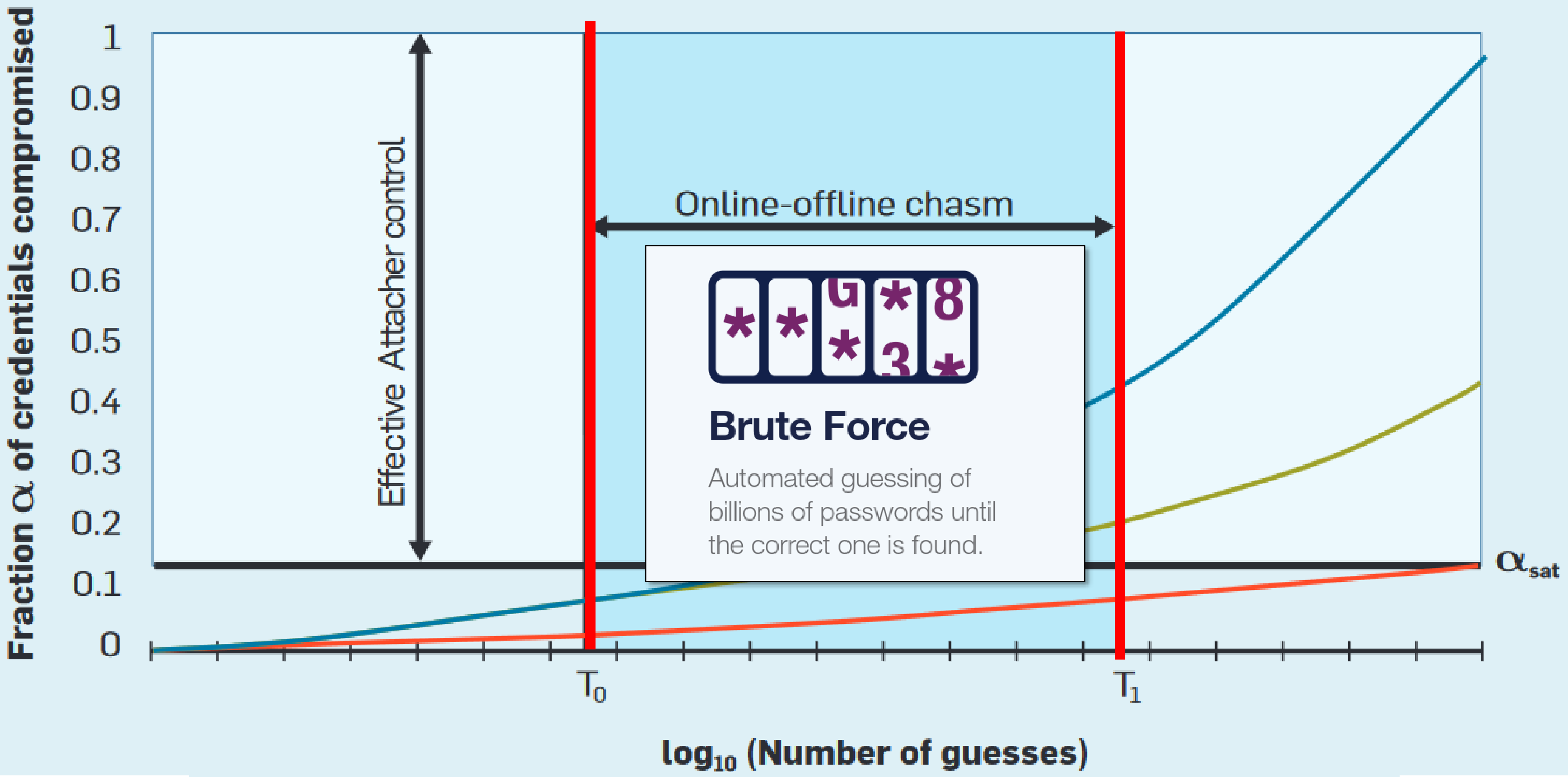
Figure 2. “Don’t care” regions where there is no return for increasing effort.

T_0 is the threshold above which online attacks cease to be a threat.
T_1 is the threshold below which passwords almost surely will not survive credible offline attacks.
α_{sat} is the threshold fraction of compromised accounts at which an attacker effectively has control of system resources. Examples for these parameters might be $T_0 = 10^6$, $T_1 = 10^{14}$, and $\alpha_{sat} = 0.1$.

Fraction α of credentials compromised
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Effective Attacher control
Online-offline chasm
α_{sat}
T_0
T_1
$\log_{10}$ (Number of guesses)

Pushing on string: The don't care region of password strength, D Florêncio, C Herley, PC Van Oorschot - Communications of the ACM, 2016

Figure 2. “Don’t care” regions where there is no return for increasing effort.



Automated guessing of billions of passwords until the correct one is found.

Pushing on string: The don't care region of password strength, D Florêncio, C Herley, PC Van Oorschot - Communications of the ACM, 2016

Online Attack Mitigation: Lockout

- Password guessing attacks work because a computer can guess many times a second
- Humans don’t guess many times a second
- One way to protect against online attacks is to rate limit password attempts
- If a user cannot guess a password in 10 tries or less, lock them out for a short time OR require another factor
- Rate limiting also works, if they fail a password entry attempt, add a short loading delay, if they fail twice, double the delay, another failure, double it again. Delays of <1s a human won’t notice, but will slow a computer down.

Offline Attack Mitigation: Hash+Salt

- If no salt is used, an attacker can pre-compute a “rainbow table” listing each password/hash combination
- A salt adds a random string to the password that is different for every password
- Even with the hash file, the attacker must still compute the hash for each password guess which takes longer than a lookup

Hash (SHA1)	Password
5BAA61E4C9B93F3F0682250B6CF8331B7EE68FD8	password
7C4A8D09CA3762AF61E59520943DC26494F8941B	123456
AB87D24BDC7452E55738DEB5F868E1F16DEA5ACE	monkey
9FC60FFF2273806ECA45B9681041AE95E9652E0D	poodle
EE8D8728F435FD550F83852AABAB5234CE1DA528	iloveyou
02726D40F378E716981C4321D60BA3A325ED6A4C	Pa$$word
783573016CC34B120833D5282CC189A2E059771A	waterloo123

Think-pair-share

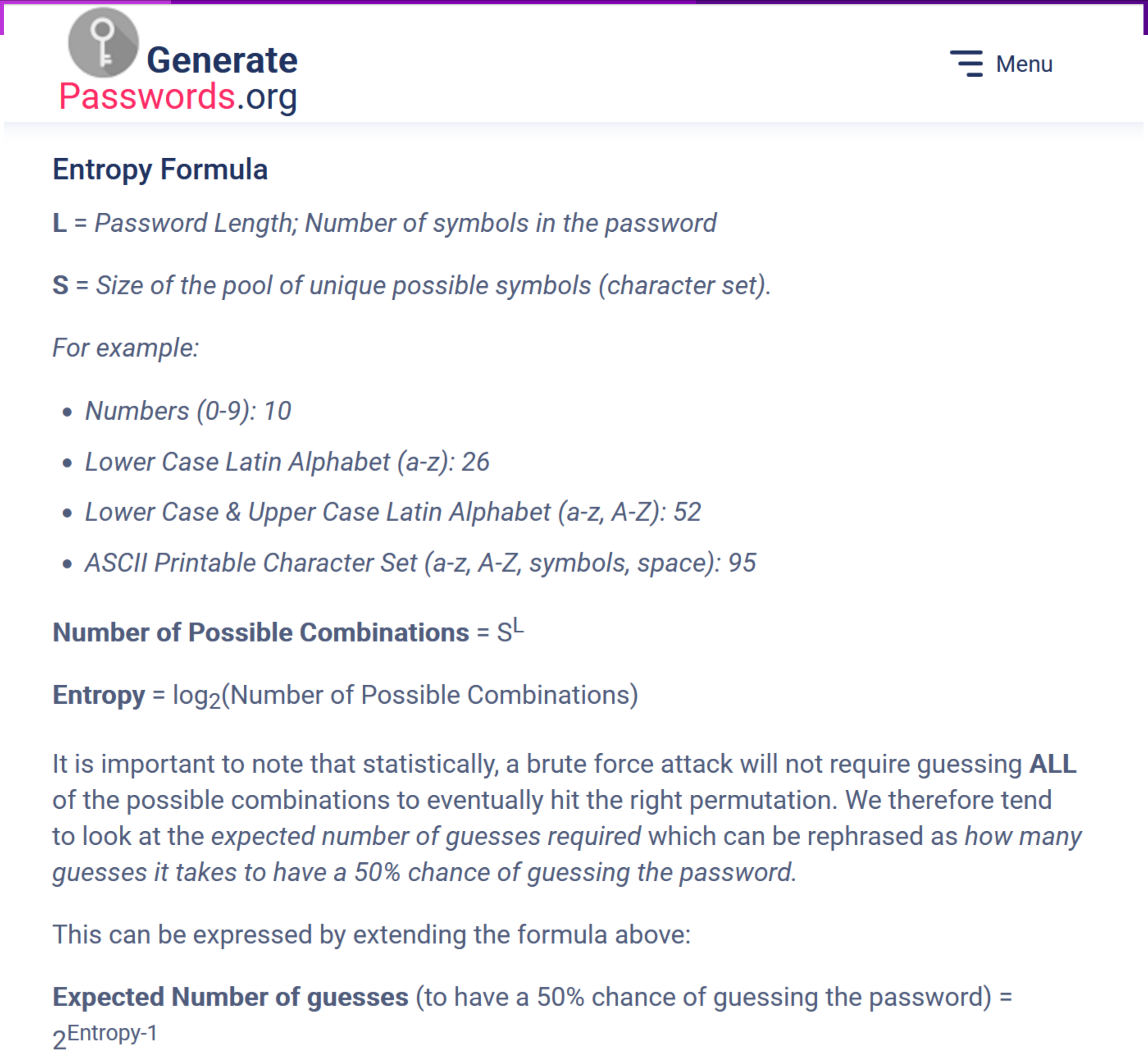
- What properties are we looking for in a good password?

- What security impact are good passwords trying to reach?

- How should password strength be defined?
  - I do not mean requirements around upper/lower case. What higher level properties are we trying to attain?

Entropy

- Roughly entropy is a calculation of how big the space of possible passwords is.
- In theory, a bigger space makes guessing harder
- But only if passwords are evenly balanced over the space....



Entropy Formula

L = *Password Length; Number of symbols in the password*

S = *Size of the pool of unique possible symbols (character set).*

For example:

- *Numbers (0-9): 10*
- *Lower Case Latin Alphabet (a-z): 26*
- *Lower Case & Upper Case Latin Alphabet (a-z, A-Z): 52*
- *ASCII Printable Character Set (a-z, A-Z, symbols, space): 95*

Number of Possible Combinations = S^L

Entropy = $\log_2$(Number of Possible Combinations)

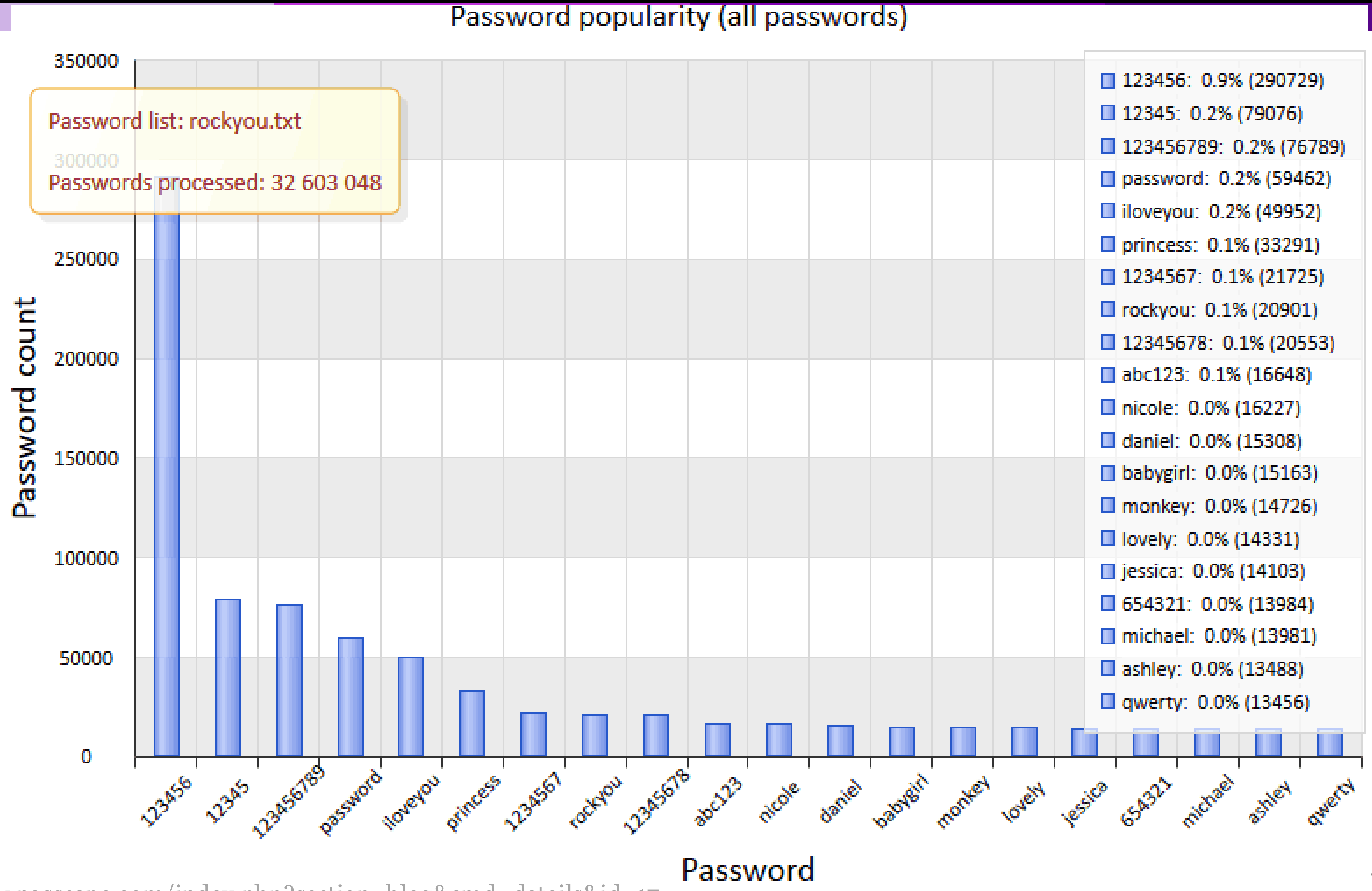
It is important to note that statistically, a brute force attack will not require guessing **ALL** of the possible combinations to eventually hit the right permutation. We therefore tend to look at the *expected number of guesses required* which can be rephrased as *how many guesses it takes to have a 50% chance of guessing the password.*

This can be expressed by extending the formula above:

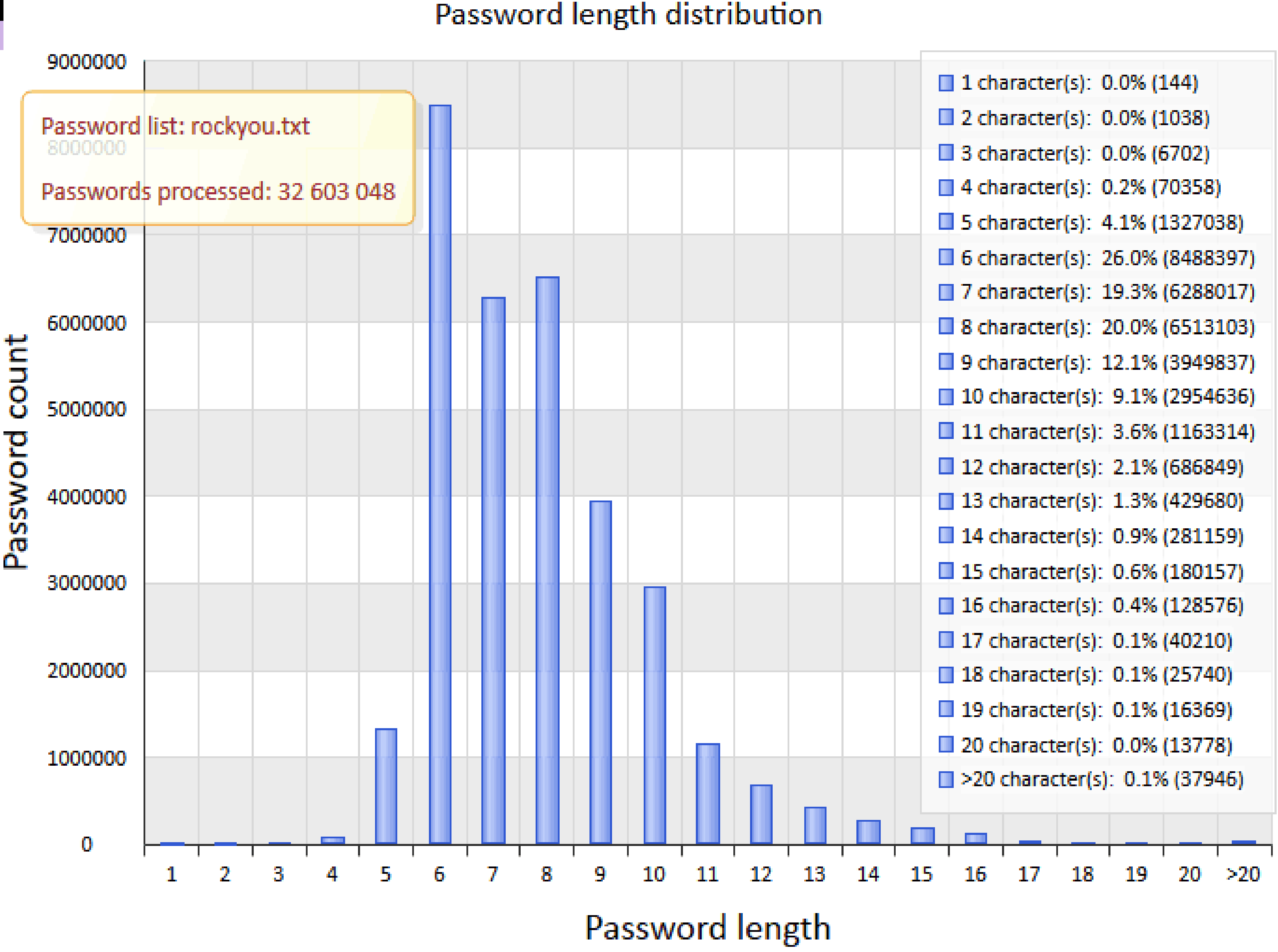
Expected Number of guesses (to have a 50% chance of guessing the password) = $2^{\text{Entropy-1}}$

Password space

- 1 character passwords made of only ASCII letters:
  - 26^1 possible passwords
- 8 characters passwords made of only ASCII letters:
  - 26^8 possible passwords
- 8 character password made of ASCII letters + 10 digits:
  - 36^8 possible passwords



https://www.passcape.com/index.php?section=blog&cmd=details&id=17



Password list: rockyou.txt

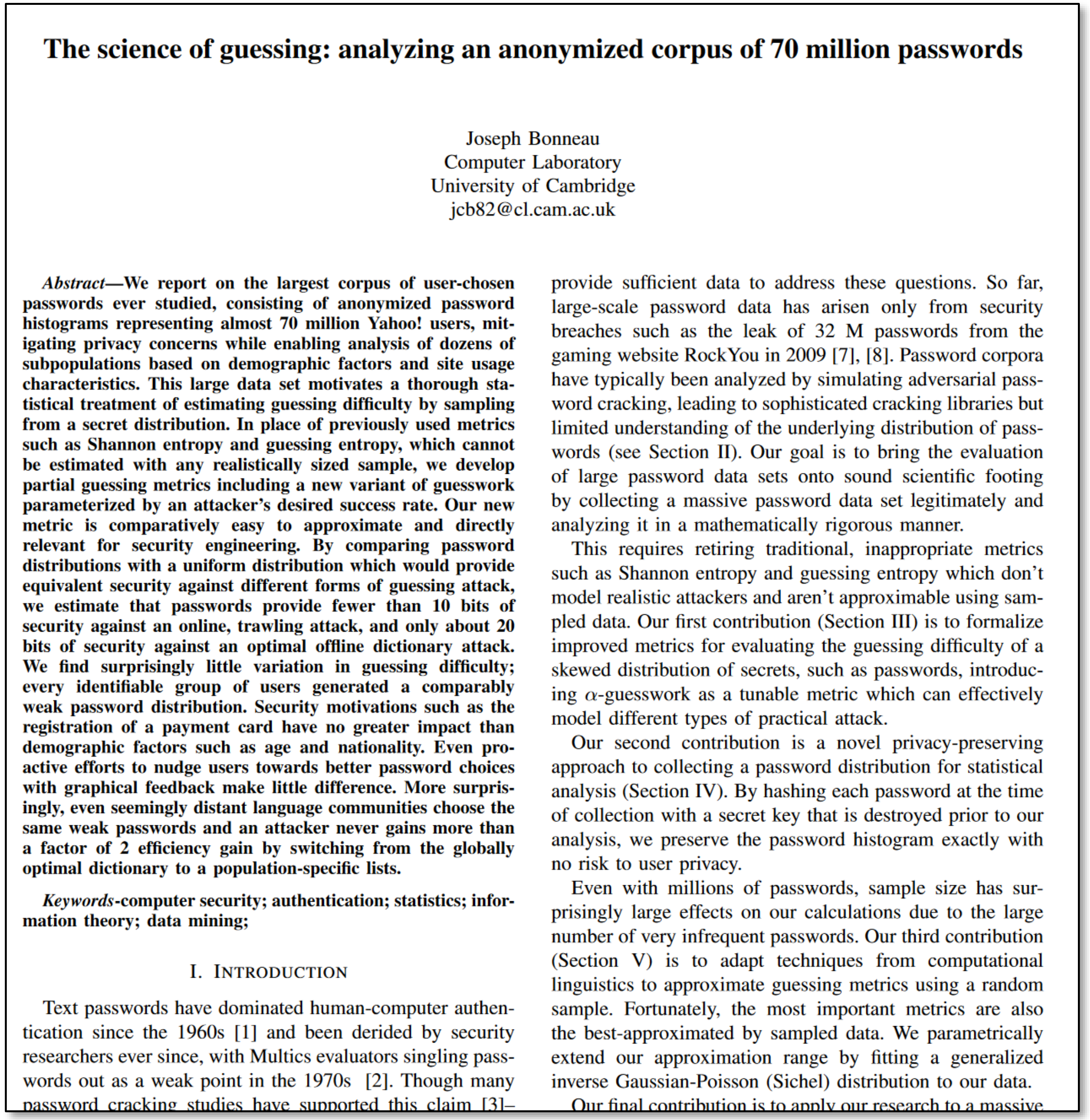
Passwords processed: 32 603 048

https://www.passcape.com/index.php?section=blog&cmd=details&id=17

PASSWORD STRENGTH

Paper: The science of guessing: analyzing an anonymized corpus of 70 million passwords

- How strong are passwords assuming that an attacker knows the frequency distribution X of human generated passwords or a close approximation X'?
- How should strength be defined and hence computed?



The science of guessing: analyzing an anonymized corpus of 70 million passwords

Joseph Bonneau
Computer Laboratory
University of Cambridge
jcb82@cl.cam.ac.uk

***Abstract*—We report on the largest corpus of user-chosen passwords ever studied, consisting of anonymized password histograms representing almost 70 million Yahoo! users, mitigating privacy concerns while enabling analysis of dozens of subpopulations based on demographic factors and site usage characteristics. This large data set motivates a thorough statistical treatment of estimating guessing difficulty by sampling from a secret distribution. In place of previously used metrics such as Shannon entropy and guessing entropy, which cannot be estimated with any realistically sized sample, we develop partial guessing metrics including a new variant of guesswork parameterized by an attacker's desired success rate. Our new metric is comparatively easy to approximate and directly relevant for security engineering. By comparing password distributions with a uniform distribution which would provide equivalent security against different forms of guessing attack, we estimate that passwords provide fewer than 10 bits of security against an online, trawling attack, and only about 20 bits of security against an optimal offline dictionary attack. We find surprisingly little variation in guessing difficulty; every identifiable group of users generated a comparably weak password distribution. Security motivations such as the registration of a payment card have no greater impact than demographic factors such as age and nationality. Even proactive efforts to nudge users towards better password choices with graphical feedback make little difference. More surprisingly, even seemingly distant language communities choose the same weak passwords and an attacker never gains more than a factor of 2 efficiency gain by switching from the globally optimal dictionary to a population-specific lists.**

***Keywords*-computer security; authentication; statistics; information theory; data mining;**

I. Introduction

Text passwords have dominated human-computer authentication since the 1960s [1] and been derided by security researchers ever since, with Multics evaluators singling passwords out as a weak point in the 1970s [2]. Though many password cracking studies have supported this claim [3]–

provide sufficient data to address these questions. So far, large-scale password data has arisen only from security breaches such as the leak of 32 M passwords from the gaming website RockYou in 2009 [7], [8]. Password corpora have typically been analyzed by simulating adversarial password cracking, leading to sophisticated cracking libraries but limited understanding of the underlying distribution of passwords (see Section II). Our goal is to bring the evaluation of large password data sets onto sound scientific footing by collecting a massive password data set legitimately and analyzing it in a mathematically rigorous manner.

This requires retiring traditional, inappropriate metrics such as Shannon entropy and guessing entropy which don't model realistic attackers and aren't approximable using sampled data. Our first contribution (Section III) is to formalize improved metrics for evaluating the guessing difficulty of a skewed distribution of secrets, such as passwords, introducing α-guesswork as a tunable metric which can effectively model different types of practical attack.

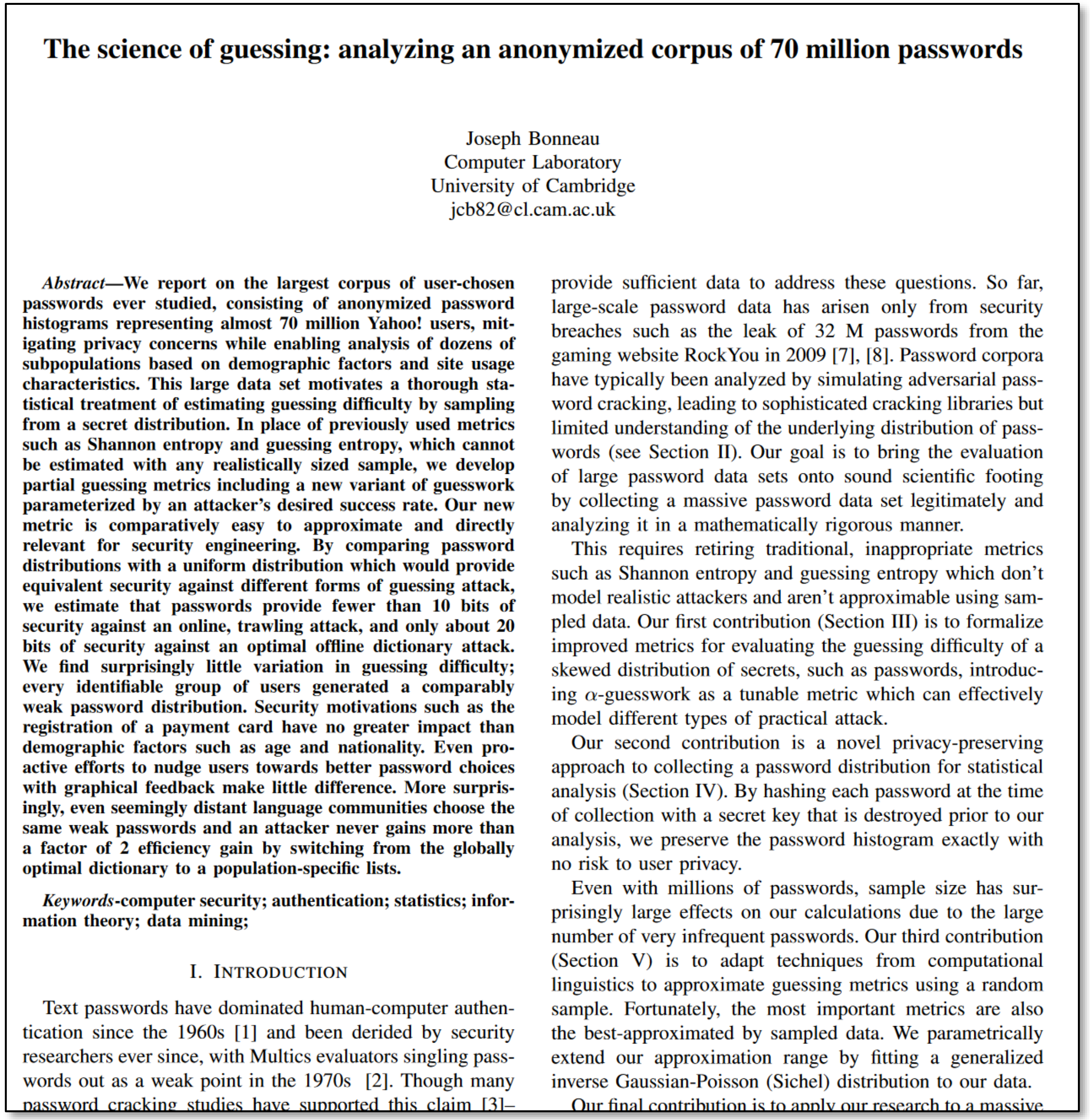
Our second contribution is a novel privacy-preserving approach to collecting a password distribution for statistical analysis (Section IV). By hashing each password at the time of collection with a secret key that is destroyed prior to our analysis, we preserve the password histogram exactly with no risk to user privacy.

Even with millions of passwords, sample size has surprisingly large effects on our calculations due to the large number of very infrequent passwords. Our third contribution (Section V) is to adapt techniques from computational linguistics to approximate guessing metrics using a random sample. Fortunately, the most important metrics are also the best-approximated by sampled data. We parametrically extend our approximation range by fitting a generalized inverse Gaussian-Poisson (Sichel) distribution to our data.

Our final contribution is to apply our research to a massive

What can we tell about this paper from just meta data?

- Joseph Bonneau. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In Proceedings of IEEE SP 2012.



The science of guessing: analyzing an anonymized corpus of 70 million passwords

Joseph Bonneau
Computer Laboratory
University of Cambridge
jcb82@cl.cam.ac.uk

Abstract—**We report on the largest corpus of user-chosen passwords ever studied, consisting of anonymized password histograms representing almost 70 million Yahoo! users, mitigating privacy concerns while enabling analysis of dozens of subpopulations based on demographic factors and site usage characteristics. This large data set motivates a thorough statistical treatment of estimating guessing difficulty by sampling from a secret distribution. In place of previously used metrics such as Shannon entropy and guessing entropy, which cannot be estimated with any realistically sized sample, we develop partial guessing metrics including a new variant of guesswork parameterized by an attacker's desired success rate. Our new metric is comparatively easy to approximate and directly relevant for security engineering. By comparing password distributions with a uniform distribution which would provide equivalent security against different forms of guessing attack, we estimate that passwords provide fewer than 10 bits of security against an online, trawling attack, and only about 20 bits of security against an optimal offline dictionary attack. We find surprisingly little variation in guessing difficulty; every identifiable group of users generated a comparably weak password distribution. Security motivations such as the registration of a payment card have no greater impact than demographic factors such as age and nationality. Even proactive efforts to nudge users towards better password choices with graphical feedback make little difference. More surprisingly, even seemingly distant language communities choose the same weak passwords and an attacker never gains more than a factor of 2 efficiency gain by switching from the globally optimal dictionary to a population-specific lists.**

Keywords-**computer security; authentication; statistics; information theory; data mining;**

I. Introduction

Text passwords have dominated human-computer authentication since the 1960s [1] and been derided by security researchers ever since, with Multics evaluators singling passwords out as a weak point in the 1970s [2]. Though many password cracking studies have supported this claim [3]–

provide sufficient data to address these questions. So far, large-scale password data has arisen only from security breaches such as the leak of 32 M passwords from the gaming website RockYou in 2009 [7], [8]. Password corpora have typically been analyzed by simulating adversarial password cracking, leading to sophisticated cracking libraries but limited understanding of the underlying distribution of passwords (see Section II). Our goal is to bring the evaluation of large password data sets onto sound scientific footing by collecting a massive password data set legitimately and analyzing it in a mathematically rigorous manner.

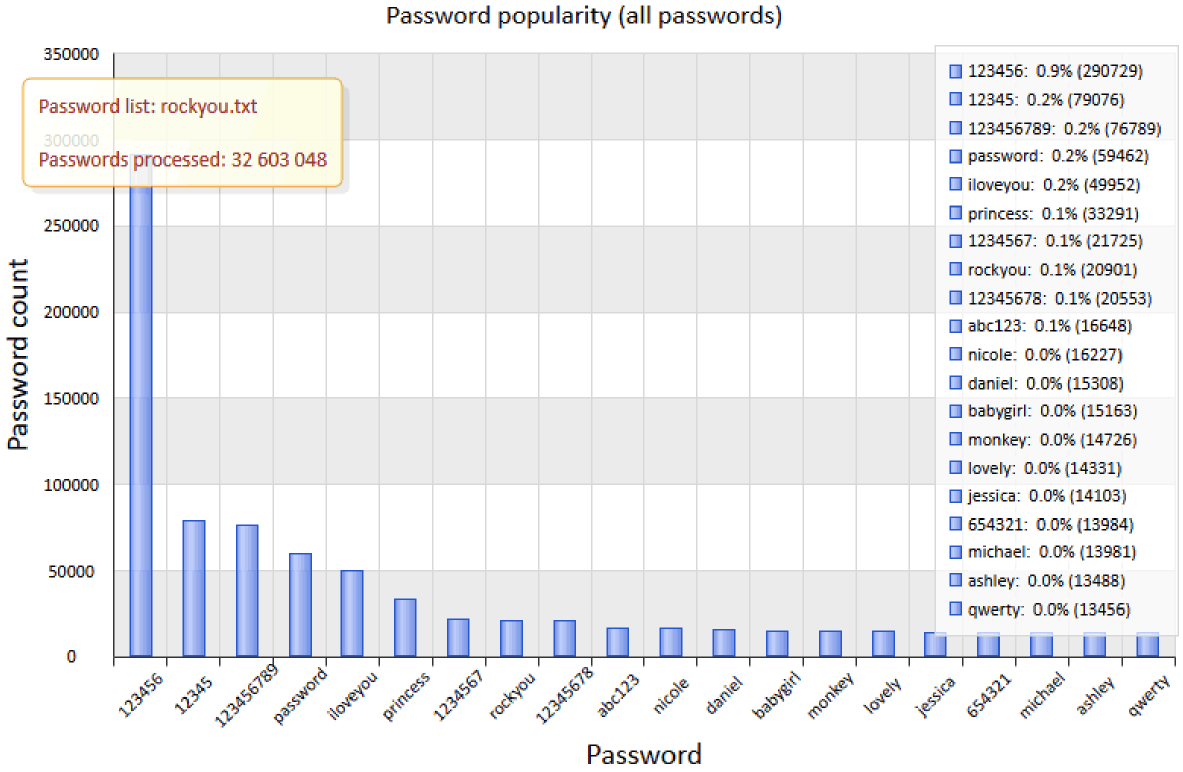
This requires retiring traditional, inappropriate metrics such as Shannon entropy and guessing entropy which don't model realistic attackers and aren't approximable using sampled data. Our first contribution (Section III) is to formalize improved metrics for evaluating the guessing difficulty of a skewed distribution of secrets, such as passwords, introducing α-guesswork as a tunable metric which can effectively model different types of practical attack.

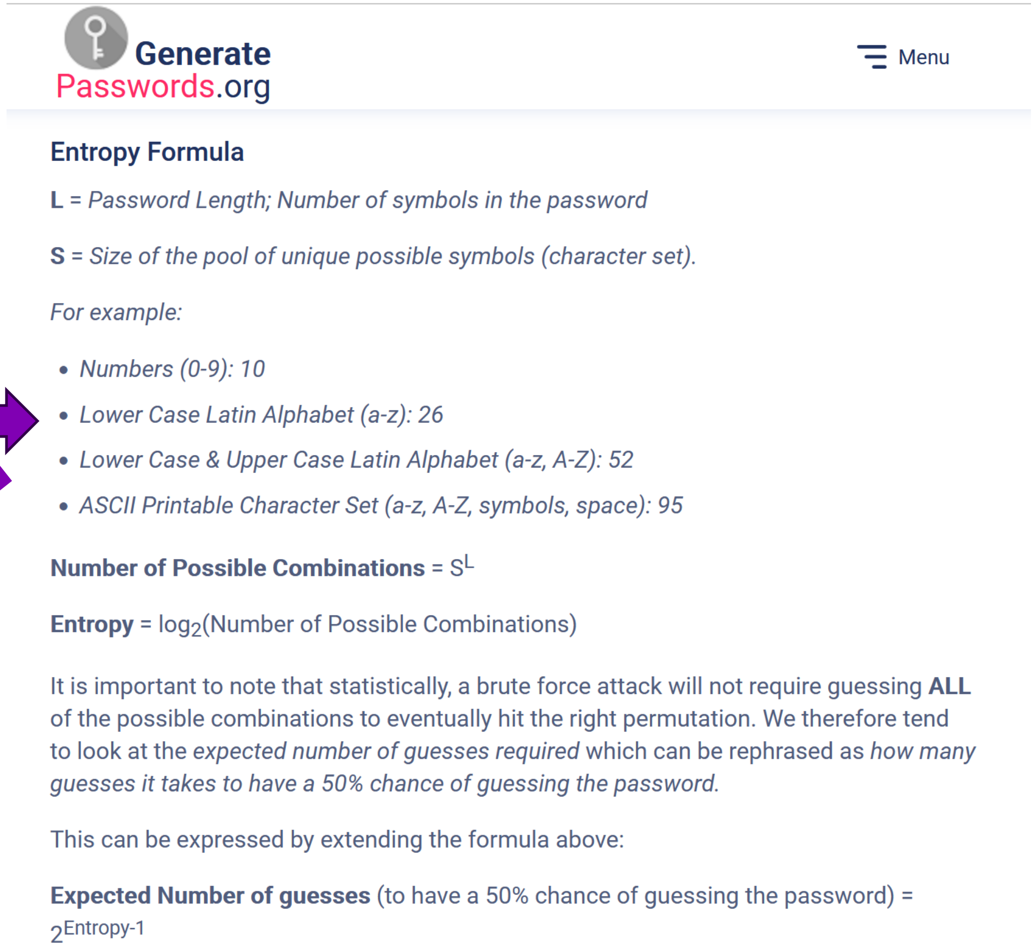
Our second contribution is a novel privacy-preserving approach to collecting a password distribution for statistical analysis (Section IV). By hashing each password at the time of collection with a secret key that is destroyed prior to our analysis, we preserve the password histogram exactly with no risk to user privacy.

Even with millions of passwords, sample size has surprisingly large effects on our calculations due to the large number of very infrequent passwords. Our third contribution (Section V) is to adapt techniques from computational linguistics to approximate guessing metrics using a random sample. Fortunately, the most important metrics are also the best-approximated by sampled data. We parametrically extend our approximation range by fitting a generalized inverse Gaussian-Poisson (Sichel) distribution to our data.

Our final contribution is to apply our research to a massive

Entropy vs frequencies





Entropy Formula

L = *Password Length; Number of symbols in the password*

S = *Size of the pool of unique possible symbols (character set).*

For example:

- *Numbers (0-9): 10*
- *Lower Case Latin Alphabet (a-z): 26*
- *Lower Case & Upper Case Latin Alphabet (a-z, A-Z): 52*
- *ASCII Printable Character Set (a-z, A-Z, symbols, space): 95*

Number of Possible Combinations = S^L

Entropy = $\log_2$(Number of Possible Combinations)

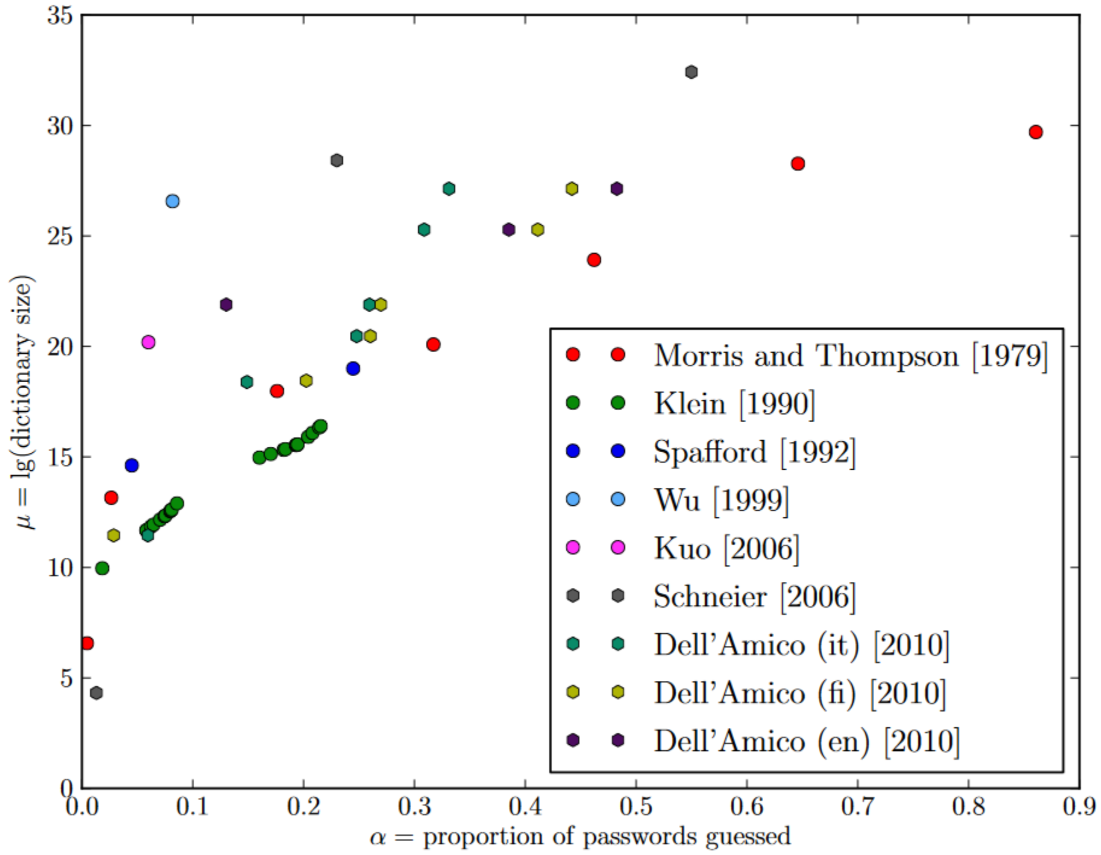
It is important to note that statistically, a brute force attack will not require guessing **ALL** of the possible combinations to eventually hit the right permutation. We therefore tend to look at the *expected number of guesses required* which can be rephrased as *how many guesses it takes to have a 50% chance of guessing the password.*

This can be expressed by extending the formula above:

Expected Number of guesses (to have a 50% chance of guessing the password) = $2^{\text{Entropy-1}}$

Dictionaries

- Lists of common passwords
- Lists of commonly used words
- Mangeling strategies: common adjustments to dictionary words
  - Password -> P@$$w0rd
  - o (char) -> 0 (num)
  - s -> $
- Theoretically dependent on user characteristics like language

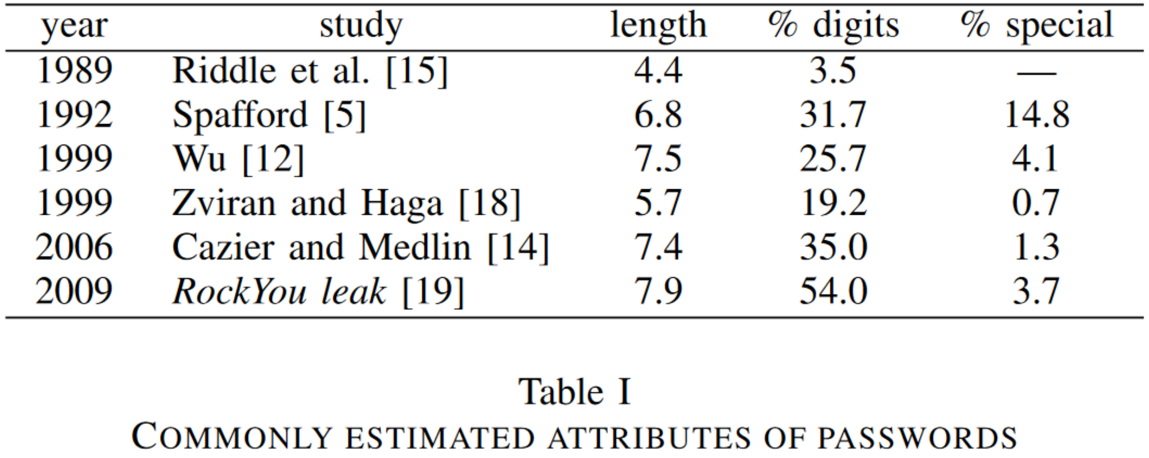


(a) Historical cracking efficiency, raw dictionary size

Joseph Bonneau. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In Proceedings of IEEE SP 2012.

How do we assess password strength?

- Length?
- Character set size?
- Character use frequency?
- Do we assess each password, or do we assess a whole password corpus?



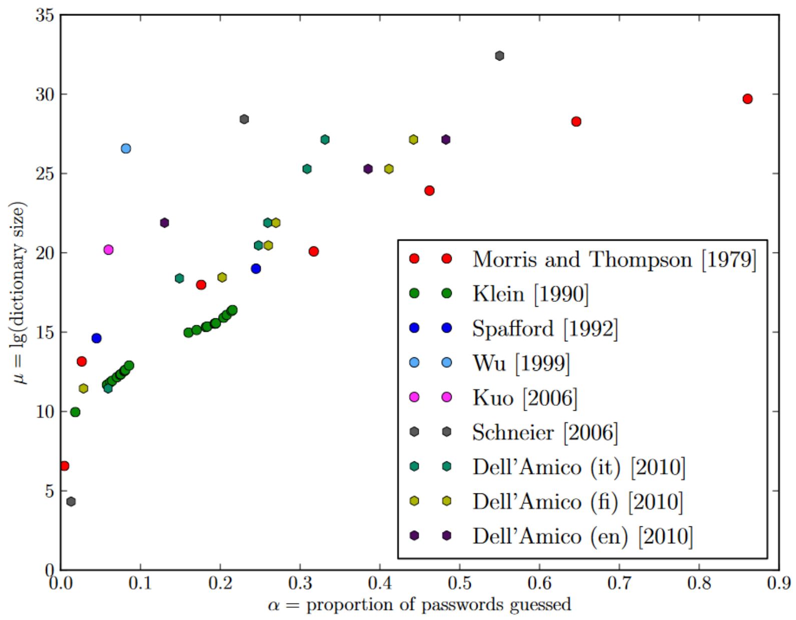
year	study	length	% digits	% special
1989	Riddle et al. [15]	4.4	3.5	—
1992	Spafford [5]	6.8	31.7	14.8
1999	Wu [12]	7.5	25.7	4.1
1999	Zviran and Haga [18]	5.7	19.2	0.7
2006	Cazier and Medlin [14]	7.4	35.0	1.3
2009	*RockYou leak* [19]	7.9	54.0	3.7

Table I
COMMONLY ESTIMATED ATTRIBUTES OF PASSWORDS

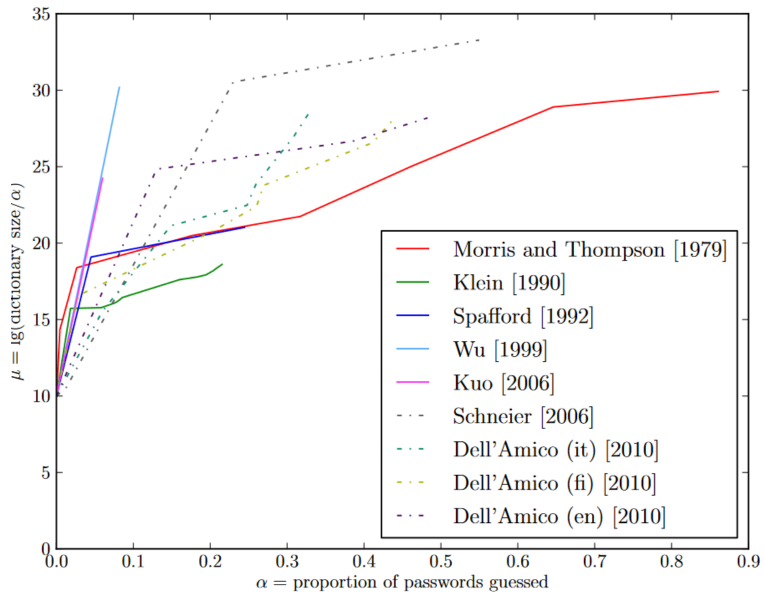
Joseph Bonneau. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In Proceedings of IEEE SP 2012.

Problems with prior approaches

- Comparability – being able to see how two scientific findings align
- Repeatability – can two researchers separately produce the same numbers
- Evaluator dependency – choices in how evaluation is done can have big impacts on numbers
- Unsoundness – they might not have fully thought through the implications of the math



(a) Historical cracking efficiency, raw dictionary size



(b) Historical cracking efficiency, equivalent dictionary size

Joseph Bonneau. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In Proceedings of IEEE SP 2012.

Entropy vs other measures

- Shannon entropy – originally intended to measure signal/noise
- Hartley entropy – how big is the distribution
- Min-entropy – what is the probability of guessing the most common password

- Guesswork: expected number of guesses to find the password
  - Sequential guessing?
  - Probabilistic guessing?
    - Where do the probabilities come from?

Joseph Bonneau. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In Proceedings of IEEE SP 2012.

Estimated guessing curves

- How well can the attacker guess based on knowing different % of the true corpus.

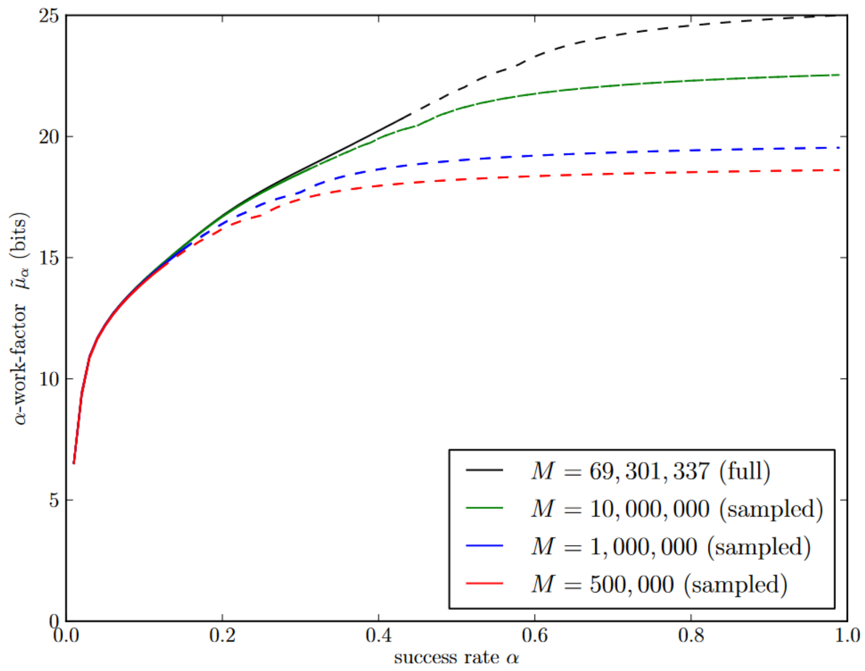
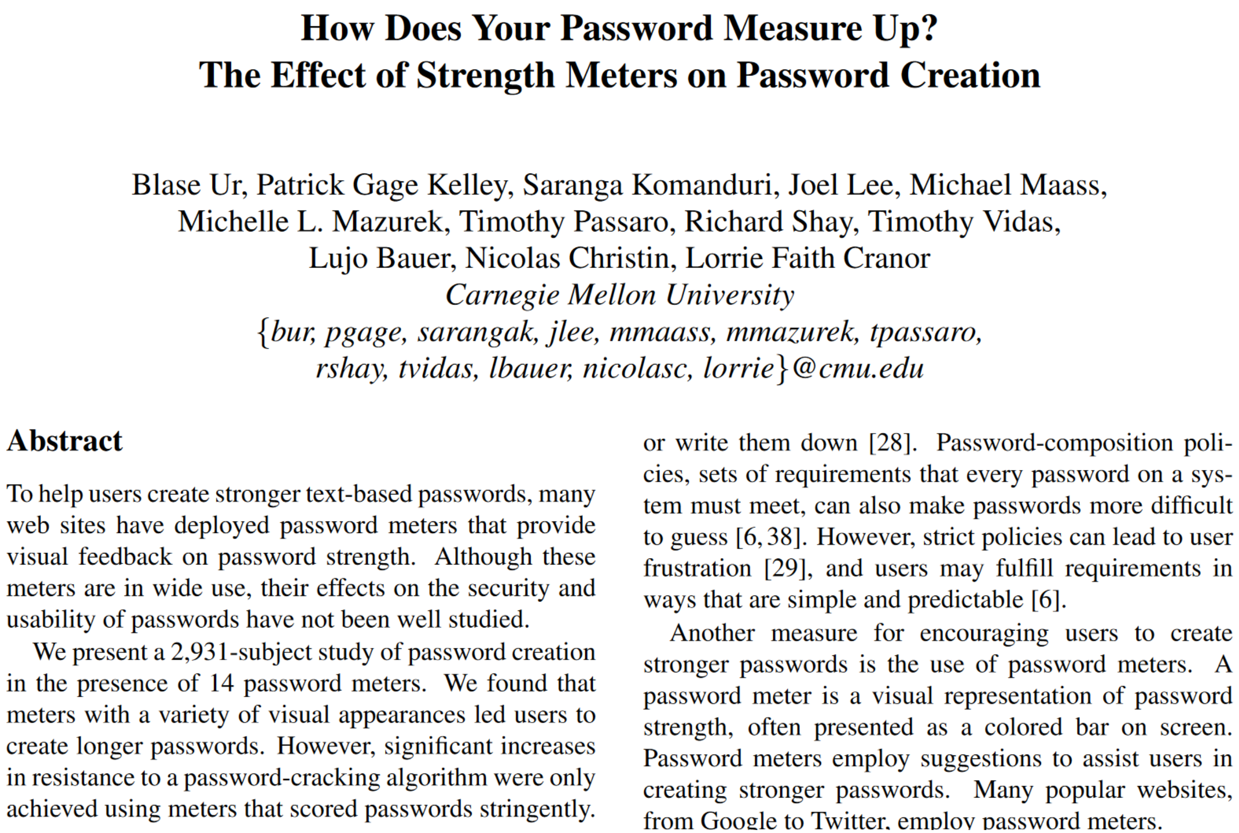


Figure 4. Estimated guessing curves with reduced sample size M. Subsamples were computed randomly without replacement, to simulate having stopped the collection experiment earlier. After the maximum confidence point α_6; there are two (almost indistinguishable) dashed plots representing the 1st and 99th percentiles from 1,000 random samples.

PASSWORD METERS

Password Meters

- Graphical indicators of password strength
- Intended to help people pick good passwords with high entropy
- What type of meter works the best?



How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation

Blase Ur, Patrick Gage Kelley, Saranga Komanduri, Joel Lee, Michael Maass,
Michelle L. Mazurek, Timothy Passaro, Richard Shay, Timothy Vidas,
Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor
Carnegie Mellon University
*{bur, pgage, sarangak, jlee, mmaass, mmazurek, tpassaro,
rshay, tvidas, lbauer, nicolasc, lorrie}@cmu.edu*

Abstract

To help users create stronger text-based passwords, many web sites have deployed password meters that provide visual feedback on password strength. Although these meters are in wide use, their effects on the security and usability of passwords have not been well studied.

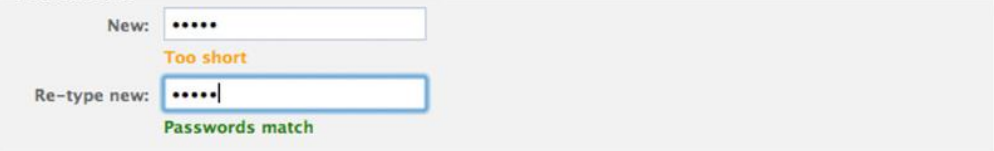
We present a 2,931-subject study of password creation in the presence of 14 password meters. We found that meters with a variety of visual appearances led users to create longer passwords. However, significant increases in resistance to a password-cracking algorithm were only achieved using meters that scored passwords stringently.

or write them down [28]. Password-composition policies, sets of requirements that every password on a system must meet, can also make passwords more difficult to guess [6, 38]. However, strict policies can lead to user frustration [29], and users may fulfill requirements in ways that are simple and predictable [6].

Another measure for encouraging users to create stronger passwords is the use of password meters. A password meter is a visual representation of password strength, often presented as a colored bar on screen. Password meters employ suggestions to assist users in creating stronger passwords. Many popular websites, from Google to Twitter, employ password meters.

Just colored words

Facebook



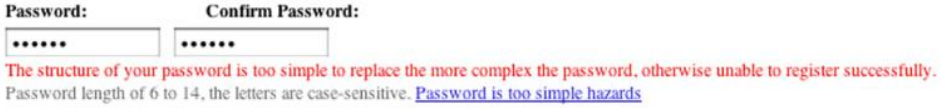
New: •••••

Too short

Re-type new: •••••

Passwords match

Baidu



Password: ••••••

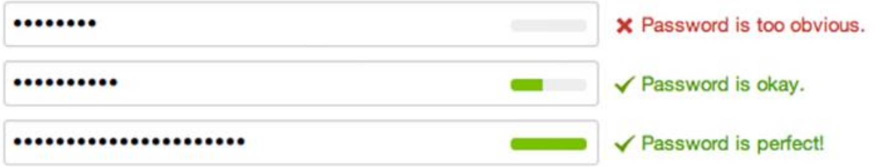
Confirm Password: ••••••

The structure of your password is too simple to replace the more complex the password, otherwise unable to register successfully.

Password length of 6 to 14, the letters are case-sensitive. Password is too simple hazards

Green bars / Checkmark-x

Twitter



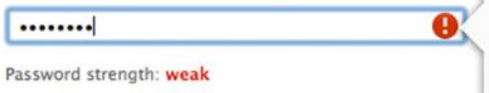
•••••••• ✗ Password is too obvious.

•••••••••• ✓ Password is okay.

••••••••••••••••••••• ✓ Password is perfect!

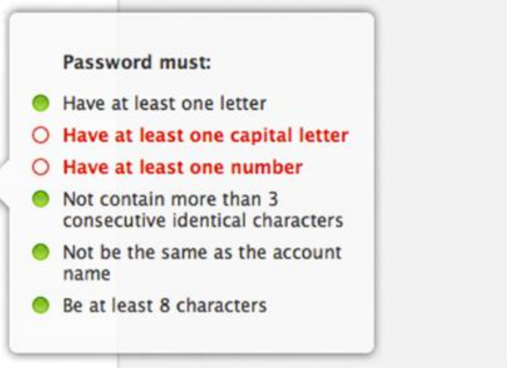
Checklists

Apple



••••••••

Password strength: weak

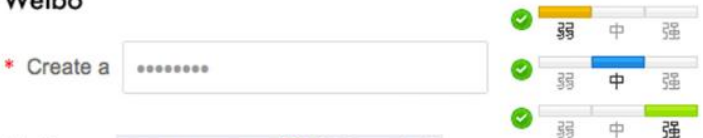


Password must:

- ● Have at least one letter
- ○ Have at least one capital letter
- ○ Have at least one number
- ● Not contain more than 3 consecutive identical characters
- ● Not be the same as the account name
- ● Be at least 8 characters

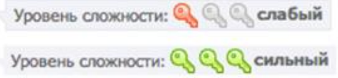
Segmented bars

Weibo



* Create a ••••••••

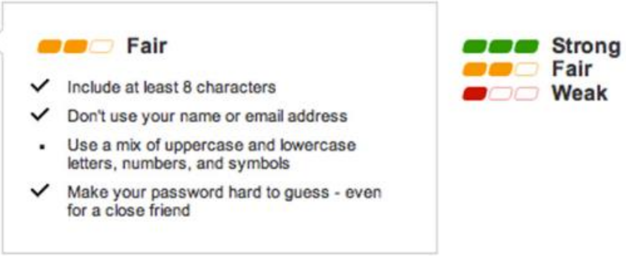
Mail.ru



Уровень сложности: слабый

Уровень сложности: сильный

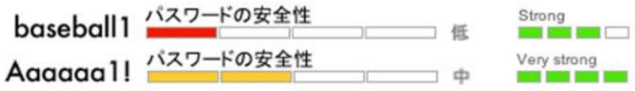
Paypal



Fair

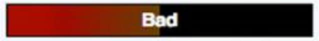
- ✓ Include at least 8 characters
- ✓ Don't use your name or email address
- • Use a mix of uppercase and lowercase letters, numbers, and symbols
- ✓ Make your password hard to guess - even for a close friend

Yahoo.jp and Yahoo

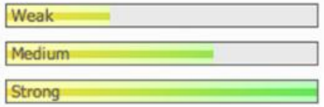


Gradient bars

Wordpress.com

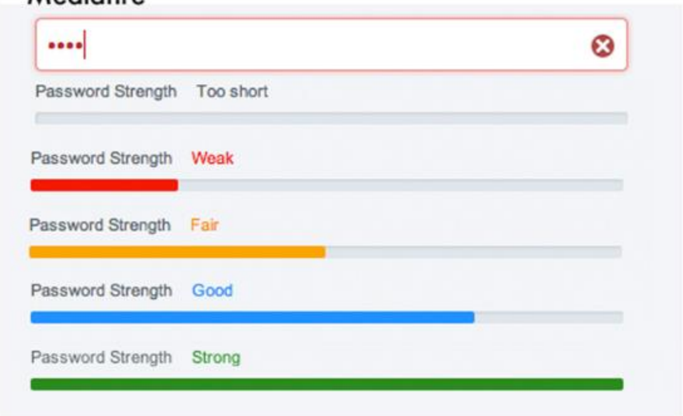


Live.com



Color changing bars

Mediafire



••••

Password Strength Too short

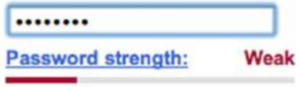
Password Strength Weak

Password Strength Fair

Password Strength Good

Password Strength Strong

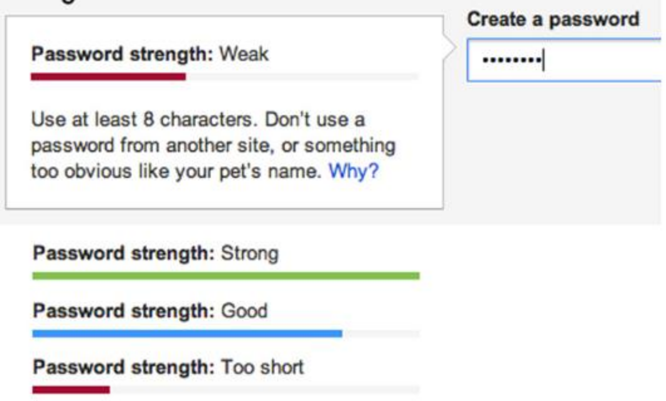
Blogger



••••••••

Password strength: Weak

Google



Password strength: Weak

Use at least 8 characters. Don't use a password from another site, or something too obvious like your pet's name. Why?

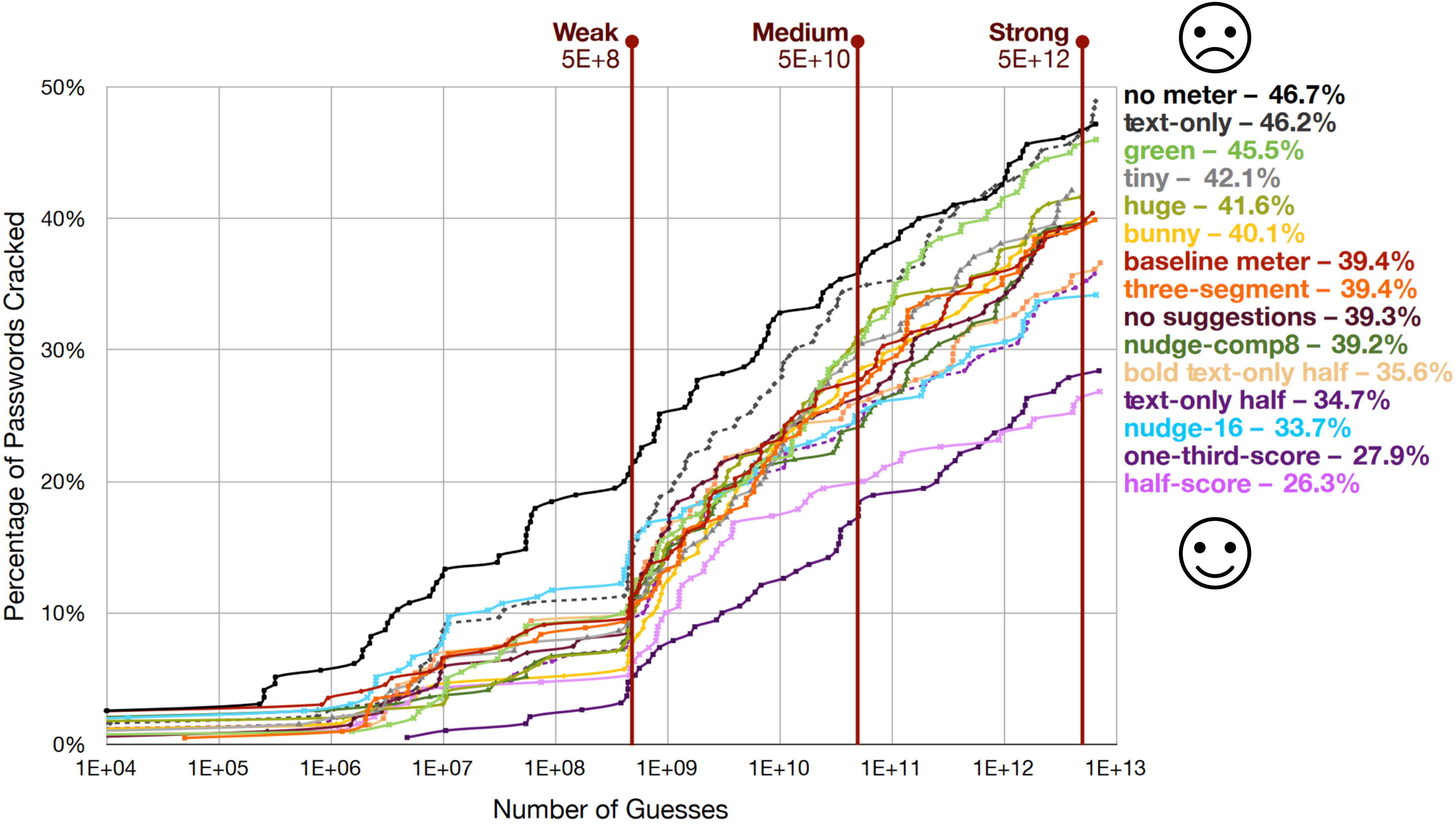
Create a password

••••••••

Password strength: Strong

Password strength: Good

Password strength: Too short


Weak
5E+8
Medium
5E+10
Strong
5E+12
☹
no meter – 46.7%
text-only – 46.2%
green – 45.5%
tiny – 42.1%
huge – 41.6%
bunny – 40.1%
baseline meter – 39.4%
three-segment – 39.4%
no suggestions – 39.3%
nudge-comp8 – 39.2%
bold text-only half – 35.6%
text-only half – 34.7%
nudge-16 – 33.7%
one-third-score – 27.9%
half-score – 26.3%
☺
Percentage of Passwords Cracked
50%
40%
30%
20%
10%
0%
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
1E+10
1E+11
1E+12
1E+13
Number of Guesses