Elinor Holt

Prof. Goldberg

CS 357

Nov 1, 2024

# Challenge 2: Brute-Force

A. General:
    a. For this challenge, I use the same method as Challenge 1, submitting possible passwords until I receive a response which contains the string “ACCESS GRANTED.”
    b. I used multithreading in this challenge to speed up the program (20 worker threads).
B. Instructions (automated in code):
    a. First I had to determine what all the possible password combinations were:
        i. The password must contain the user’s name. The possible names were given: ['alice', 'bob', 'carol', 'eve']
        ii. The password must contain the user’s birthday in an ISO 8601 date format and we know the user is a millennial and therefore born between 1981 and 1996.
    b. Using this information, I mapped out all the possible combinations of name and ISO birthday format:
        i. name + YYYY-MM-DD
        ii. name + YYYYMMDD
        iii. name + MM-DD

iv. name + MMDD

v. name + YYYY-DDD

vi. name + YYYYDDD

vii. name + YYYY-Www-D

viii. name + YYYYWwwD

ix. YYYY-MM-DD + name

x. YYYYMMDD + name

xi. MM-DD + name

xii. MMDD + name

xiii. YYYY-DDD + name

xiv. YYYYDDD + name

xv. YYYY-Www-D + name

xvi. YYYYWwwD + name

c. Then, starting from the date 1981-01-01 and ending at 1996-12-31, I try each password combination (shown above) with each name in every iteration of the loop. And iterate the date by one day after each loop.

d. If a response contains the string “ACCESS GRANTED,” the loop breaks and the successful password is printed to the screen.

e. Then the password can be pasted into the login with “admin” as the username.

C. References:

a. https://en.wikipedia.org/wiki/ISO_8601