How do websites protect your passwords?

How do websites store your passwords? How do they keep your logins secure? And what is the most secure method websites can use to track your passwords?

It's normal to hear about a data breach these days. A breach can happen with a popular service like Gmail or a piece of software that most of us have forgotten, like MySpace.

One of the worst things a hacker can figure out is your password. This is especially true if you go against the standard advice and use the same login on multiple platforms. But password protection is not only your responsibility.

So how do websites store your passwords? How do they keep your logins secure? And what is the most secure method websites can use to track your passwords?

Worst case scenario: Passwords are stored in plain text

Consider this situation: A large website has been hacked. Cybercriminals have circumvented any basic security measures the website takes and can take advantage of a flaw in the site's architecture. You are the customer. That site already stores your details. The page ensures that your password is secure. But what if that web platform stores your passwords in plain text.

Passwords in plain text are like a lucrative bait. They don't use algorithms to be unreadable. Hackers can read passwords as simple as the way you are reading this article.

It doesn't matter how complicated your password is: A plain text database is a list of all users' passwords, clearly written, including any numbers and additional characters you use.

And even if hackers don't crack the site, do you really want a certain site admin to be able to see your secret login details?

You might think this is a very rare problem, but an estimated 30% of e-commerce websites use this method to "secure" customer data!

An easy way to find out if a site stores passwords in plain text is that right after you sign up, you receive an email from the site listing your login details. In that case, you may want to change any other websites that use the same password and contact the company to warn them that their security is too poor. It is not possible to be 100% certain of course, but this is a pretty clear indication and the website really shouldn't send such things in emails.

Coding: Sounds good, but isn't perfect

Many websites turn to encryption to protect users' passwords. The encryption process scrambles your information, making it unreadable until two keys - one held by you (that's your login details) and the other from the company in question - appear together.

You've also used encryption in many other places. Face ID on the iPhone is a form of encryption. So is the passcode. The Internet runs on encryption: The HTTPS you can see in the URL means that the website you are visiting uses SSL or TLS protocols to verify the connection and aggregate data. But in reality, the encryption is not perfect.

Encryption can make you feel secure. But if a site is protecting your password using their own, a hacker could steal the site's password, then find your password and decrypt it. It won't take much for a hacker to figure out your password; that's why major databases are always a big target.

If the site's key (password) is stored on the same server as your password, your password may also be in plain text.

Hash: Surprisingly simple (but not always effective)

Password hashing sounds like jargon, but it's simply a more secure form of encryption.

Instead of storing the password in plain text, the website runs the password through a hash function, like MD5, Secure Hashing Algorithm (SHA)-1 or SHA-256, which turns the password into a completely different set of digits. This can be numbers, letters or any other character.

Your password could be IH3artMU0. It can turn into 7dVq$@ihT and if a hacker breaks into the database, that's all he can see. Hackers cannot re-decrypt the original password.

Unfortunately, things are not as secure as you think. This is better than plain text but still not a problem for cybercriminals.

It is important that a particular password generates a particular hash. There's a good reason for that: Every time you log in with the password IH3artMU0, it automatically passes that hash, and the site allows you to access it if that hash and the function are in the site's database web match.

In response, hackers have developed rainbow tables, similar to cheat sheets. They are lists of hashes, already used by others as passwords, that a sophisticated system can quickly run through, like a Brute Force attack.

If you have chosen a really bad password, that password will be high on the rainbow table and can be easily cracked. Complex passwords will take longer.

Today's Best: Salting and Slow Hash

Nothing is invulnerable: Hackers are always actively working to crack any new security system. Currently, there are more powerful techniques implemented by the most secure websites. Those are smart hash functions.

Salted hash is based on cryptographic nonce, a random data set generated for each individual password, which is often very long and complex.

These extra digits are added to the beginning or end of the password (or email-password combination) before it passes through the hash, to resist attempts made with the rainbow table.

Generally it doesn't matter if salts are stored on the same servers as hashes. Cracking a set of passwords can take a lot of time for hackers, even harder if your passwords are complex.

That's why you should always use strong passwords, no matter how confident you are in the security of the site.

Websites also use slow hashing as an additional measure. The most famous hash functions (MD5, SHA-1 and SHA-256) have been around for a while and are widely used because they are relatively easy to implement.

While still applying the salt, slow hashes are even better at resisting any speed-based attacks. By restricting hackers from making significantly fewer attempts per second, they take longer to crack, thus making the attempts less valuable, and also giving the success rate lower.

Cybercriminals must consider whether it's worth attacking slow hashing systems that take time versus "fast fixes". For example, medical institutions often have a lower level of security, so the data obtained there can still be sold for surprising amounts.

If a system is under 'stress', it can slow down even more. Coda Hale, a former Microsoft software developer, compares MD5 to the most notable slow hash function, bcrypt (others include PBKDF-2 and scrypt):

"Instead of cracking passwords every 40 seconds (as with MD5), I would crack them every 12 years or so (when the system uses bcrypt). Your password may not need that kind of security. and you may need a faster comparison algorithm, but bcrypt lets you choose a balance between speed and security".

And because a slow hash can still be done in less than a second, users won't be affected.

You finished reading the article "**How do websites protect your passwords?**" edited by the **TipsMake** team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.