Brute-forcing common password hash functions

Speeding up brute-force attacks by moving hash calculations to GPUs.
Password hashing is a normal precaution against hacking

• Password hashing is nowadays an industry standard and common good practice
  – This does not mean it’s universal. Some firms do not do this.
  – When found out this has caused a lot of bad PR for firms.
  – Still most firms make the mistake of using normal hash algorithms for their passwords instead of proper password hash algorithms.

• Compromising user accounts is still the goal of some hackers
  – Plaintext passwords are usually unavailable even to the system.
  – A compromised server still usually yields a list of hashed passwords along with other user information.

• After an attack the password hashes need to be “cracked” before any gain can be made from the system.
Password verification in hashed systems

- The system does not store the plaintext password on non-transient memory.

- How can the password then be verified?
  - The original password is hashed with a suitable hash algorithm and the corresponding hash-value is stored in a database.
  - Upon login the submitted password is hashed with the same algorithm and the resulting hash is compared with the stored value.
  - If the algorithm is chosen well and is strongly collision resistant, a match of the hashes means that the passwords match.
  - Is not actually true, since the hash-space of an algorithm is not infinite, so eventually a collision is bound to happen, but the likelihood is astronomically small.
Cracking hashes

• Usually the method used in creating the hash is known to an attacker.
  – If access is gained to a server the actual computations used in the program can be observed in the code.
  – Usually it is just mentioned in the server that passwords are hashed using this or that algorithm.
  – The information of the algorithm used is not that vital to the security and even then any security measures taken can be circumvented with a known plaintext attack.

• Three general methods are used for breaking unknown passwords with only a known hash.
  – Brute-force
  – Dictionary attacks
  – Rainbow tables

• All are some kind of trade-offs.
Rainbow-tables: Exchanging memory footprint for speed

• Large pre-computed tables of hashes for most used passwords.

• Is not guaranteed to crack all passwords, but common libraries can contain millions of pre-computed hashes.
  – CryptoHaze, an open source implementation has full tables for MD5 and NTLM up to 8 characters and SHA-1 to 7 characters.
  – These can require terabytes of memory to use, but speed up the cracking by magnitudes.

• Rainbow-table based attacks are easily thwarted by using a “salt” in computing the hash.
  – Salt is any piece of bits that is either appended or XORred to the password before the hash operations.
  – Causes the pre-computed hash values to be incorrect unless the same salt was used in generating the tables.
Dictionary attacks
Trading completeness for speed

• Instead of just brute-forcing a password educated guesses about normal password generation methods are made and used in selecting passwords to use.

• Truly random passwords are not even tried with this method and thus such passwords are safe from attack.

• Stands on the fact that most users do not choose their passwords randomly.
  – One of the most used passwords is still password.
  – In an attack on Gawker Media in 2010 the most used passwords in the material were 123456, password and 12345678. (password123 also made an appearance.)
  – The site was and is mainly targeted toward tech-savvy persons, so the general quality of passwords can be even lower.

• Instead of using a lot of computing power to guess randomly generated strings, a dictionary of common passwords is created with likely methods of customizing them and this is used to generate guesses.

• Works well when the target is random users instead of a single specific user in the database.

• The best passwords remain uncracked but with the same amount of effort that it would take to crack a well selected password, tens or even hundreds of low-security passwords can be compromised.
Brute-force
Completeness disregarding speed

• Systematically going through the available keyspace for the hash algorithm until a matching hash is produced.

• Extremely time intensive, but guaranteed to yield the password given enough time.

• Previously ran on dedicated hardware to gain enough speed to be made feasible.

• With the advent of GPGPU solutions the limit of feasibility has risen and continues to rise as even faster GPUs are released and computers with parallel GPUs become easier and cheaper to get.
Password must be at least X characters long

• Historically minimum password length has been enforced to make brute-forcing infeasible.

• On web services a lower limit of six or eight characters is quite typical.

• Unfortunately modern GPUs can exhaust an eight character keyspace of ascii character set in less than a day.

• With the addition of dictionary attacks, even lengthier passwords are easily cracked in short time-frames unless the password is chosen well.
Cracks in the walls

Some standards have mistakes

- In the present almost no one uses WEP standard for encrypting WiFi traffic, as problems with the standard allow it to cracked in minutes with normal laptops.
- Most users have moved to WPA or WPA2 –encryption that is considered safe.
  - Still some considered setting up and connecting to the access point to be too difficult
- This caused the drafting and addition of WiFi Protected Setup to most routers.
- WPS unfortunately has a vulnerability that requires the attacker to guess an eight characters long numerical string to gain admin access to the router.
- Most routers allow disabling of the WPS and users should do so if they wish to protect their networks.
  - This doesn’t mean that the router is protected though.
  - The standard does not require the router to actually switch off the authentication when the mode is disabled and thus most don’t.
  - The attack only requires sending authentication packets to the router until a correct passcode is guessed and the router gives admin access.
  - Deficiencies in the hash algorithm used to shield the passcode make this attack even more feasible than it should have been.
  - Reaver program developed by Tactical Network Solutions can crack a WPS-enabled network in 4-10 hours while running only on the CPU.
Using GPUs for hash calculations

• Calculating a huge amount of hashes for millions of password guesses is the epitome of SIMD.

• The same operations are always performed in the same order. Just the starting bit string is different.

• Practically all password brute-forcing is done on GPUs.
  – Open source implementations can be found with one or two short searches on Google.

• One example is Bitcoin mining, which is basically done on GPUs. A simple program for performing mining on your own home computer can be found on Bitcoin’s wikipages.
Is it possible to protect yourself from these attacks?

• Short answer yes, long answer yes, for a time.
• Proper password hygiene goes a long way, but does not always help.
  – Microsoft Live truncates all passwords to sixteen characters before hashing, so longer passphrases are useless.
• The most helpful thing would be for all firms to move to proper password hashing functions.
  – Most used today are MD5 and SHA-1
  – These were created for producing secure hashes with low collision frequency, not password storage.
  – They are generally pretty fast.
  – PBKDF2 is a password hash function endorsed by the United States.
    – Based on SHA-1, but works by running SHA-1, adding salt and running SHA-1 again.
    – In normal operation runs a thousand cycles of SHA-1, also a thousand times slower than normal SHA-1.
• Proper password hash functions use secure hash algorithms, but run them many times in sequence.
  – Brute-forcing and even dictionary attacks become infeasible as the time required for a single password grows thousand-fold.
  – Most are adaptive, so that the amount of iterations can be increased later on to account for the increase in available computing power.
  – Can be considered unbreakable. The password is exchanged more often than it would take to crack it.
Thank you and stay safe!

Questions?