Is MD5 Easy to Crack? (and how long does it really takes)

infosecscout.com/is-md5-easy-to-crack

Patrick Fromaget

Many legends are discussed about the MD5 algorithm, but it’s time to break the myths and talk about real facts. Is the MD5 algorithm still safe to store passwords? That’s what we’ll see in this article.

The MD5 algorithm is no longer considered safe to store passwords, as it’s coming more and more easy to crack them. As an example, it’s possible to brute force an 8-characters password in a few minutes.

The short answer is that MD5 is becoming easier and easier to crack. I’ll explain why in this article, how the hackers are doing this, and what you can do against it.

Can you crack MD5?

The MD5 algorithm is a one-way hash function, it’s not reversible. So, there is no direct method to decrypt a hash and get back the original password.

But it doesn’t mean that MD5 is safe, we’ll see why.

How safe is MD5 hash?

In 2005, Bruce Schneier, an American cryptographer, declared that the MD5 algorithm was broken. The main reason was that collisions has been detected in the hash function.

That’s still an issue today, but not the only one. With the high improvements in the computer technology, there is another important one: speed. MD5 has been designed to be particularly fast to use, and that’s made its success.

You can convert thousands of passwords to MD5 hashs in a few seconds, so the system can have a great response time. It was an important factor at the beginning of the Internet. But now, this strengh plays against it.

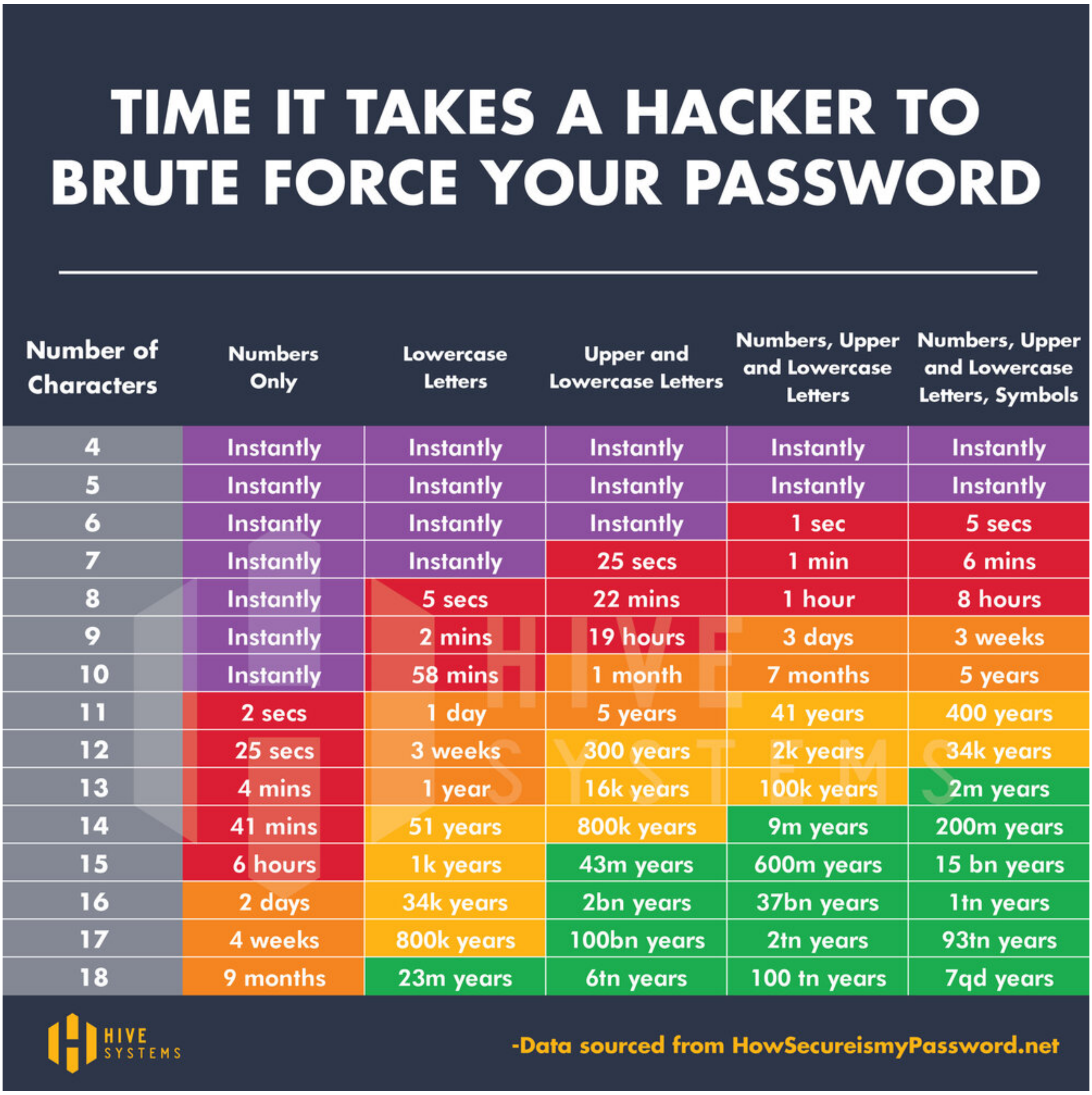
MD5 is too exposed to brute force attacks. With current GPU, computers can quickly generate billions of MD5 hash from random words until they find the corresponding one. That’s the principle of brute force attacks (as explained in this article).

How long does it take to crack MD5 passwords?

With the current technology, it takes a computer 8 hours to crack a complex 8-characters password (numbers, upper and lowercase letters, symbols).

So, that’s pretty fast. The computer will try every combination until finding the correct one. And it doesn’t really take much longer if there is a huge list of passwords to crack. Each combination is converted to the corresponding MD5 hash and compared to the whole list in no time.

The best way to protect you against this is to use long passwords. Complexity had a bit of time, but password length is the main factor. To be safe, never use a password shorter than 15 characters. A passphrase like “ILoveMD5Online.org” will be much complicated to crack than “56%AfZ@”, and also way easier to remember.



TIME IT TAKES A HACKER TO BRUTE FORCE YOUR PASSWORD

Number of Characters	Numbers Only	Lowercase Letters	Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters, Symbols
4	Instantly	Instantly	Instantly	Instantly	Instantly
5	Instantly	Instantly	Instantly	Instantly	Instantly
6	Instantly	Instantly	Instantly	1 sec	5 secs
7	Instantly	Instantly	25 secs	1 min	6 mins
8	Instantly	5 secs	22 mins	1 hour	8 hours
9	Instantly	2 mins	19 hours	3 days	3 weeks
10	Instantly	58 mins	1 month	7 months	5 years
11	2 secs	1 day	5 years	41 years	400 years
12	25 secs	3 weeks	300 years	2k years	34k years
13	4 mins	1 year	16k years	100k years	2m years
14	41 mins	51 years	800k years	9m years	200m years
15	6 hours	1k years	43m years	600m years	15 bn years
16	2 days	34k years	2bn years	37bn years	1tn years
17	4 weeks	800k years	100bn years	2tn years	93tn years
18	9 months	23m years	6tn years	100 tn years	7qd years

HIVE SYSTEMS

-Data sourced from HowSecureismyPassword.net

This infographic From Hive Systems shows how much time the computer needs to crack your password, depending on its length and complexity. Remember that technology improve fast and hackers can use supercomputers if the attack is worth it (with companies password for example).

To know if your password is safe enough, you can use this tool that has been used to generate this table. It will show you how much time will be required to crack your password.

How to use safer passwords?

```
As a whole, use long passwords with character variety (always one symbol at
least), don't user real words. Use a different password on each website or
application. A password manager can help you to keep them safely, don't store them
anywhere else.
```

Anyway, that’s not the goal of this article, but it was an important reminder. MD5 is not safe, but using short passwords won’t be safe, whatever the algorithm used by the developer.

How MD5 decryption works?

As there is no reverse function to decrypt a MD5 hash, MD5 decryption doesn’t exist. But techniques like brute-force or dictionary attacks are really efficient to have the same result. The idea is to hash a huge number of words into MD5, and try to find a match.

I have an entire article on this topic that you can read for more details, but I’ll give you a summary here.

Brute-force attacks

The first way to decrypt an MD5 hash is to brute force it.
A brute force attacks goal is to try many words, convert them into MD5, and check if the MD5 hash is corresponding to what we are looking for.

In general, you’ll start with the shortest password (example: “a”), convert it to a MD5 and see if there is a match. If not, you continue with the next possibility. Computers can do this for days until they find a match, it doesn’t require any human interaction.

The hacker will just check from time to time if there are new results, and continue from there, if he has find something interesting.

If you want to try this on your computer, you can read this article, where I give you a tool you can use. You’ll better understand how powerful these techniques is by seeing it running on your PC.

MD5 databases

An MD5 database use the same principle, but will store each hash in a file. This way, it’s possible to quickly check matches against a file containing lots of MD5 hashes.

If a hacker is doing this regularly, it could increase the speed by having a database containing the most common passwords, for example.

What are alternatives to MD5?

MD5 is considered broken, but it’s not the only one, SHA1 has the same problems. Overall, the current recommendation is to use stronger algorithms like SHA-256, but other options are possible like bcrypt.

If you are a bit lost to choose a safer solution, I have a few articles that should help you to do just this:

- MD5 vs SHA256: Which is Better? (Speed, Safety, …)
- What's the difference Between MD5 and SHA1?

Also, I highly recommend using salt to store passwords, whatever your choice for the hashing algorithm, it will highly decrease the chance of your passwords to be cracked. With a good salt, databases attack won't be as efficient, and brute-force attacks will take much more time.

Don't know what is salt in cryptography? Read my article here to quickly get the main idea. After that, you just need to find the best way to implement it depending on the technology you use.

Whenever you're ready for more security, here are things you should think about: