Stephen Robinson

CSCI 373.001

Homework 1 - Passwords

February 6, 2019

Task 4 in the first homework involves utilizing the free password cracking tool John the Ripper. While the original requirements for this task were to use John the Ripper on the password file generated using the program written in Task 1 and then evaluate our own password, this was changed to explore the tool on our own passwords using different encryption methods after several students mentioned that the tool was unable to crack the scrypt hashes we developed in Task 1.

I began by using the password ‘password’ with an MD5 hash (with a salt of only 2) and also again with a DEScrypt hash (with a salt of 8). As expected, both of these passwords were quickly cracked as seen in figures 1 and 2 below.

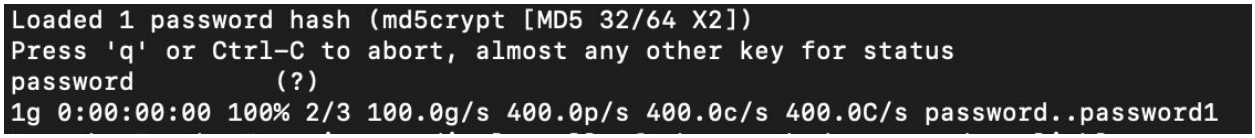
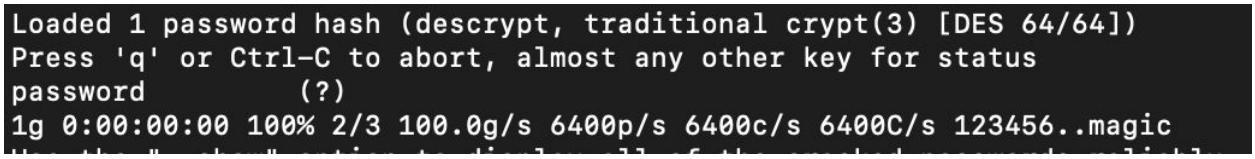



Fig 1 & 2- an MD5 (above) and DEScrypt (below) hash of ‘password’




However, when changing the password to something outside of simple one word phrases, the time to crack jumps astronomically. Even something as simple as the two dictionary words

‘apple’ and ‘sauce’  concatenated and hashed with the same MD5 algorithm as used above, resulted in a runtime outside of my reasonable willingness to wait.

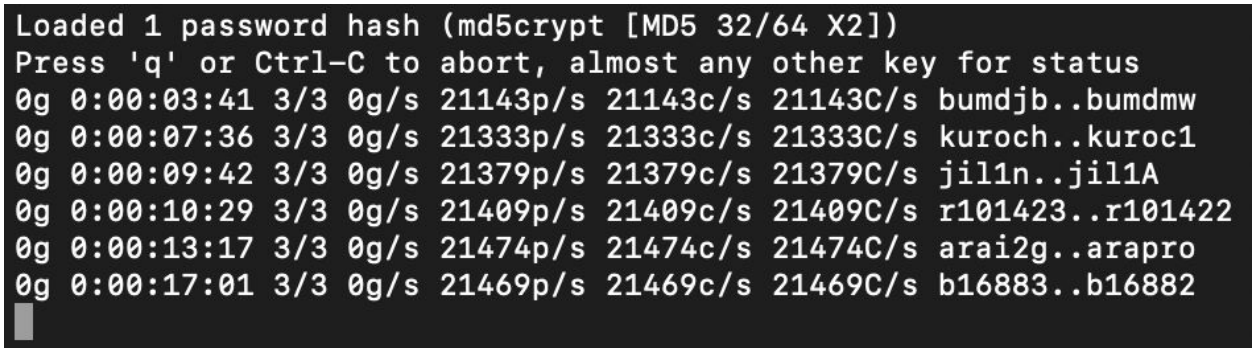



Fig 3 - an MD5 hash of ‘applesauce’ being cracked

Because of the drastic increase in runtime, it is clear that this tool utilizes a list of preferred common passwords before resulting to brute-force attacks. Reading through the documentation for the tool confirms this suspicion as the author of the utility even recommends uploading updated lists of common passwords to increase efficiency. However, the longer runtime of this particular example did uncover that even though John the Ripper had not yet cracked the hash, it was capable of determining exactly which hashing algorithm was used. By simply evaluating the characteristics of my provided hash, the utility was able to drastically decrease the problem-space.

While the utility John the Ripper is certainly more sophisticated than anything I have written to-date, its’ relative inability to quickly crack “simple” passwords consisting of only English language dictionary words proves the vast security benefits of using hashing algorithms to secure information. The resulting turnaround for successfully brute-force cracking uncommon passwords is often too great to be worth the time and computational resources spent even when using a utility as lauded as John the Ripper.