Password market report

Date
25. 1. 2024

Author
Passwd s.r.o.

Version
1.0

Introduction **3**
Context **3**
What is hashing or hash function? 4
Dictionary attacks 4
Disclaimer **5**
Data **5**
Methodology **5**
Results **7**
Names **7**
Cities **7**
Countries **8**
Colours **9**
Companies **9**
Sports **10**
Technical **10**
Conclusion **12**
Contact **13**

Introduction

At Passwd, we deal with passwords daily. We use (pseudo)random passwords for our services, but occasionally, we must generate credentials (usernames and passwords) for purposes like UI mockups, demos, or testing data. Over time, curiosity led us to question: What are the most commonly used passwords? We expected words like "password," swear words, birthdays, etc., to be prevalent.

As our interest grew we decided to analyze password leaks and find out for ourselves. And since some of the stuff seemed hilarious, we decided to share.

Context

One might assume that no service in 2023 stores passwords in plain text. Unfortunately, this isn't the case. Many services still store plain text passwords, and some even email them to users who forget them.

As this report serves educational purposes, no service should store raw passwords. The industry standard is to keep password hashes, allowing services to hash and compare passwords rather than storing and comparing raw passwords.

However, hashed passwords are not immune to attacks; they remain vulnerable to brute force and dictionary attacks, which can be mitigated by using salt.

Salt is a random string hashed together with the password. Since it’s random and differs for each password, it prevents using a precomputed combination of passwords and hashes (rainbow tables). This means that even if two users have the same password, they would have different hash values.

What is hashing or hash function?

A hash function is a one-way function that converts a string of any length into a fixed-length string. One-way function means it's easy to compute a hash from the original string, but impossible to compute the original string from its hash.

MD5 hash of a string password is 5f4dcc3b5aa765d61d8327deb882cf99, while

MD5 hash of a string password1 is 7c6a180b36896a0a8c02787eeafb0e4c.

This also illustrates why adding salt is so effective.

Dictionary attacks

A dictionary attack is a technique of breaking a victim's password by entering a list of possible words (a dictionary) and trying to guess the password.

This attack is slightly more sophisticated than a brute-force attack. Instead of entering random sequences, the attacker starts with the most likely combinations.

Disclaimer

Passwd is a team password manager that runs on Google Cloud. Each customer has their instance on Google Cloud, and we don't have access to the instance or stored passwords.

The reader can check Passwd architecture in our security whitepaper.

Data

We won’t disclose the source of the data, yet we can say it comes from multiple breaches in the past year. In total, the data set contains almost 30M entries.

It's essential to note that this dataset isn't representative of the entire market. There’s no control over the samples, which creates a bias (e.g. if half of the dataset comes from the British market, London may be a much more used token than it would have been in an unbiased dataset).

Methodology

Using the techniques described above, the raw data was transformed into plain text passwords where possible.

For the purposes of this report, we tokenized the data. This means we split the passwords into actual words (e.g. “ILoveThomas” resulted in 3 tokens “i”, “love”, and “thomas”).

From this list, we removed common stop words (e.g. “the”, “a”) and performed stemming using a Suffix-stripping algorithm (e.g. “running” and “runs” were replaced with “run”).

We also tried to preserve relations between tokens, especially if they change meaning by themselves (e.g. “LosAngeles” remains “losangeles”, not “los” and “angeles”).

Finally, we counted the occurrences of each token.

Results

As per the methodology, the results aren't necessarily real passwords but tokens used in passwords. Nevertheless, many tokens are actual passwords. Notably, the most used token in passwords was... "*password*."

Names

The most used tokens were names, with all the top 10 names within the first hundred tokens. This wasn't true for any other category.

Top 10 names used in passwords:

1. thomas
2. justin
3. michael
4. martin
5. jennifer
6. robert
7. jessica
8. david
9. richard
10. michelle

Cities

Predictably, big cities like New York, London, or Los Angeles were among the top 10. Unexpectedly, Liverpool and Helena also ranked high, possibly due to their popularity as a football club and a common female name, respectively.

Top 10 cities used in passwords:

1. phoenix
2. liverpool
3. amsterdam
4. newyork
5. london
6. austin
7. losangeles
8. helena
9. chicago
10. berlin

Countries

America (USA), Pakistan or Mexico being among the Top 10 wasn't surprising. We wouldn't be surprised by almost any big country along the Mediterranean. However, Portugal coming in second place was a surprise.

1. canada
2. portugal
3. america
4. pakistan
5. ireland
6. thailand
7. mexico
8. france
9. scotland
10. russia

Colours

1. gold
2. silver
3. black
4. yellow
5. purple
6. orange
7. green
8. blue
9. white
10. brown

Companies

While discussing this category, we agreed that no big company would surprise us. We were definitely expecting luxury brands like Porsche, Ferrari or Mercedes to be present. Yet Yamaha topping the list exceeded expectations.

1. yamaha
2. porsche
3. ferrari
4. samsung
5. cocacola
6. adidas
7. mercedes
8. disney
9. toyota
10. monster

Sports

Among sports, we pretty much expected football to top the chart. Especially when it encapsulates two sports - American and European football. To support the popularity, the word soccer was also the third most popular.

1. football
2. baseball
3. soccer
4. tennis
5. basketball
6. hockey
7. swimming
8. cricket
9. golf
10. volleyball

Technical

The word password was not only top in our technical category but also by far the most used token in passwords. And the Top 10 list contains another 3 variations.

1. password
2. passw0rd
3. master
4. qwerty
5. internet
6. password1
7. system
8. qwerty123
9. admin

10. p@ssw0rd

Conclusion

In conclusion, our analysis of password data from various breaches over the past year has provided valuable insights into password trends and user behavior. Despite advancements in security practices, it is disconcerting to note that some services still store passwords in plain text, posing significant risks to user accounts.

The prevalence of certain tokens in passwords, such as names, cities, and technical terms, underscores the importance of creating strong and unique passwords. The overreliance on easily guessable passwords like "password" or variations thereof remains a pervasive issue, leaving accounts vulnerable to dictionary attacks.

While our dataset may not represent the entire market and exhibits biases, the observed trends highlight areas where users and service providers can enhance their security practices. As a team password manager, Passwd remains committed to providing a secure environment for our teams, utilizing Google Cloud infrastructure and ensuring the confidentiality of stored passwords.

Contact

Marek Elznic

Product Owner

marek.elznic@passwd.team

Passwd s.r.o.

Evropska 11
160 00, Praha 6