18CSE361T Web Programming

Register Number : RA2011031010031

Name of the Student: Stuti Jain

Semester / Year : Vth / IIIrd

Department : CSE

Specialization : INFORMATION TECHNOLOGY

SRMI INSTITUTE OF SCIENCE AND TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR -603 203

BONAFIDE CERTIFICATE

Register No.

Certified to be the bonafide record of work done by *Stuti Jain* *of* CSE *IT* *, B. Tech Degree course in the Practical* **18CSE361T**

Web Programming *in SRM Institute of Science and Technology, Kattankulathur during the academic year 2022-2023(Odd Sem).*

Signature of the Faculty **Signature of the HOD/NWC**

Submitted for University Examination held on SRM Institute of Science and Technology, Kattankulathur.

Examiner-1 **Examiner-2**

INDEX: -

ABSTRACT: -

Text based passwords are the most widely used authentication mechanism in multiuser environment. Creating strong password is often a difficult task for users. Multiuser environments employ some password composition policy for users, which require a password containing alphabets (both lowercase and uppercase), numerical digits and special symbols. These policies become obstacle for users in creating good password that satisfies the specific policy. Users favor memorability factor without considering its security against an attacker. Memorability is important because if a user forgets password then usability of the system decreases. This paper presents a ProActive random password generation technique. A random password is generated by inserting random digits and special symbols in a randomly chosen word. Generated password is checked against certain attacker approaches through ProActive analysis. If ProActive analysis results positive then password is discarded and process starts all over again. To help users in memorizing password, both how the password is generated and word used to generate the password is sent.

Introduction: -

Text based username-password is the most commonly employed authentication mechanism in many multiuser environments. These multiuser applications, while registering users to their application, some applications allow users to create password their own and others generate random password and supply to users. Various surveys have shown users created passwords are less secure than system generated passwords. Most user created passwords can be found in common password lists on internet. The user created passwords can be guessed easily, with a bit of social engineering like user's personal information or type of application. System generated passwords cannot be guessed easily and have no relevance with the user's personal information and type of application but are hard to remember.

To enhance the security of user created passwords, system administrators and organizations employ a set of rules, called password composition policy, which users should incorporate in their passwords. One of the password composition policy suggested by NIST is that a minimum of eight character length password and it must include at least one uppercase letter, one lowercase letter, one digit and one special character [1]. The purpose of such policies is to expand the search space of passwords. An important fact is that when users are allowed to select their password, they favor memorability of password without considering its security against an attacker. In environments where such password composition policies are used, users try to find an easy escape from it. For example, users create passwords like "#Diamond#", "Alok@123", which can be guessed easily and weak but satisfies the policy. Text based password authentication systems involve a tradeoff between security and memorability of passwords. Some passwords are easy to remember but also easy to guess for an adversary. Random passwords are hard to remember and hard to crack because they are made up of arbitrary sequence of characters [2]. Several studies have examined how password composition policies affect users. In a study by Komanduri et al. reveals how password composition policies influence the predictability of passwords and as well how they affect the user behavior and sentiments. Their results demonstrate that successfully creating a password is significantly more difficult under stricter password composition policies. They measured how many people failed at least once to create an acceptable password and further observed how users deal with it [3].

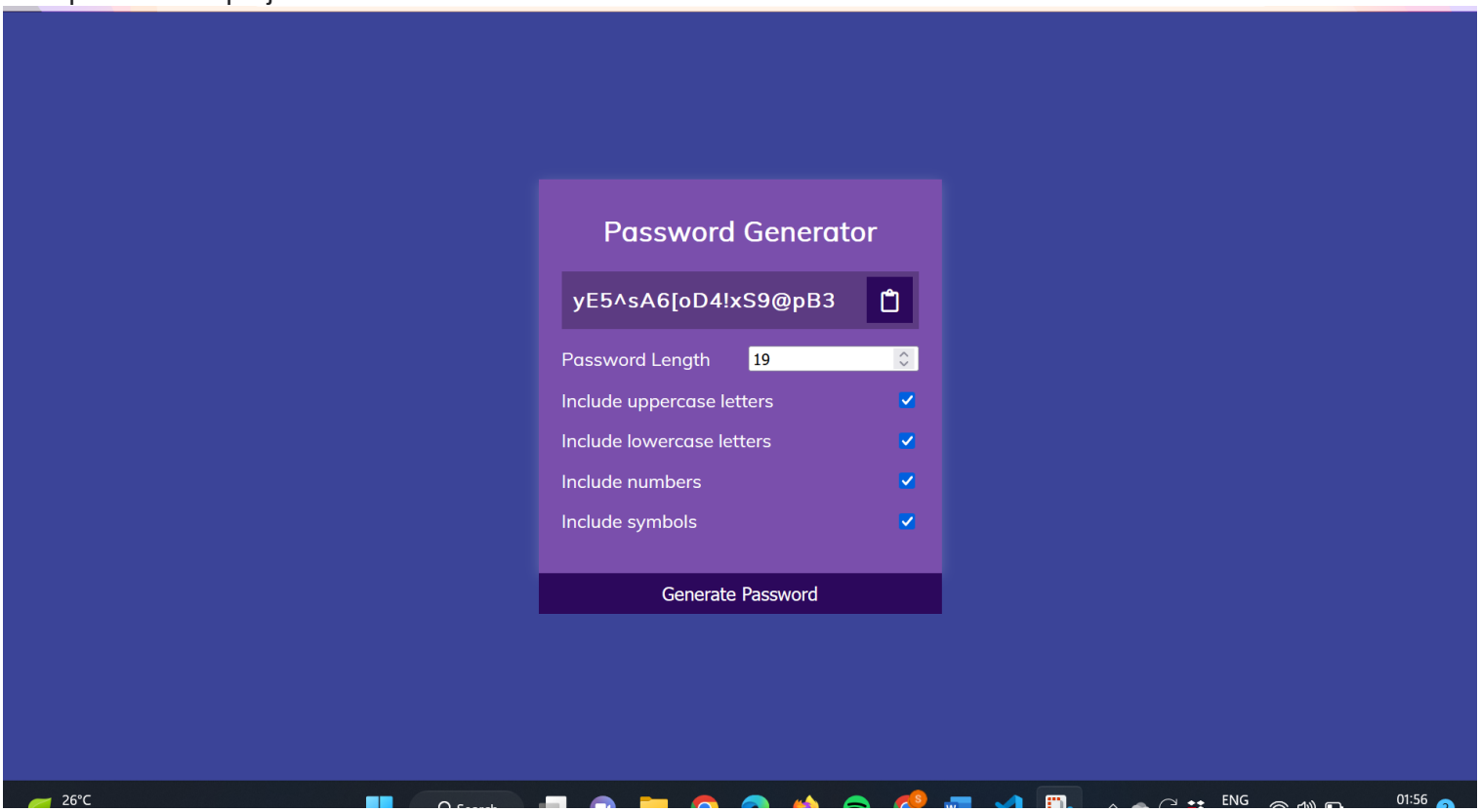
Requirements: -

- VS studio code

Technologies used: -

- HTML
- CSS
- JAVASCRIPT

Snapshot of the project: -



Conclusion: -

The password generator was deployed safely and in working condition.