

Password Protection

Find the Code On: <https://github.com/kdutta9/CS-Education>

Note: This is intended for beginners in cyber-security, just to display how the code behind an authentication system works. Then, you can walk through a SQL Injection, to be noted later.

Overview: Creating a Python script (can be done in any language) that can only allow an authorized user to log in.

Writing The Code:

First, start with how we can log in some user with some username and password, using Python input statements and print statements. It will look something like this:

```
correct_user = "name"
correct_pw = "pass"

user = input("Please enter your username.")
pw = input("Please enter your password.")

if correct_user == user and pw == correct_pw:
    print("Congrats, you're in!")
else:
    print("Account not recognized. Please try again.")
```

Great, we're able to log someone in! But it seems like it'll take a long time to do this for many accounts. Let's store all our usernames and passwords in a dictionary, so that we can check our authentication efficiently.

```
auth = {
    "user1": "pass1",
    "user2": "pass2",
    "barney": "din0saur",
    "elmo": "red123",
    "johnny84": "cobrakai4life!"
}

user = input("Please enter your username.")
pw = input("Please enter your password.")

if user not in auth or auth[user] != pw:
    print("Account not recognized. Please try again.")
else:
    print("Congrats, you're in!")
```

Awesome, we can handle many users! However, sometimes we mistype, so it may be a good idea to add functionality to give 3 login attempts. We can use a while loop with a counter variable, and then stop the function when needed.

```
auth = {  
    "user1": "pass1",  
    "user2": "pass2",  
    "barney": "din0saur",  
    "elmo": "red123",  
    "johnny84": "cobrakai4life!"  
}  
  
user = input("Please enter your username.")  
pw = input("Please enter your password.")  
  
if user not in auth or auth[user] != pw:  
    print("Account not recognized. Please try again.")  
  
else:  
    print("Congrats, you're in!")
```

Nice job coding this! Now, let's do an interactive tutorial on how hackers may exploit these authorizations, and how we can protect it (SQL Injection).

Tutorial: <https://www.hacksplaining.com/exercises/sql-injection>