
COSC 5326 Final Project (200 Points)

Suspense: December 07, 2017 NLT 6:00PM

OBJECTIVES:

- Demonstrate mastery of bash shell scripting and the Linux OS.
- Write a bash shell script that makes use of various Linux terminal commands.
- Understand file management, permissions, user accounts, groups, services in Linux.
- Demonstrate creativity and originality.

PROBLEM DESCRIPTION:

1. This assignment is individual effort. Evidence that you received help on this assignment will result in a zero or may be referred to the University as plagiarism.
2. Recall the COSC 5326 Midterm Exam. During that test, you were tasked to use Linux terminal commands to answer questions in a “Capture the Flag” (CTF) style format. For this project, your task is to create a bash shell script and associated documentation that will create a small set of Linux OS CTF-like challenges/puzzles.
 - a. Part 1: Create a bash shell script named `challengeinstaller.sh` that, when executed as root privileges, will install a set of challenges/puzzles on the Linux system upon which it is run.
 - i. The script sets up the challenge environment by creating user accounts, directories, and files; by setting permissions and passwords; by configuring/starting services, etc.
 - ii. Then, once the script completes execution, the puzzles can be attempted.
 - iii. The shell script provides portability; that is, it can be copied onto any Linux system, executed, and then the challenges can be attempted.
 - iv. Your script should create at least four challenges of increasing difficulty (one easy; one moderate difficulty; and one difficult).
 - v. The challenges should test a user’s knowledge of the Linux OS, terminal commands, etc. and their problem-solving skills.
 - b. Part 2: Create a plain-text document called `challengedocument.txt` with a set of questions (one question per challenge) to help guide the user in solving the challenges.
 - i. The answer to each question should be a flag (e.g., a hash value).
 - ii. For each question, the document should also have the answer to the question.
 - iii. For each question, the document should describe a walkthrough of how the puzzle can be solved.

- iv. Your document should also contain installation instructions for the script: where to place the script; how to execute the script; etc.
 - v. Your document should also contain any special instructions for playing the challenges (e.g., challenges require `ssh` login).
- 3. A significant portion of your grade will be based on the cleverness and uniqueness of each challenge (compared to your classmates). Thus, I do not expect to see many, if any, challenges that are similar to each other.
 - a. Each of your challenges should be distinct; that is, from each other.
 - b. Your challenges should be distinct from those of other students; though do not coordinate with other students to ensure that they are different.
 - c. Your challenges should be distinct from those that you saw on the midterm exam.
- 4. Ideas for challenges:
 - a. One or more challenge might rely on a previous challenge. For example, each challenge is contained within a separate user account and the flag of the previous question is the password for the next user account.
 - b. Perhaps design a problem with an intended security flaw (e.g., permissions set incorrectly) so that the challenge is to “hack” the account). The security flaw should not be so obvious that it fails even a cursory test. Also, other than the intended flaw, you should work to make sure that each challenge cannot be solved easily by some unanticipated approach.
 - c. Perhaps one or more challenges require remote access to the system through `ssh` or web browser. In this case, your script should have started the proper server and, if necessary, modified the system firewall to allow remote connections (see the textbook for details).
 - d. Take a look at <http://overthewire.org/wargames/bandit/> for some other ideas.

SUBMISSION:

- Review the Evaluation below to ensure you have met all the requirements.
- Thoroughly test your script and your challenges.
- Submit electronic copy of `challengeinstaller.sh` and `challengedocument.txt` to Canvas.

EVALUATION

- | | |
|---|---------------------------|
| a) Project is late or not submitted at all. | -200 |
| b) Shell script does not execute correctly. | -100 |
| c) Does not have at least four challenges. | -50 per missing challenge |
| d) Lack of originality of challenges. | -30 per challenge |
| e) Lack of cleverness of challenges. | -5 per challenge |
| f) Unintended flaw in challenge. | -20 per challenge |

- | | |
|--|------------------|
| g) Necessary server is not started correctly. | -50 |
| h) Installation instructions are unclear. | -50 |
| i) Walkthrough is unclear. | -20 per question |
| j) Special instructions are unclear. | -50 |
| k) Bash shell is inadequately commented for readability. | -20 |