**Fredrick Onduso Ondieki**

**CST-221 Operating System Concepts**

**CST- User Interface and Security Assignment**

**John Zupan**

**April 21, 2019**

**Git Url: https://github.com/FREDDYSMALLZ/Operating-Systems-Concepts-CST-221.git**

**CST-221 User Interface and Security**

Activity Directions:

In this assignment we are required to develop a user interface for **three different functions** of an operating system and how each of the functions could be secured using various authentication schemes. Secondly, we are required to support the three different authentication system (one single factor, one two-factor, and one multi-factor) choosing features that deem suitable to design the system. Most importantly the design will address three main security areas:

1. Proposed Security Authentication Scheme
2. Proposed Security Authentication Factor
3. Required Security Policies

**Single Factor Authentication Scheme**

The single factor authentication system secures a system for example a network or website that requires and identifies its users by using only one category of credentials (Rouse, 2015). The most common single factor authentication we have today is password based. The password mostly relies on the system administrators or the user who sets up the account for use. One of the best practices that a user can employ is creating strong passwords which ensures that no one can access or guess it easily.

In the attempt to create a single factor authentication scheme, I decided to enable *sudo* from the terminal. This is where the *sudo* superuser scheme is executed in the terminal, hence the user interface is blended into the terminal itself. The authentication factor then would be the password entered for the superuser account. Required security policies include but are not limited to; **password validation, password anonymity, password entry timeout**, and **unique character processing.**

• Password must be entered in the terminal

• Password characters will not be displayed back to the user

• String length above max length will not be submitted and generate an error

• Non-Alphanumeric characters will not be accepted

• Password will be encrypted and compared to stored password

• Password will be stored in encrypted form

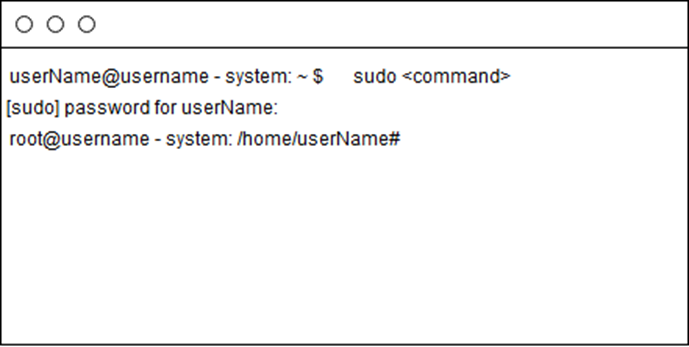
• Method will check for proper input, blank field will be rejected

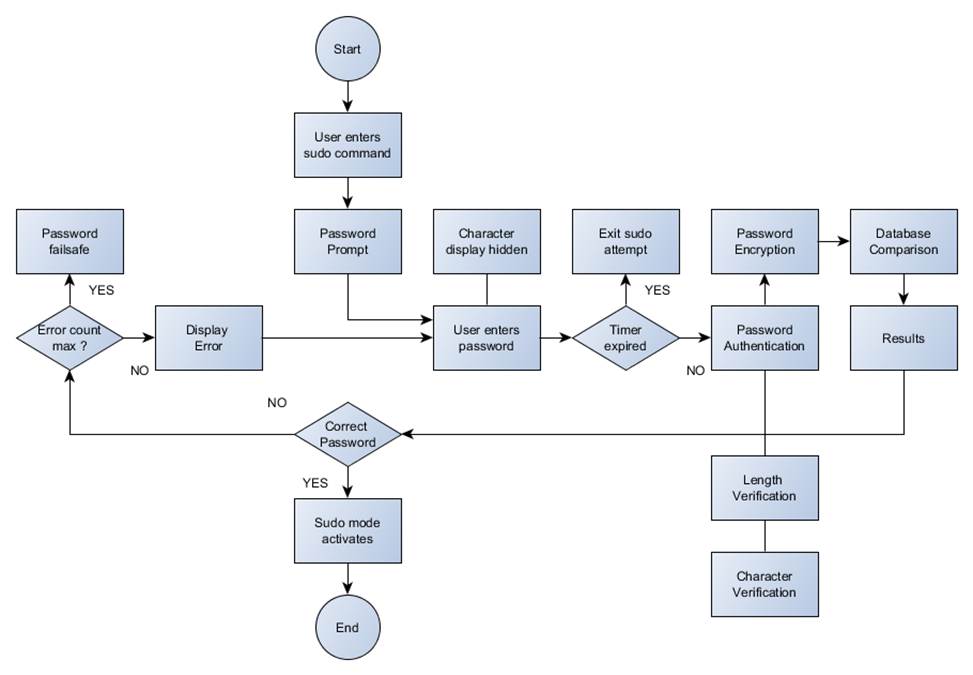
• Trigger failsafe if wrong password entered too many times

• Discard sudo attempt after set duration of user idleness

• Password restrictions, minimum 8 characters, 1 special character

The screenshot and flow chart illustrate how the single factor system works where the sudo user superuser scheme is executed in the terminal.





**The two-factor authentication Scheme**

A two-factor authentication system is an additional layer of security to the single factor authentication scheme. This helps businesses using such system to address any vulnerabilities that may come alongside the standard password-only approach. In addition, Two-factor authentication methods rely on users providing a password as well as a second factor, usually either a security token or a biometric factor like a fingerprint or facial scan.

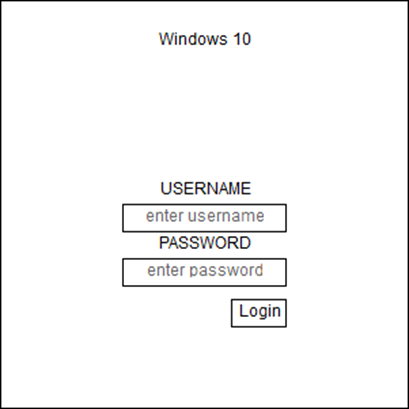
**Username and Password verification**

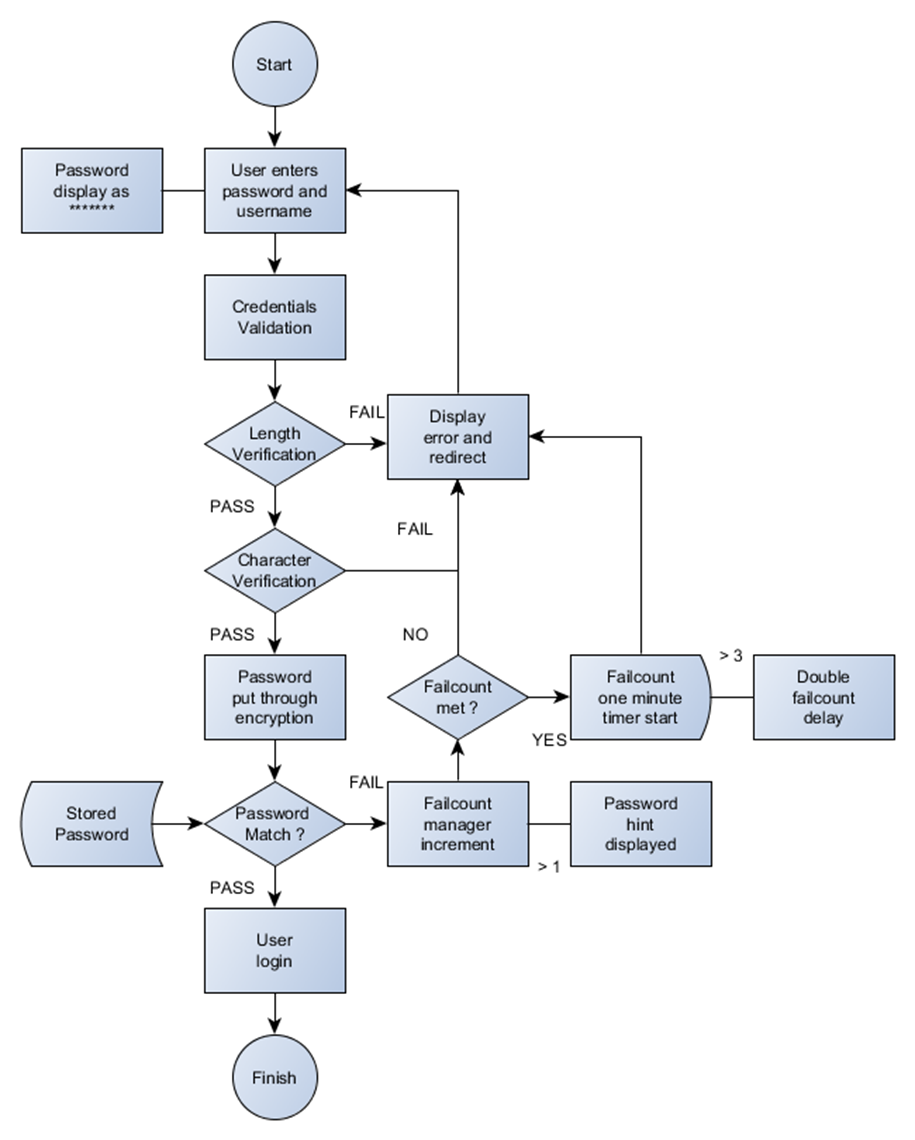
The username and password verification for the two-factor authentication occurs for login purposes thus has its own dedicated screen. What a user needs for such a scheme is to have the right username and password to access the resources assigned to that account. Security policies on this scheme include but are not limited to; **username validation, password validation, password encryption, password anonymity, unique character processing, length verification**.

* Username and password tied to specific host on system
* Username will have a max length of 50 characters
* non-alphanumeric characters are supported for username and password
* Password will have a max length of 25 characters
* Password restrictions, minimum 8 characters, 1 special character
* Encryption supported for password field only
* Password will be displayed as bullets instead of character representation
* Password reminder triggered after failed attempt
* Password locked for 1 minute after 3 failed attempts, time doubles for additional failures

**The Screenshot and the flow chart**

When the username is entered, the characters are displayed on the username field. On the other hand, the password is entered on the password field and hidden in form of special characters. Both the username and password need to be validated and authenticated so that the user will be authorized to access the resources.





**Multi-factor Authentication Scheme**

This is a system that requires more than one form of system authentication method from the list of independent categories of credentials to ascertain and authorize the user to log in to a system account. According to Rouse, 2015, this system combines two or more non-dependent credentials; password, security token and biometric verification.

The main objective of multifactor authentication is to add one more layer of security to the two-factor authentication scheme which in return will make the unauthorized user to have difficult time accessing computing devices, databases or even networks. For example, even if the attacker breaches the first layer of security, the attacker would at least have another layer of security to breach before the system resources can be accessed.

**Database Access**

Most of the database connections are established within the program functions. These connections are established though a set of interface connections which can be a combination of multiple factors. These include, database port, database name, root username, and root password. Security policies include but are not limited to**; port authentication, username authentication, password authentication, database authentication and connectivity monitoring**.

• Port input checked for length and numerical data

• Username authentication compared to database entry

• Username length check, maximum characters 50

• Password restrictions, minimum 8 characters, 1 special character

• Password encryption

• Password length check, maximum characters 25

• Password authentication compared to database entry

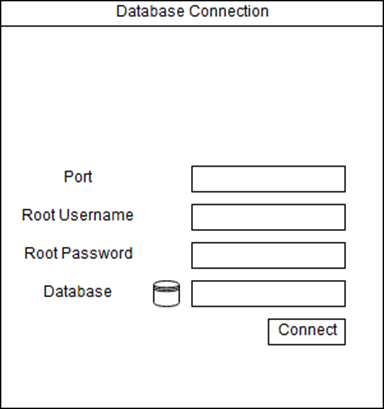
• Target Database authentication

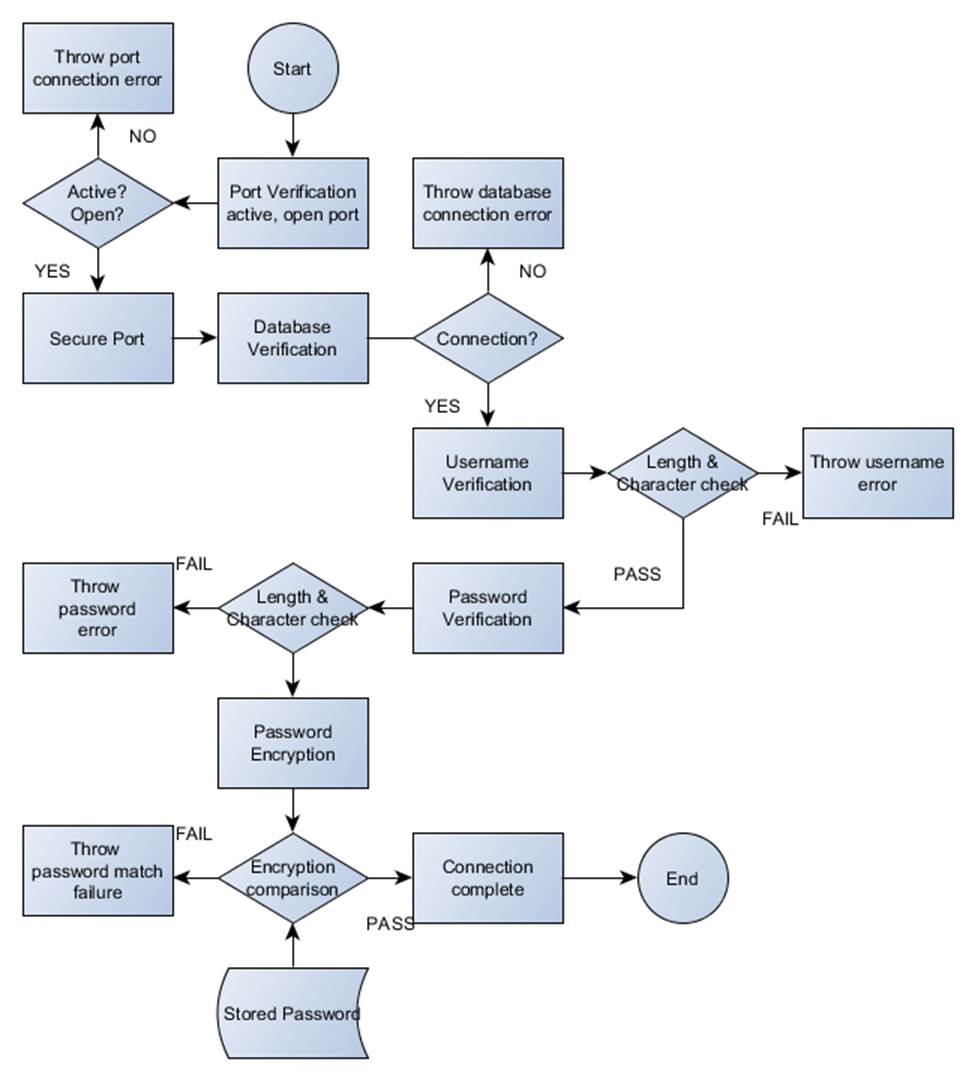
• Database length check, maximum characters 50

Multi-factor authentication has received allot of attention recently mostly on the healthcare sector because of the way the technology can better protect against unauthorized data access. Multifactor authentication examples have existed in one form or another for many years. In recent times, however, one of their primary goals has been to eliminate traditional passwords. Passwords can be difficult to remember (because they change frequently), may be easily compromised and can lead to chain-reaction breaches when a user chooses the same password to secure multiple, disparate resources (Possey, 2018).

Some of the benefits of multi-factor authentication scheme include; Improve Security, Achieve Compliance, Increase Flexibility and Productivity. Lastly, With the increase of cyber-attacks on organizations, password strength cannot be relied on as the only layer of protection for an organization to preventing threat actors from gaining unauthorized access. Although not bullet-proof, multi-factor authentication is a proven way to lessen the likelihood of a data breach via a compromised password.

**The Screenshot and the flow chart**





References

Carter, S. (2017, June 14). The Challenges and Benefits of Multi factor Authentication - MFA 101, Part 2. Retrieved April 21, 2019, from https://blog.identityautomation.com/the-challenges-and-benefits-of-multi-factor-authentication-mfa-101-part-2

Hald, D. (n.d.). 8 reasons you should turn to multi-factor authentication. Retrieved April 21, 2019, from https://techbeacon.com/security/8-reasons-you-should-turn-multi-factor-authentication

Possey, B. (2018, February). What are some useful multifactor authentication examples? Retrieved April 21, 2019, from https://searchhealthit.techtarget.com/answer/What-are-some-useful-multifactor-authentication-examples

Rouse, M. (2015, March). Multifactor authentication (MFA). Retrieved from https://searchsecurity.techtarget.com/definition/multifactor-authentication-MFA

Rouse, M. (2015, March 15). Single-factor authentication (SFA). Retrieved April 21, 2019, from https://searchsecurity.techtarget.com/definition/single-factor-authentication-SFA