1. Cloud Strike

#include <stdio.h>

#include <string.h>

#include <ctype.h>

#define MAX\_LENGTH 100

int ppassword(char \*password) {

int total = 0, i, u = 0, l = 0, d = 0, s = 0;

if (strlen(password) >= 8)

total++;

for (i = 0; password[i] != '\0'; i++) {

if (isupper(password[i]))

u = 1;

else if (islower(password[i]))

l = 1;

else if (isdigit(password[i]))

d = 1;

else

s = 1;

}

total += u + l + d + s;

return total;

}

void vulnerability(char \*password) {

if (strlen(password) < 8)

printf("Password is too short\n");

if (strpbrk(password, "ABCDEFGHIJKLMNOPQRSTUVWXYZ") == NULL)

printf("No uppercase letter\n");

if (strpbrk(password, "abcdefghijklmnopqrstuvwxyz") == NULL)

printf("No lowercase letter\n");

if (strpbrk(password, "0123456789") == NULL)

printf("No digits\n");

if (strpbrk(password, "!@#$%^&\*()\_+{}|:>?<,./;[]\\/-=") == NULL)

printf("No special character\n");

}

void file() {

FILE \*file = fopen("credential.txt", "r");

if (!file) {

printf("Error: Unable to open file.\n");

return;

}

char line[MAX\_LENGTH];

printf("Compromised Credentials:\n");

while (fgets(line, sizeof(line), file)) {

char user[MAX\_LENGTH], password[MAX\_LENGTH];

if (sscanf(line, "%99[^:]: %99[^\n]", user, password) == 2) {

printf("User: %s, Password: %s\n", user, password);

printf("Password Strength: ");

int total = ppassword(password);

printf(total <= 2 ? "Weak\n" : (total == 3 ? "Moderate\n" : "Strong\n"));

printf("Identified Vulnerabilities:\n");

vulnerability(password);

printf("\n");

} else {

printf("Error: Invalid file format.\n");

}

}

fclose(file);

}

int main() {

file();

return 0;

}