|  |  |  |
| --- | --- | --- |
| Luke Pepin | CSE 4400 - HW 1 | 2/5/2025 |

1. Web Browser Saved Passwords:

Google Chrome saves passwords to your Google Account, meaning they are stored in the cloud. The server does not know the plaintext passwords, as they are encrypted before being sent and decrypted in the browser upon retrieval. Chrome does not provide a master password. As a result, the password system requires passwords to be accessed with the operating system login credentials, which may differ across devices.

On the other hand, Firefox follows a nearly opposite process. Passwords are saved in a logins.json file in the user’s profile directory. A master password is optional. However, much like Chrome, Firefox requires the user to enter their operating system credentials or master password to view the saved passwords. Users have the option to store passwords in the cloud with Firefox Sync, which receives encrypted passwords that are then decrypted by the user's browser when needed.

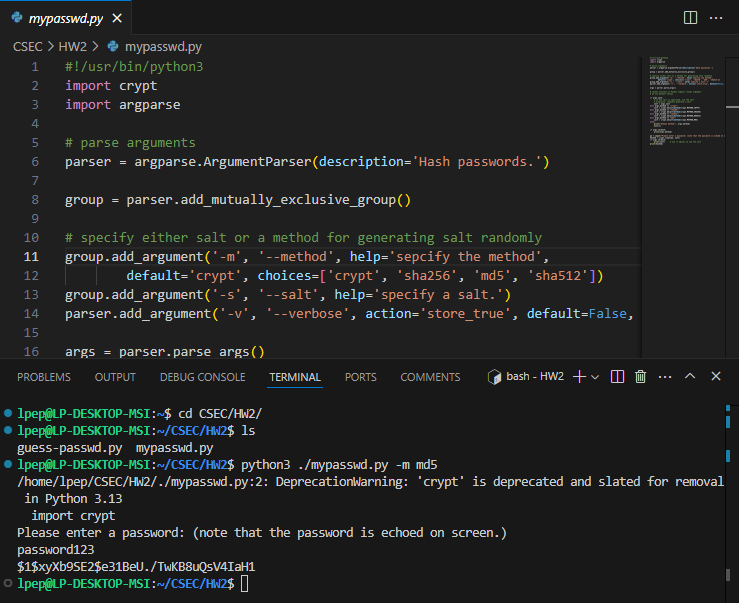
Sources:

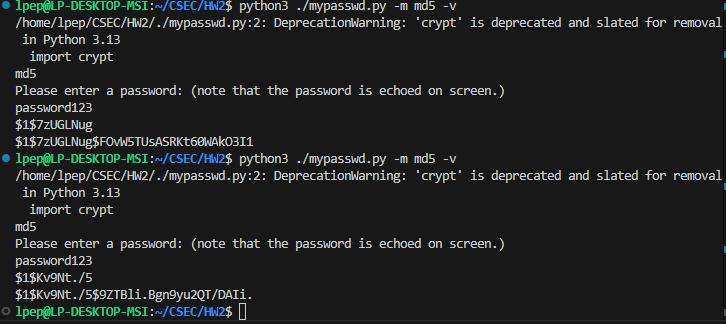
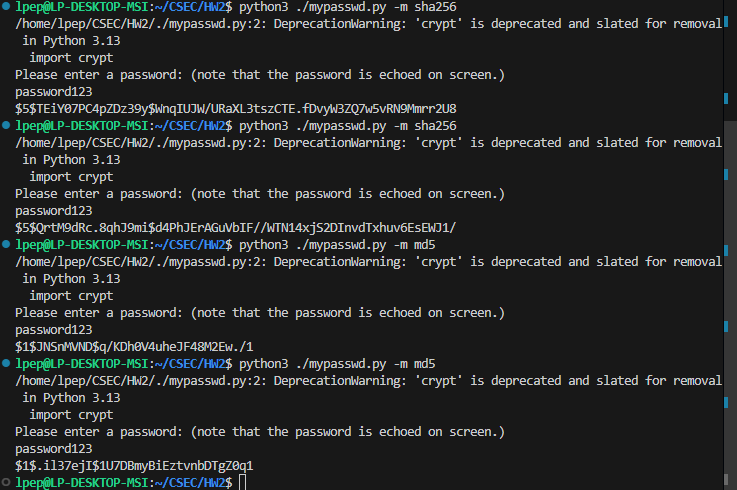
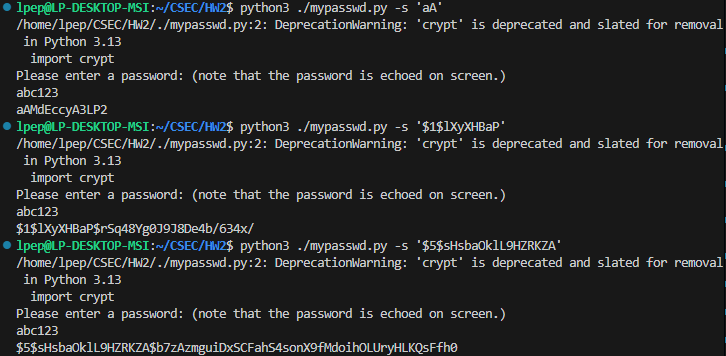
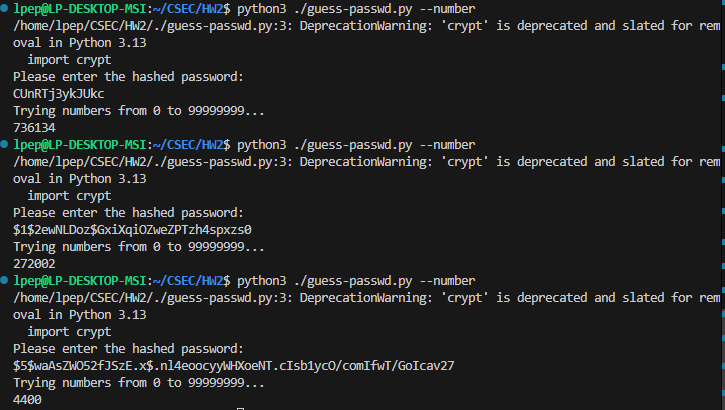
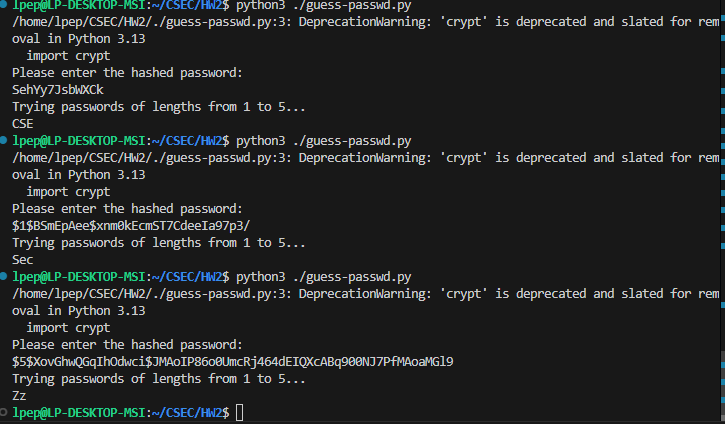
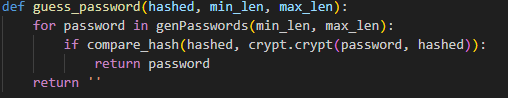
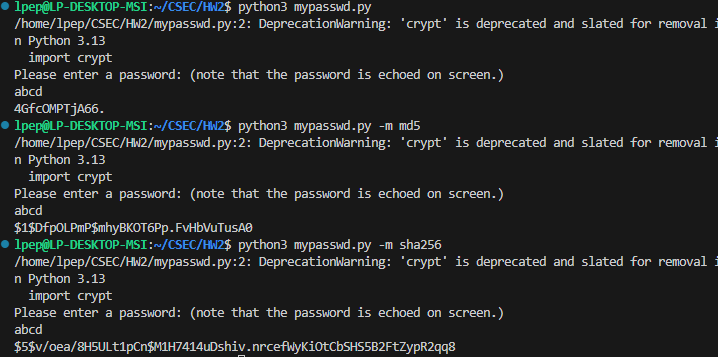
<https://support.google.com/chrome/answer/95606?hl=en&co=GENIE.Platform%3DDesktop>

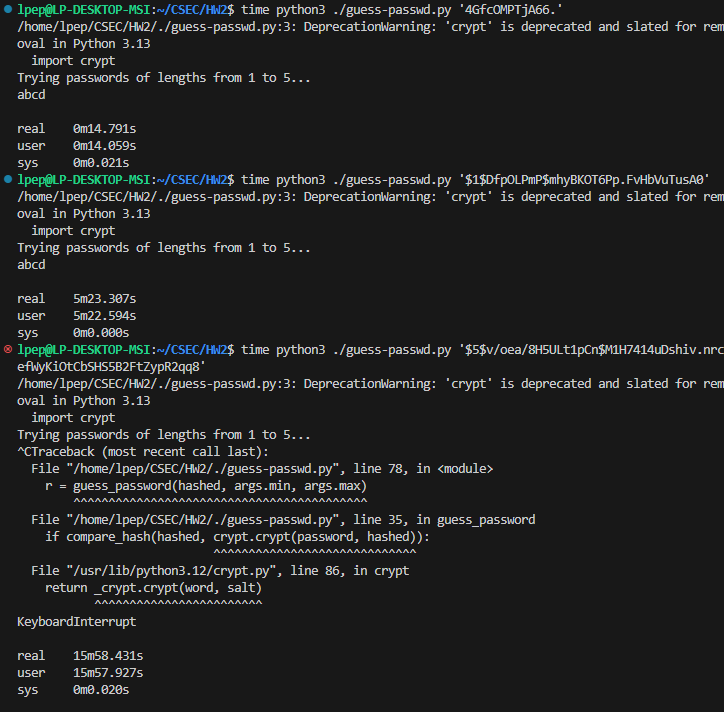
<https://support.mozilla.org/en-US/kb/how-firefox-securely-saves-passwords?form=MG0AV3&form=MG0AV3>

1. Programing Languages Crypto Libraries:

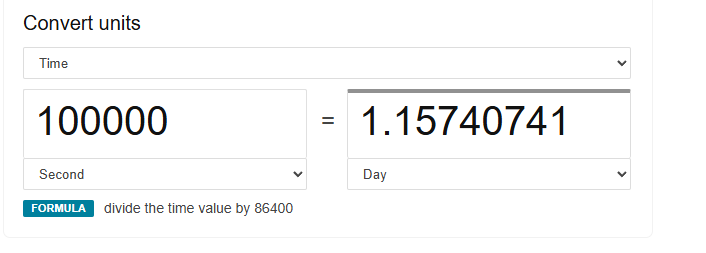
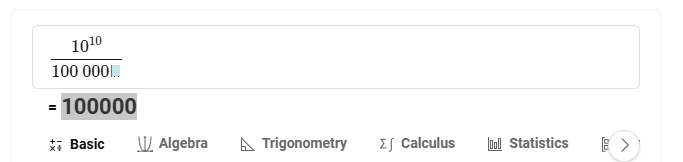
Default Run (md5):



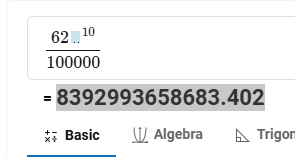
1. Different Methods: In the following problem two questions are asked. The first question “With the same method, do you see the same hash for the same password?”, as you can see in the first screenshot both methods md5 and sha256 are ran twice with the same password each time different hashes are outputted. The second question “Given a hashed password, can you find the salt?”, on the second screenshot the same command is ran twice with the –v option which prints out the salt from the hashed password, both times the salt is the first 11 characters (for md5, different methods have different salt lengths) in a hashed password, as a result it is clear that given a hashed password, if you know the length of the salt, the salt can easy be found. 
2. Salt ‘abc123’: The given password, was salted with the given salts and the last hash value of all three are the same as specified.
3. Guess-password-numbers: Each of the given hashes were run with the command ‘python3 ./guess-passwd.py --number’
4. def guess-password: In screenshot one the revised guess-password method is shown. Afterwards the function was tested with the given hashes and the passwords were guessed. 
5. Speed Check: First for each hashing method the same password was hashed, next each hash was run in guess-password with time to measure the speed to crack each method. The results are as follows: Crypt 14s, md5 5m23s, sha256 (cancelled) 15m 58s. Using sha256 as a baseline, the approximate rounded ratios are: crypt is 68 times faster and md5 is 3 times faster than sha256. 

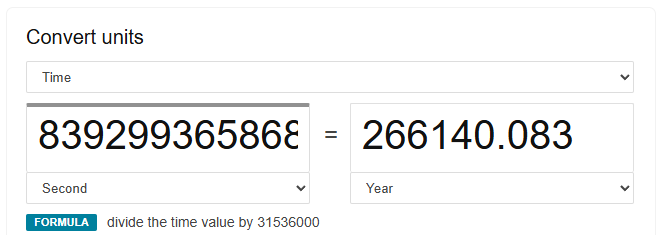


1. Time calculations:
2. Ten decimal digits:



1. Ten decimal digits, uppercase and lowercase:





1. Four English words:

