Password Validator & Generator

Michael Cummins

Marist College

Abstract

I have developed a program that takes input from the user and decides whether or not the given string is a strong enough password that meets standard password best practices. However, a second function of the program is that users are able to choose either if they want to input a password that would be checked for strength, or be randomly given a secure password. For randomly generated passwords, a hint will be given to help users remember it. For example, if the program outputs “jKx4NY>"tZY)xk=>” another line would read “jack KOREAN xbox 4 NUT YELP > " tokyo ZIP YELP) xbox korean = >.”

In order to provide flexibility in the system, the user would also be able to check which preferences they want in order to meet their needs. For example, some people may not be able to remember where certain special characters go in their password. To counter this I feel that if a user were able to select which special characters appeared in their calculated password, they would be provided with a better more personalized service. While this service may offer the ability to reduce strength, I believe that it is better to implement this into the system in order to serve a wider array of potential users. If a user submits a phrase that is easily guessable such as “password, “P@ssw0rd”, “123”, or “P@ssw0rd123” the program will give them a randomly generated password that does not take into account their input. The program would also output a message stating why the given input was not taken into account when determining the new password.

Introduction

Internet security and privacy is of vital importance in the technological age. As a Computer Science student with a minor in cybersecurity, I have learned that very few people actually know how to protect their personal credentials. My hope is to develop a program that will make it easier for people to know exactly what makes a password strong.I plan to accomplish this by not hiding the methods that are used to decide if a users' input is weak or strong.

Requirements

The user must have a computer with the eclipse IDE downloaded with the windowbuilder library in order to properly run the program. The computer does not need to be connected to the internet in order to access the program.

Literature Study

For password validators, nothing else exists out there that uses a DFA to discern how strong the users' password is. In terms of the portion of the program that will randomly generate a password, there are plenty of online programs that will do that for you. I felt that if my program offered both validation and generation it would be more flexible and complete.

User Manual

The program is easy to learn and easy to use as the main goal of it is to be operated by users who may not be as tech savvy as the average person. Users need only input a string that consists of alphabetical, numerical, and special (!,@,#,$,%,^,&,*,(,)) characters. After doing so, they can then click a button that will tell them whether or not their given input is secure and meets all guidelines. The password generator is accessed by clicking the “generate password” button located on the bottom of the form. It will display a pop-up window with the generated password.

Conclusion

In conclusion I believe that this program is a necessity in order for the general population to be informed as to how they can better protect their personal information.