**CSE 5382 - Secure Programming**

**Assignment 3**

Submitted by: Biswa Ranjan Nanda (1001558251)

Part 1

Submission:

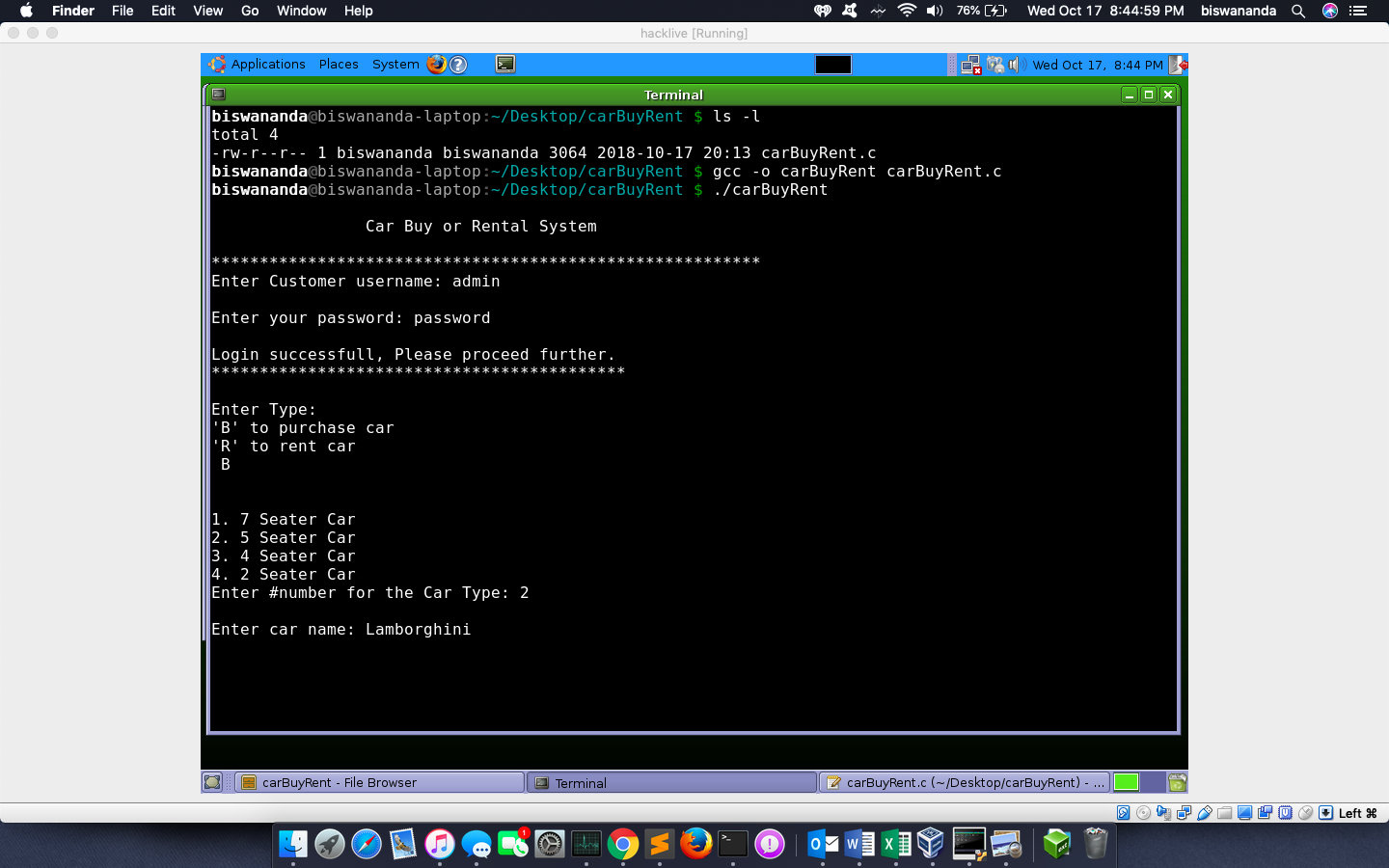
* MacOS – local Machine
* Tools – Oracle Virtualbox
* ISO File - hacking-live-1.0.iso

Steps for execution are mentioned below:

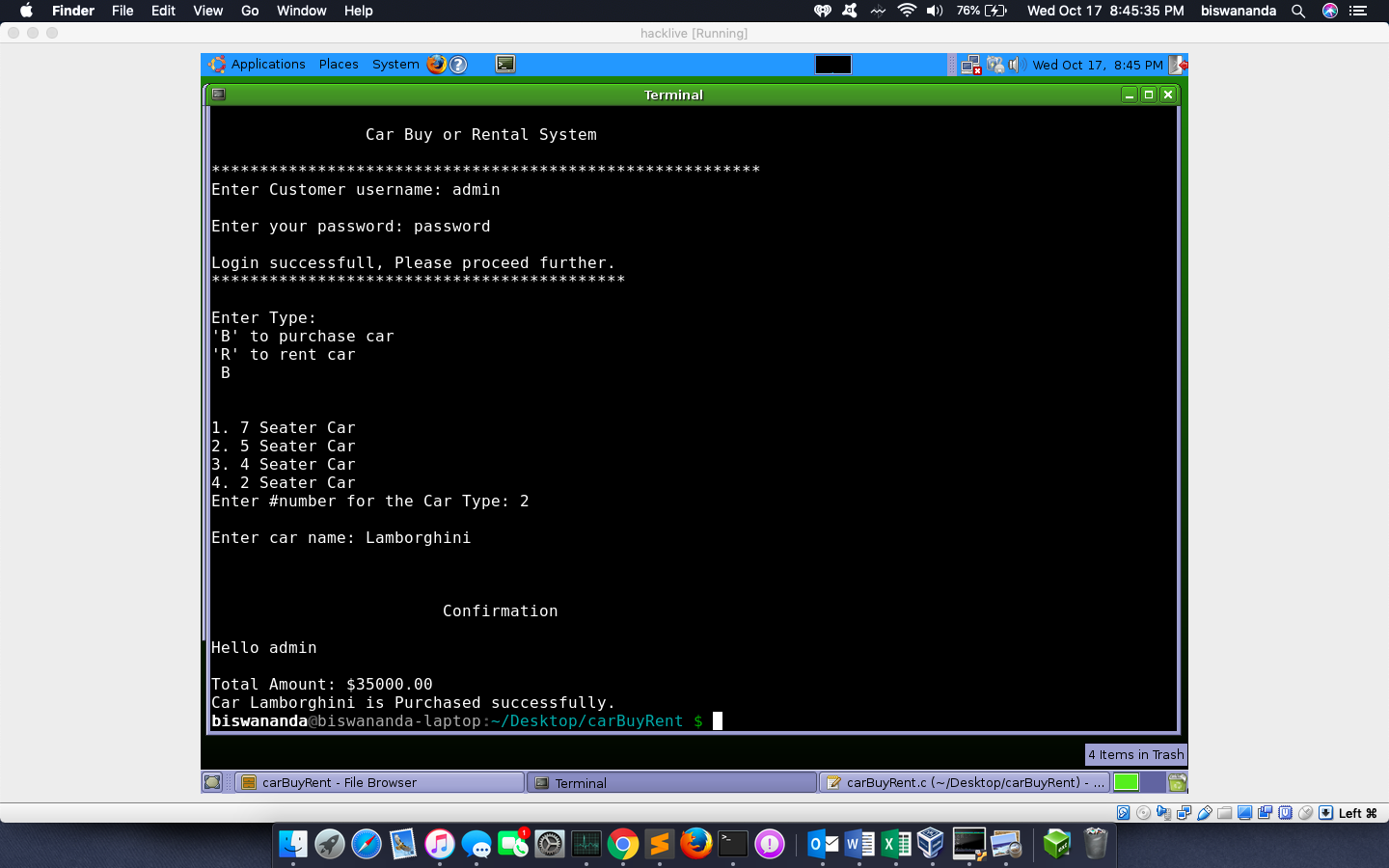
* Source code and the executable file (built in linux machine) are attached in the zip file

1. Type ls -l to list the files in the terminal
2. Type **gcc -o carBuyRent carBuyRent.c** to compile the file, and generate its objectfile carbuyRent
3. Type **./carBuyRent** for execution
4. Enter username as **admin** and password as **password**
5. Choose B for Car purchase or R for Car Rental

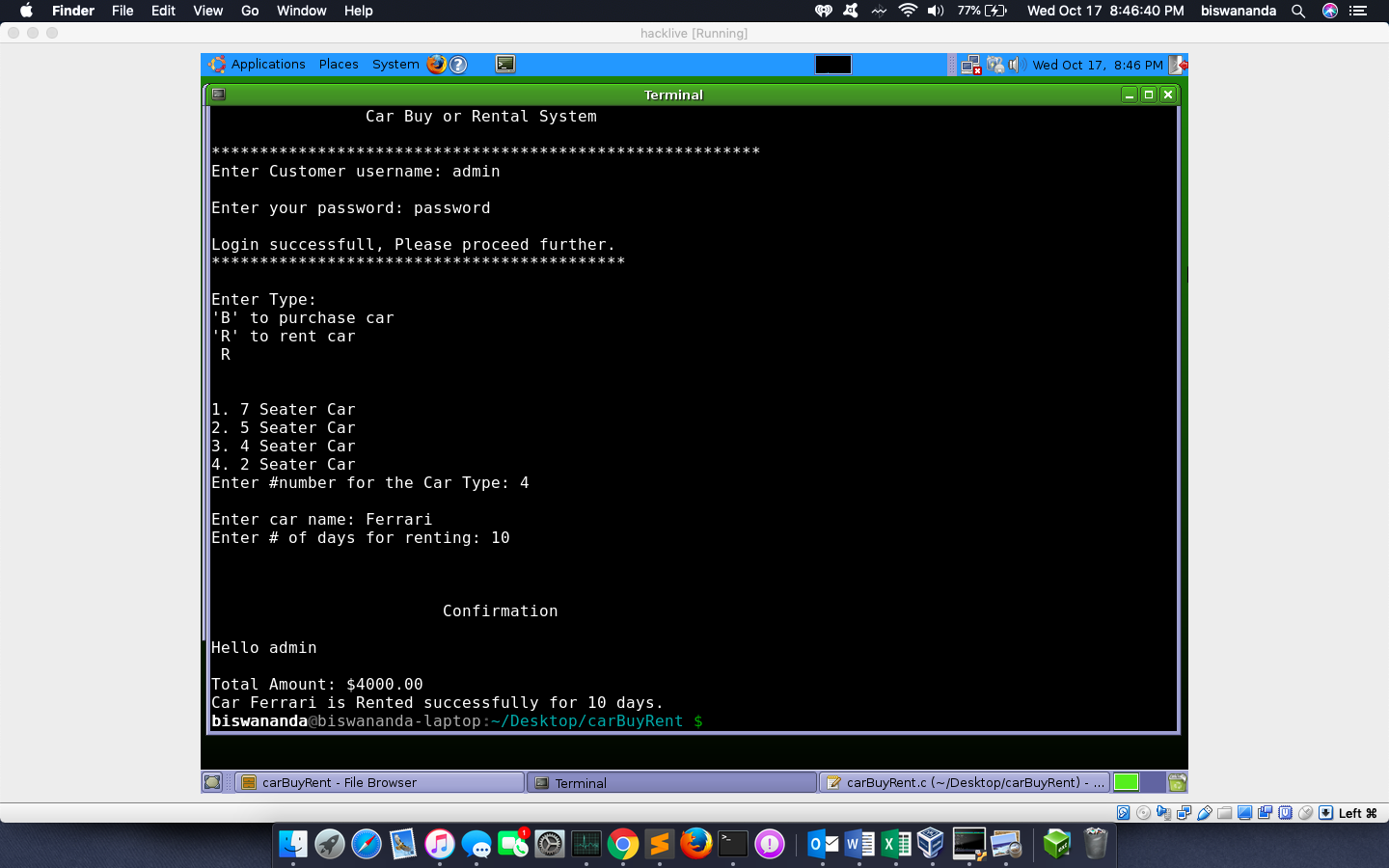
//Buying a car



**//Buying car**



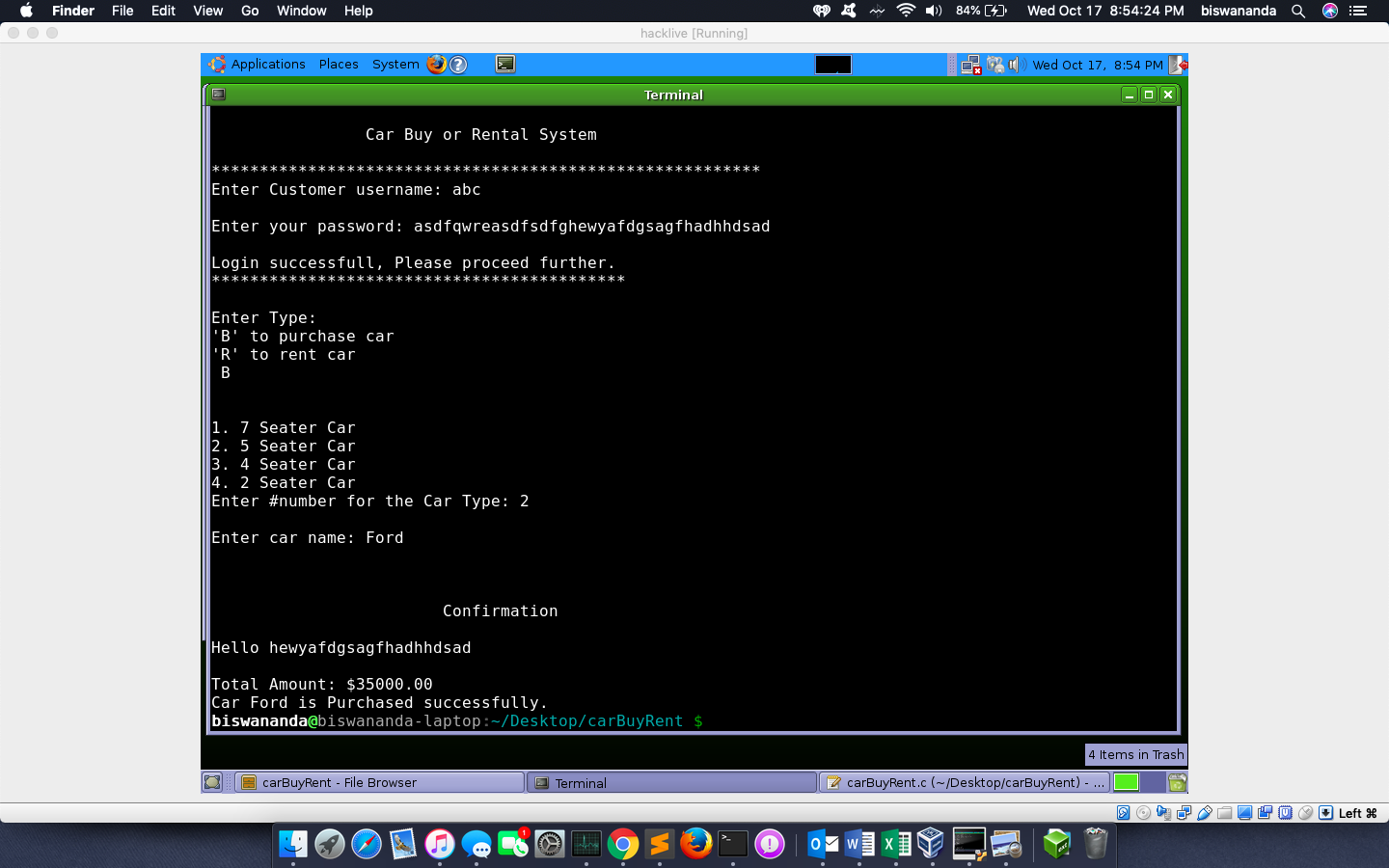
**// Car Rental Request**



**How to Exploit:**

**Exploit method1:**

1. Follow Step1 to Step 3 from **Successful Execution**.
2. Use **./techshop** to execute the file.
3. Enter **username** as **abc** and **password** as **asdfqwreasdfsdfghewyafdgsagfhadhhdsad** This allows us to login due to buffer overflow, as shown in Fig 3. (next page).



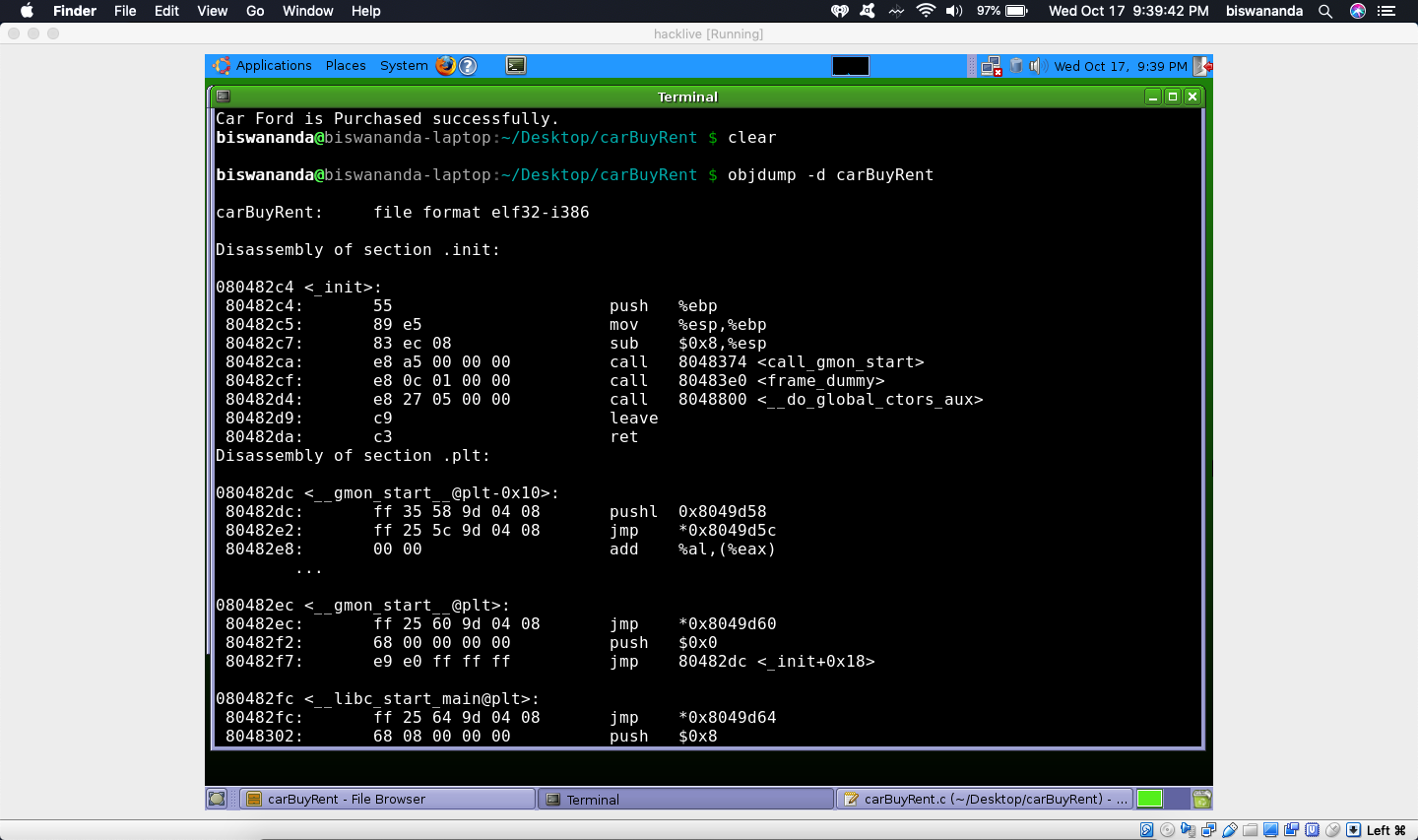
**Exploit Method2:**

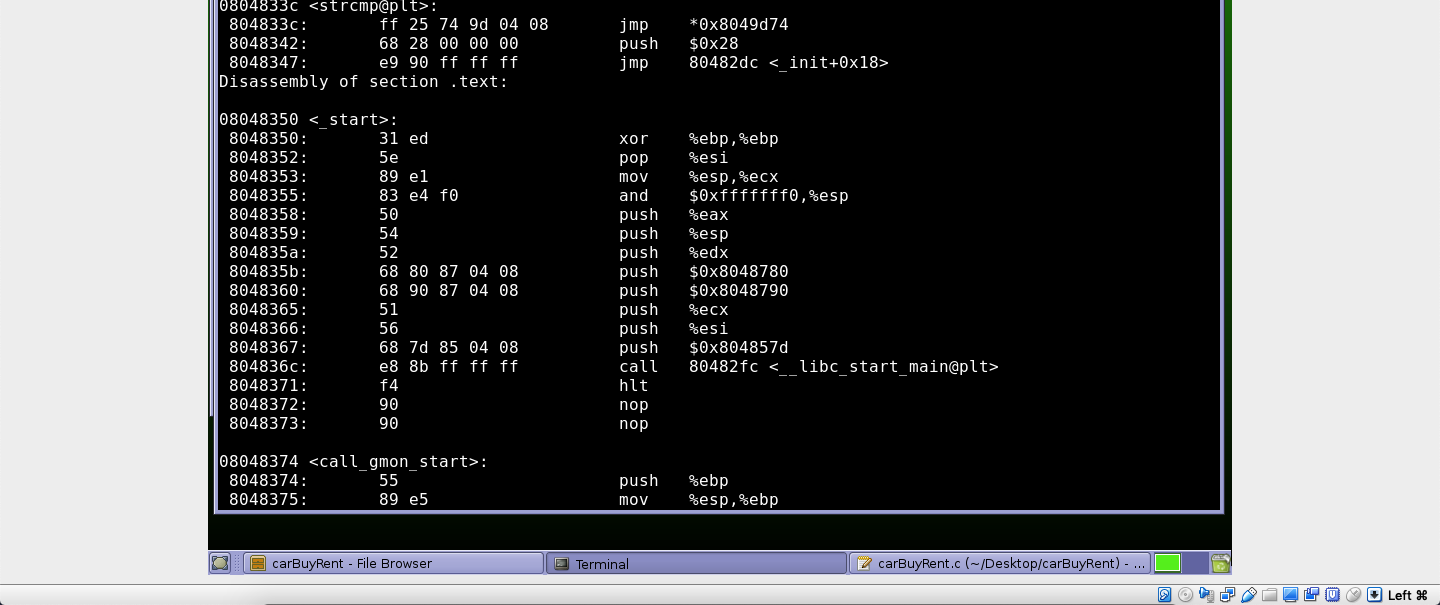
1. Exploit method 2 can be used the hacker doesn’t knows the password length for which he can perform disassembly of the program using **$ objdump -d carBuyRent**.

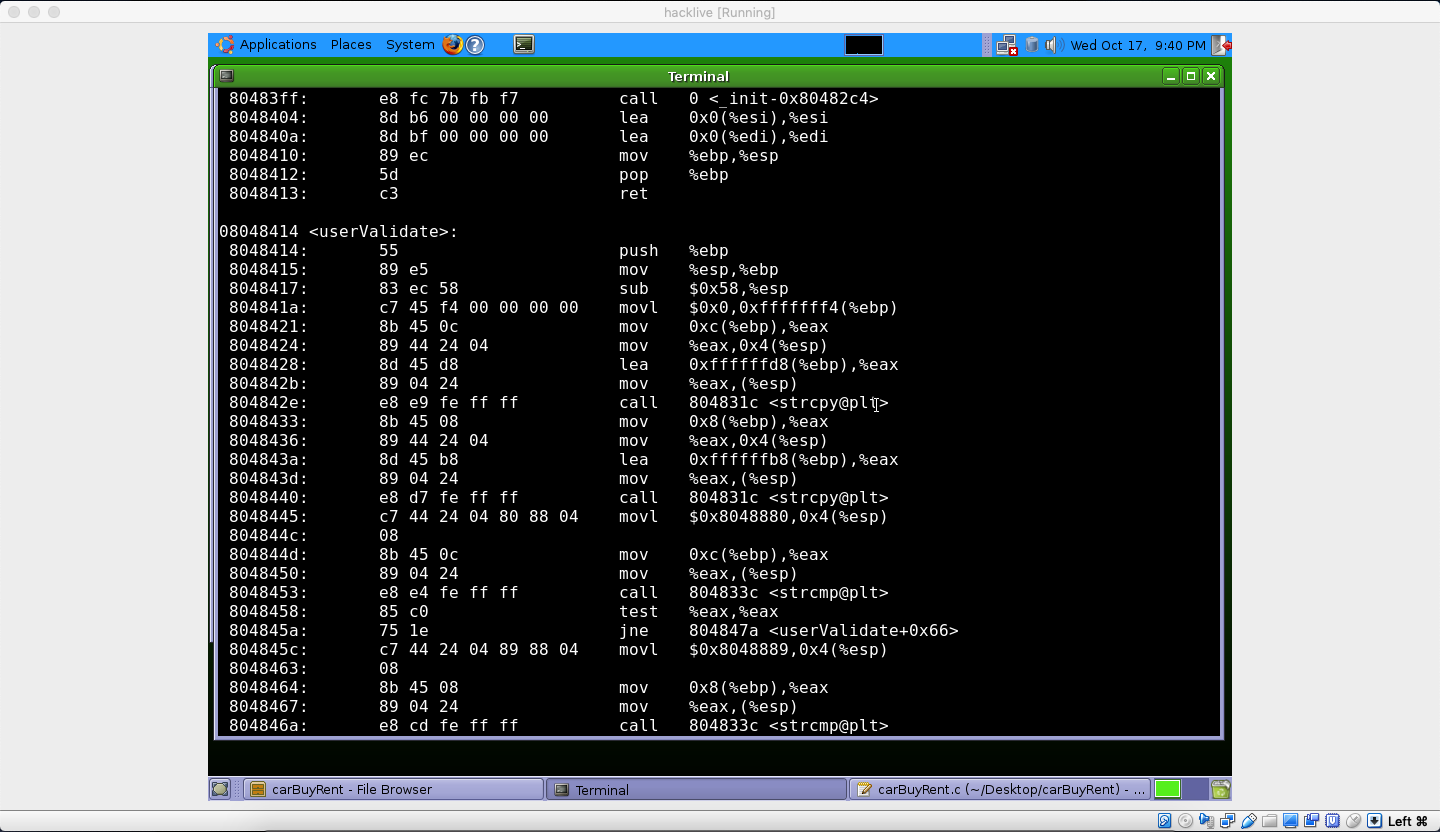
Understanding the concept of memory allocation and de-allocation of every variables and functions, using which we can override the stack and we can understand the buffer size and the address of the variables and functions. This makes it easier to exploit. Using the command as mentioned below, our program can be exploited.

**$python --version 'print "abc"+"a"\*60+"\x14\x84\x04\x08"' | ./carBuyRent where "any login" + "anypassword"**

where **\*n** is the buffer size allocated, and **\xad\xdr\xe\xss** is the return address of the first print statement.







**References:**

**1.** [**http://collabedit.com**](http://collabedit.com)

**2. https://www.youtube.com/watch?v=hJ8IwyhqzD4**

**3. https://www.ibm.com/developerworks/aix/library/au-unixtools.html**

**4.** [**https://dhavalkapil.com/blogs/Buffer-Overflow-Exploit**](https://dhavalkapil.com/blogs/Buffer-Overflow-Exploit)

**5.** [**https://www.thegeekstuff.com/2013/06/buffer-overflow/?utm\_source=feedly**](https://www.thegeekstuff.com/2013/06/buffer-overflow/?utm_source=feedly)

**6. <https://security.stackexchange.com/questions/104411/problems-with->overflowing-the-return-address-on-stack-x86-64**