CST–321 Security

Activity Directions:

In this assignment, you will do research on Linux security, as well as develop some bash scripts to support Linux system administration. The following are the tasks you need to complete for this assignment:

1) Research Buffer Overflow:

a) In 100–200 words, explain what a buffer overflow is.
b) Provide a diagram showing what happens to the system (stacks and segments) at runtime when a buffer overflows occurs in a C programming.
c) Explain the issues and harm that a buffer overflow can cause and how this is feasibly possible in a C program.
d) Explain the techniques that are used to prevent buffer overflows, the techniques that are used to defeat buffer overflows, and also what defensive mechanisms have been put in place in the operating system that prevent harm to a system due to buffer overflows.

2) Research Zero-Day Exploit:

a) In 100–200 words, explain what a zero-day exploit is.
b) Read the article "Modernism, Christianity, and Business Ethics: A Worldview Perspective," located in the topic Resources. In 100–200 words, using the article and your research, discuss from a Christian worldview perspective how ethical issues can arise in a zero-day exploit and how this knowledge could be used to benefit you or others around you.

3) Research what Kali Linux:

a) In 100–200 words, describe what Kali Linux is and how it is used to train security professionals.
b) Create a table that lists 10 of the standards tools that are included in the Kali distribution as well as explain the tools' function and how the tool is used in cyber security training.
c) In 100–200 words, from your reading of the article "Modernism, Christianity, and Business Ethics: A Worldview Perspective," as well as the results from your research, discuss from a Christian worldview perspective the ethical issues that could arise related to knowing and using a system like Kali Linux and how this knowledge could be used to benefit you or others around you.

4) Write bash script to test for password strength:

a) Write a script to check and validate passwords. The objective is to flag "weak" or easily guessed password candidates.
b) The password will be provided as an argument to the script. To be considered acceptable, a password must meet the following minimum qualifications:
    I. Minimum length of 8 characters
    II. Must contain at least one numeric character
    III. Must contain at least one of the following non-alphabetic characters: @, #, $, %, *, +, =
c) Your script should print out which tests did not pass, or print a clear message if all tests did pass. Take one or more screenshots of the output demonstrating all the positive and negative test cases. Zip up the source code (not the binaries) in a single zip file.
d) Tips:
    a. Make sure you are using the bash shell (line 1 is #!/bin/bash)
    b. Use the =~ regex operator
    c. Use the newer [[ ]] syntax for the conditional test expressions
    d. Regex Tutorial: https://regexone.com/
    e. Regex Tester: https://regex101.com/

5) Write a bash script to manage users:

a) You will develop a bash script to manage users. The script will take three script arguments: an input filename of users, a group name, and an operation flag.
b) The input filename of users to manage will be a text file with a list of users to add to the system or remove from the system. Each line in the file will contain a User ID and an Encrypted Password, which are separated by a space character as a delimiter. Your input user file should have at least five users. You will need to encrypt your passwords via the command line using openssl and place the generated password in your input user file.
c) Write a script that adds and removes users to a Linux system using the input file provided to the script. The users should be added to group provided in the script argument (using the groupadd command). You will want to check if the group already exists (using the getent group command). Use an operation of -a to signify to add the users to the system and an operation of -r to signify removing the users from the system. When adding users (using the useradd command), make sure you create the users home directory. When you remove the user (using the userdel command), make sure to delete the users home directory. Your script should check that the proper number of script arguments are given, check for empty and blank lines from the input file, and provide feedback to the user while the script is executing. Note, you will need to run your script with elevated privileges using sudo. Take a screenshot of the output and with screenshots proving your user was

added and removed from the system. Zip up the source code for the bash script in a single zip file.

d) Tips:
    a. Use openssl via the command line to generate your Encrypted passwords.
    b. Use the proper options when you add and remove users such that the home directory is created and removed respectively.
    c. Use the proper options when you add a user, so the user is created in the specified group.
    d. Google research prompts:
        i. Using openssl to generate passwords
        ii. How to read lines of a file in a bash script
        iii. How to parse lines of text n a bash script
        iv. How to check if a group already exists

You will want to refer to your textbook readings in Chapter 9 as resources to support this assignment.

Deliverables:

1. Cover sheet with your name, the name of this assignment, and the date.
2. Research results and discussion for buffer overflow.
3. Research results and discussion for zero-day exploit.
4. Research results and discussion for Kali Linux.
5. Screenshot and zip file of code for Password Strength script.
6. Screenshots and zip file of code for User Admin script.
7. Package all of the above into one document and upload it to the digital classroom.
8. Zip up the shell scripts in a single zip file and upload it to the digital classroom.