Matthew Kachar

COSC4931

2/11/16

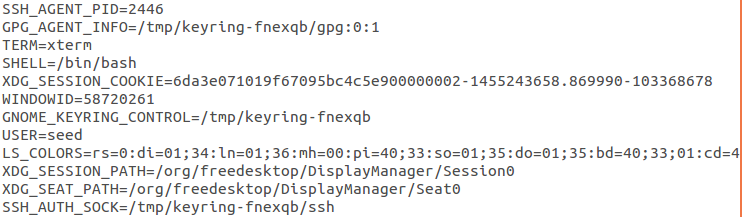
Project 3

Environment Variable and Set-UID Program Lab.

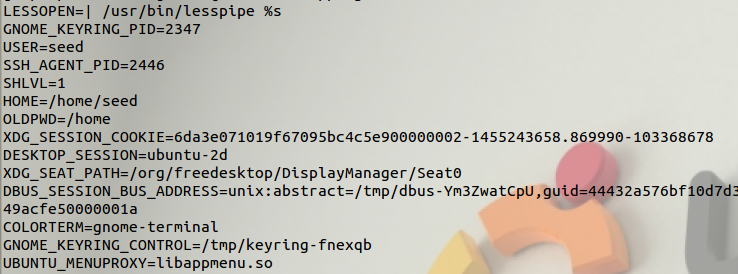
Lab Objective: Understanding environment variables and their effect program and system behavior as well as understanding the importance of identifying these effects as a programmer.

For the first task of compiling the given C code to create to separate parent/child output files. It seems as though each file is a carbon copy of the other, as illustrated by the diff command. The second program calls the address which has access to where the environment variables are stored whereas the first program runs the process of printing the data in each address using an array to call them.

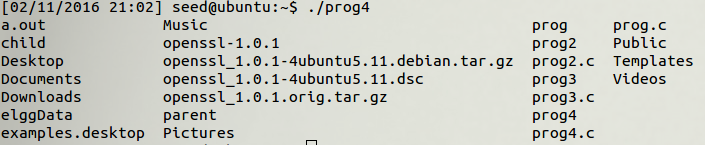
The output between calling the array of environment variables and the execve() function is virtually the same.



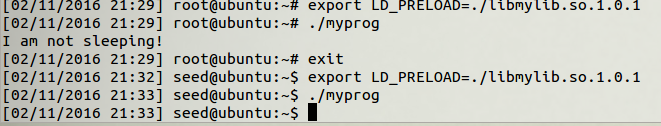
However calling the environment variables using the system function looks since it does not print the SSH Agent and process ID first



As for giving the root privilege to the program, this allows the user to access and change the environment variables which are also called by the child processes. This can be particularly detrimental when forking other processes. We can also give programs access to directories the same as other commands like ls. In this example the program is executing the system function with root privilege.



The Set-UID program has the ability to change other program outputs dynamically through the OS by way of the loader/linker system. In the lab example the output of myprog.c remains the same until the username variable changes from root to seed.



By giving a program certain privileges and executing which uses certain function such as the system() function, we give the user an undesired amount of access to root directories which can have perhaps unintentional consequences. And it is entirely possible for the user to do damage to the system or its library. Something as simple as implementing functions without a return inhibits the user from modifying or wiping out files in one step.