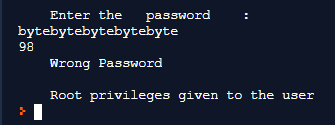
**Sean Poston**

**Dr. Mitra**

**CY201 Lab 02**

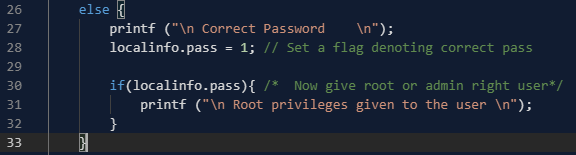
<https://repl.it/@seanposton4/CY201Lab2>

**Q2:**



Obviously, this isn’t the desired result. If a user inputs too many characters for the password, it will cause a buffer overflow that causes the “pass” flag to flip and allow root privileges for the user. There are a couple things that can make this more secure.

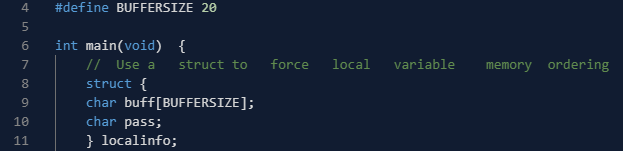
**Q3:**



One is to move the root privileges statement inside of the else statement with the “Correct Password” line. You don’t want to give root privileges without receiving a proper password, so it doesn’t make sense for the root privileges to be separate from the correct password.

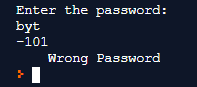


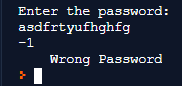
Another move to secure this more, is to limit the input of the user, as to avoid segmentation faults. scanf is much more secure than gets, and it allows for the limiting of input values.

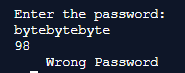


Finally, just to help matters, we can increase the size of the buffer to help avoid users accidentally putting in too many characters. This isn’t necessary after the other steps are taken, but it’s still helpful.

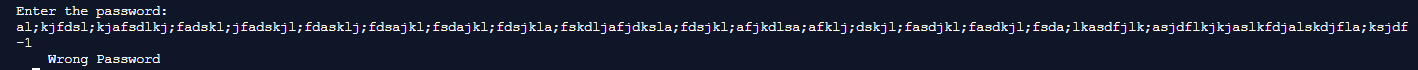
**Input Tests:**







One more for good measure:



Correct Password:

