Improper Restriction of Excessive Authentication Attempts

We have likely all forgotten a password and have been locked out of an account after too many unsuccessful attempts. Without this protection, a hacker’s chances of guessing a user’s password skyrocket. They can simple try to guess it as many times as they want.

In order to mitigate this vulnerability, we create a simple login program that asks a user for their username and password up to 3 times. If they get it correct, they get to login, if they do not, they get up to 3 attempts before they get an error message.

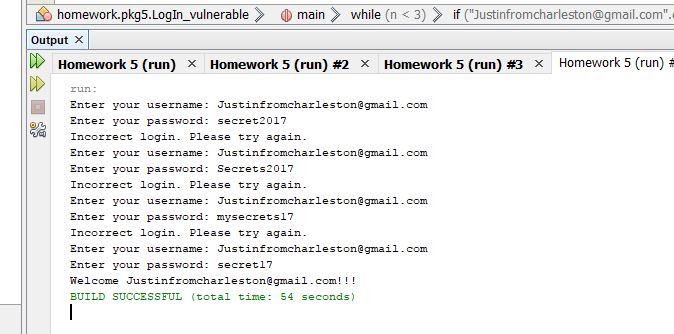
Here are the correct credentials for our program:

Username = “Justinfromcharleston@gmail.com”

Password = “secret17”

Vulnerable-

As you can see, our application lets us attempt to login more than 3 times, giving the user a higher chance of guessing a password correctly. Four our statements, n will always be less than three because there is no code that tells it to change. This gives a hacker an unlimited number of opportunities to guess the user’s password.



Mitigated-

In order to mitigate this vulnerability, we simply add a nested if statement that tells while loop to break when the user has entered 3 incorrect logins. Our n variable increments by 1 each time they enter incorrect credentials. In our screen capture, we can see that the application has cut us off at 3 unsuccessful attempts.

