**CSE 5382 - Secure Programming**

**Assignment 3**

Submitted by: Biswa Ranjan Nanda (1001558251)

Part 2

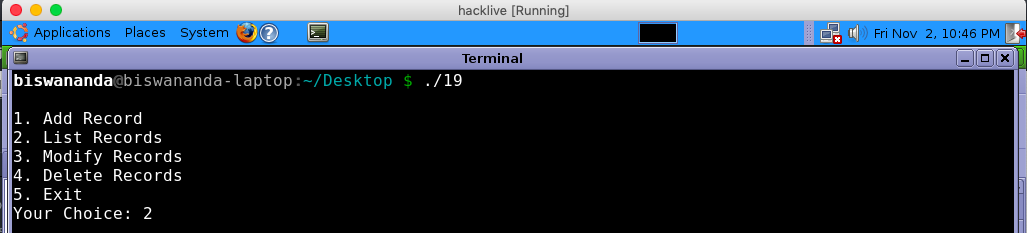
Submission:

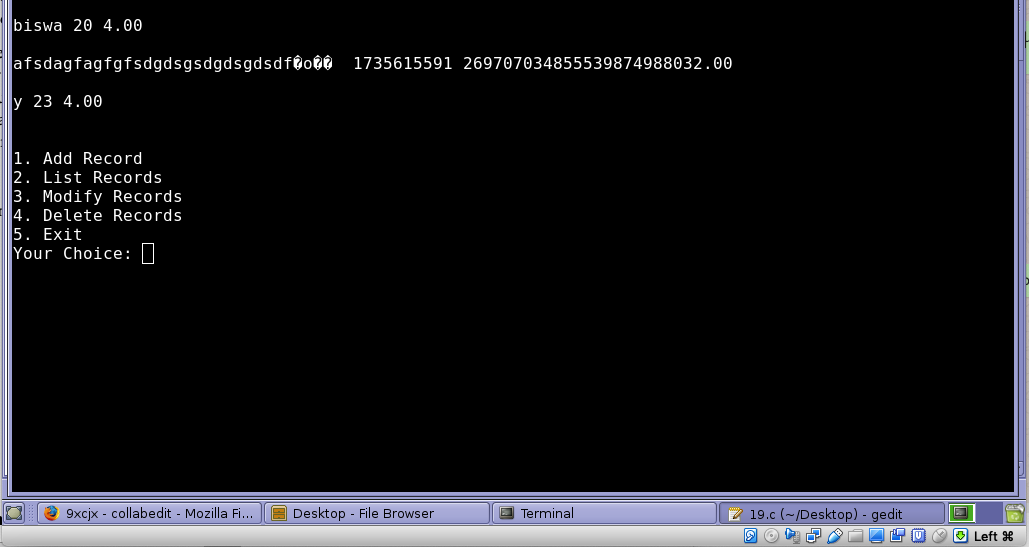
First 7 Programs

**Program1:** 19.c

**Understanding/Description:** student record system where the input is name, age, GPA to perform the functionalities like to add a record, List a record, modify and delete a record.

**Exploit:** From the code we observe that the variable name has been set to take 20 characters, while giving a long name more than 20 characters, the code takes the input for age and GPA input but the buffer overflow occurs for the age and GPA that can be seen in the screenshots.

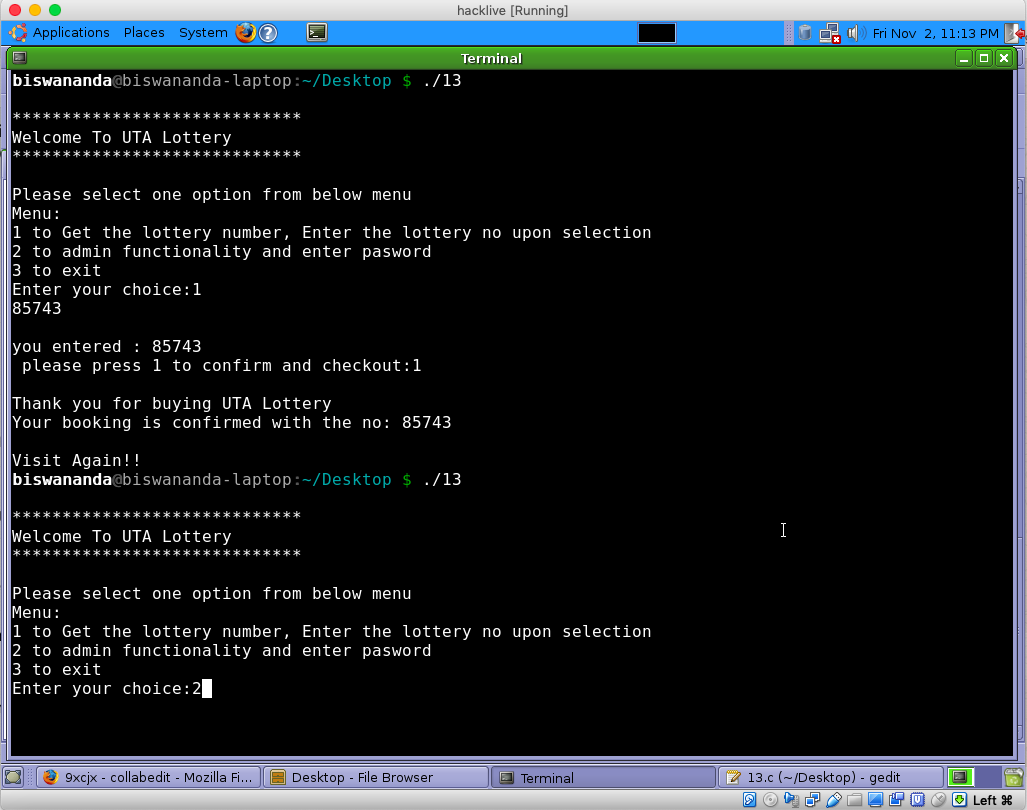


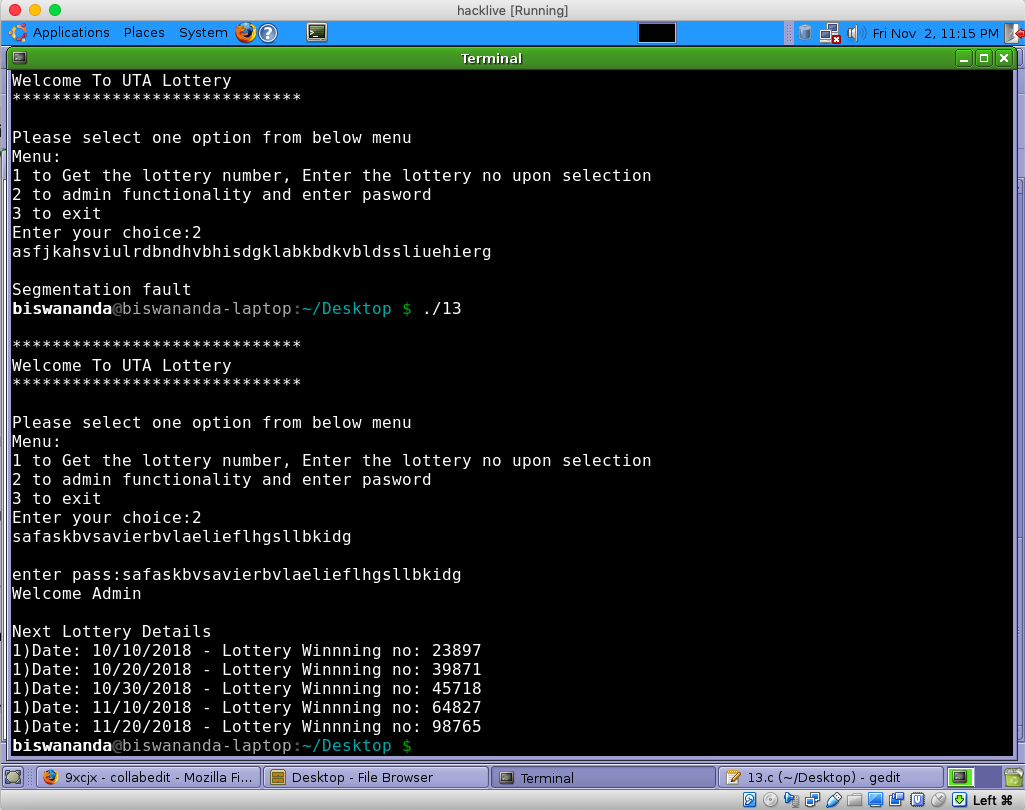


**Program2: 13.c**

**Understanding/Description:** UTA lottery System. The program takes 5-digit lottery number for a purchase option, and another option for admin functionality to check the winning lottery ticket. In the program we can see the correct password for the admin user is admin

**Exploit:** As the buffer size in the program to pick the password input is set to 20, while trying to provide a huge password more than 20 characters for the admin user, the buffer is overflowed, and we can log in successfully to check the winning lottery tickets. Segmentation fault can also be seen, while providing a password size of 47 chars. The Screenshots for the exploitation can be found below.

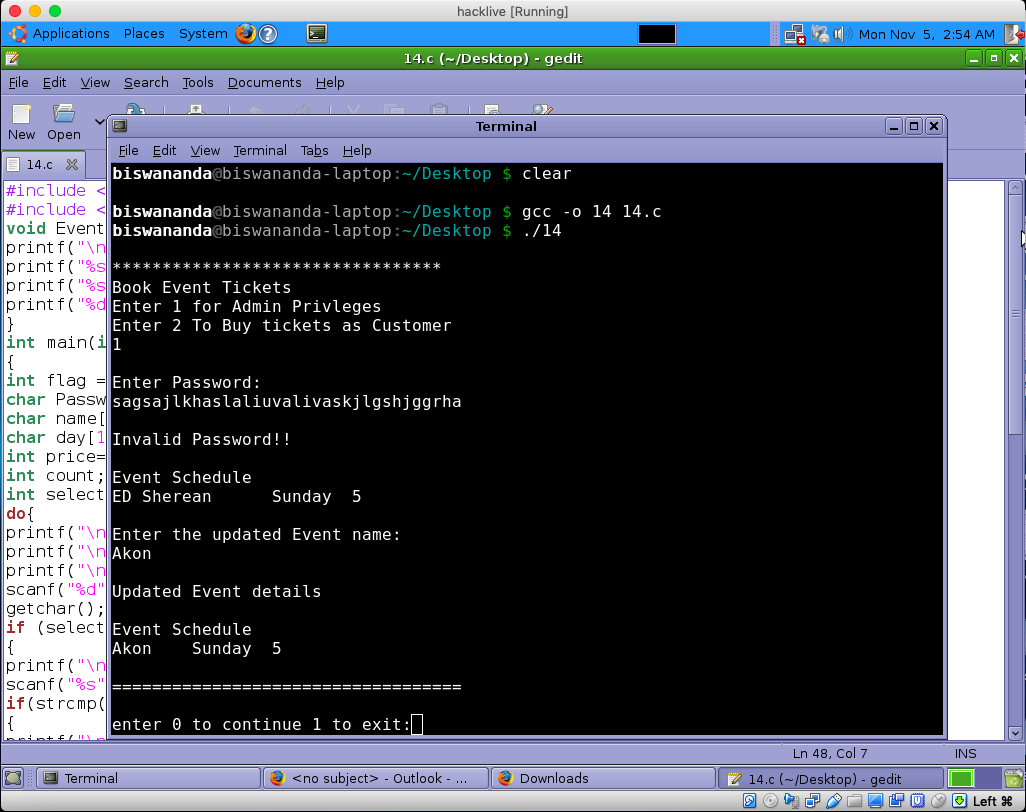




**Program3: 14.c**

**Understanding/Description:** Event Ticket booking program with 2 options i.e. admin privileges create events and buy tickets as customer.

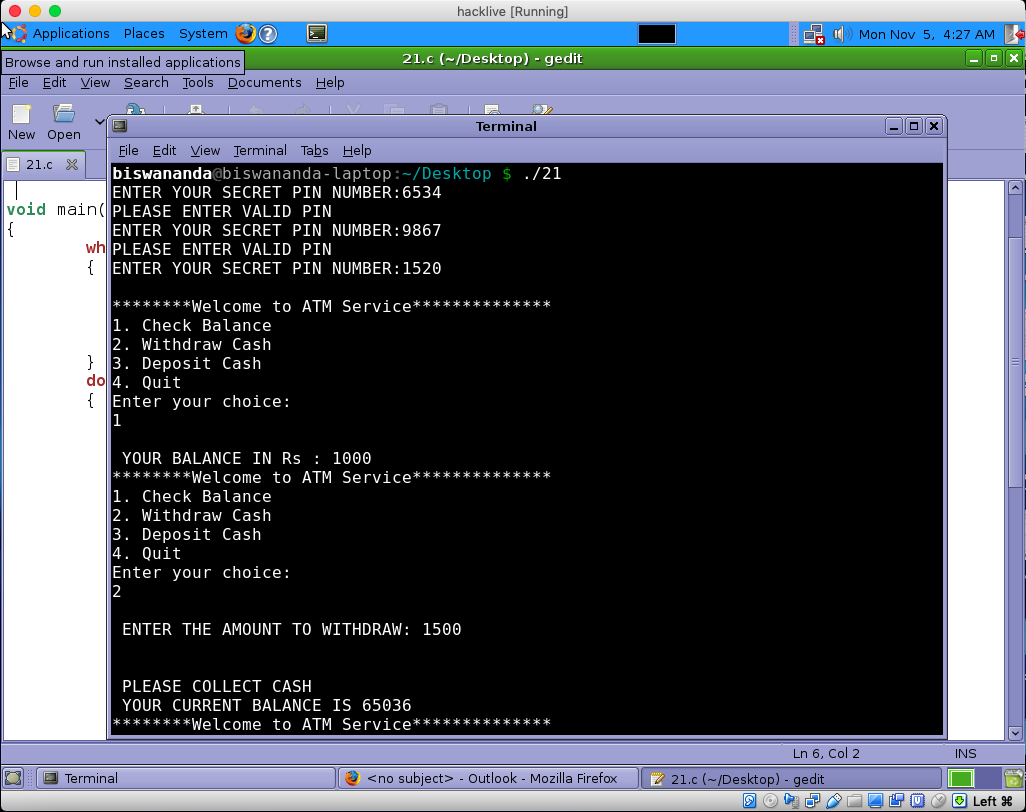
**Exploitation:** Since the variable password is set to 10 chars, while giving a huge password around 25 to 30 characters, implies to buffer overflow, and we can access the admin privileges. The screen shot for the same is provided below.

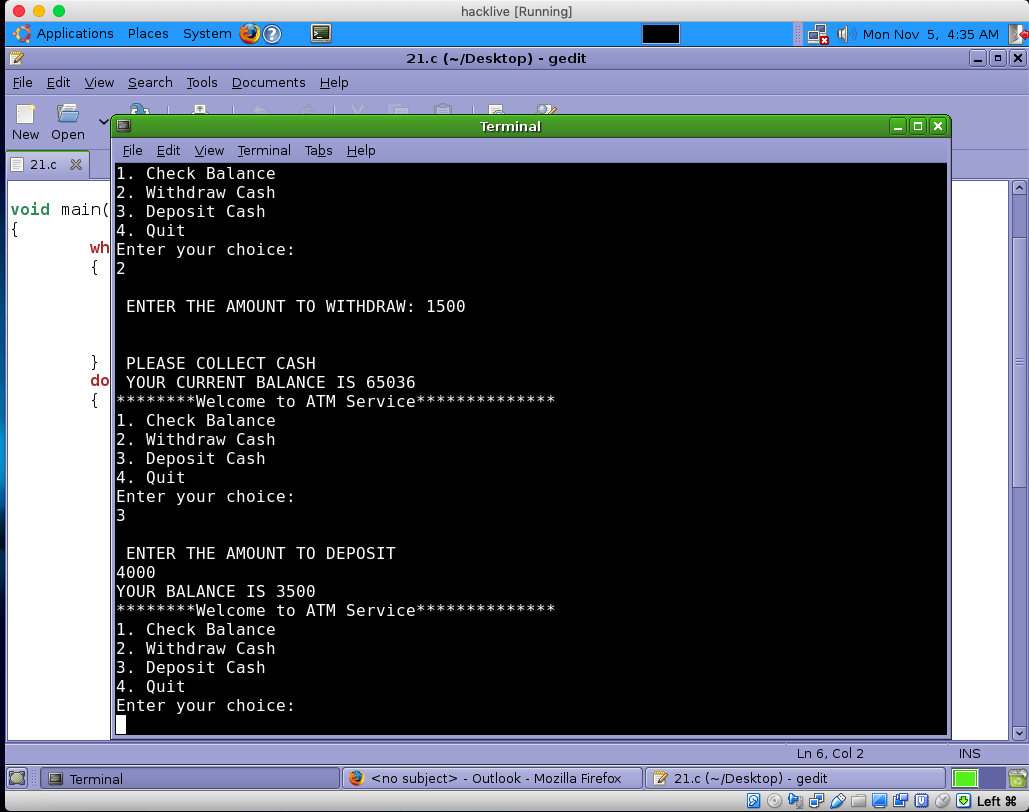


**Program4: 21.c**

**Understanding/Description:** ATM Service program with 3 options i.e. Check balance, Withdraw money, Deposit Money. The Pin is preset to 1520. So, while putting different pin didn’t work

**Exploitation:** it can be seen from the program the use of unsigned short for (amount=1000, deposit, withdraw) is not a good idea, because of which it is vulnerable to the integer overflow. So, while trying to withdraw more money from the available pre-balance as 1000 we can see the current balance increases to 65036. Also, after choosing the deposit option and depositing 4000, the balance shows as 2500 because of the overflow. Screenshot mentioned below:

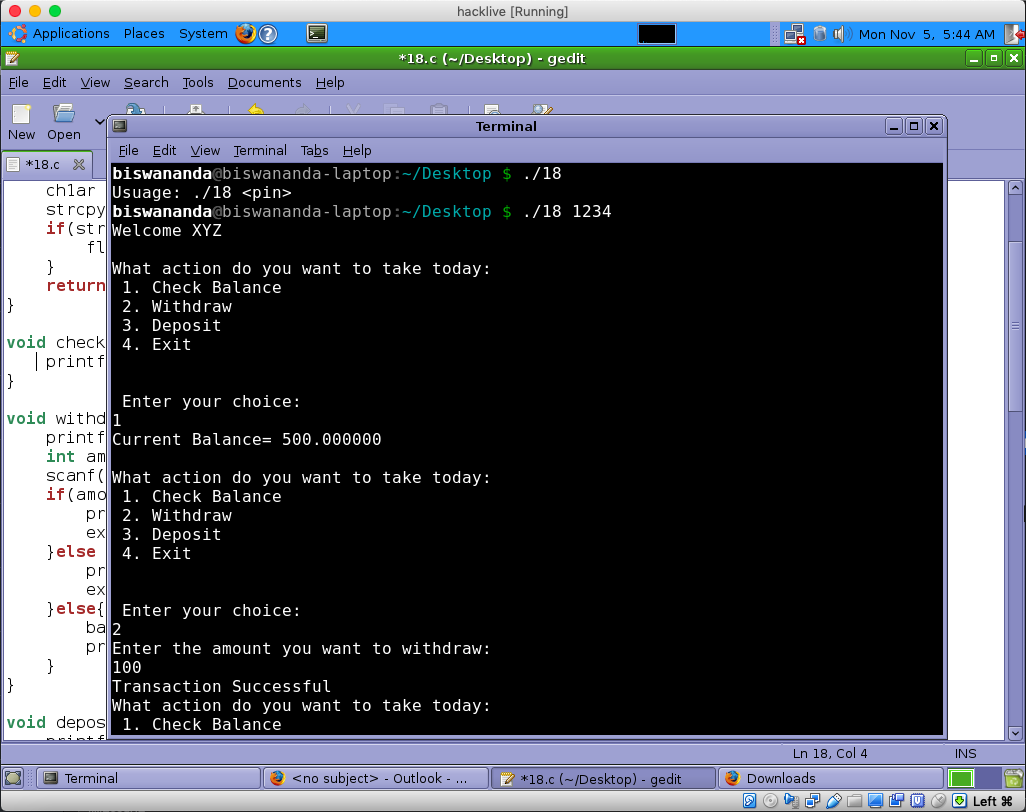


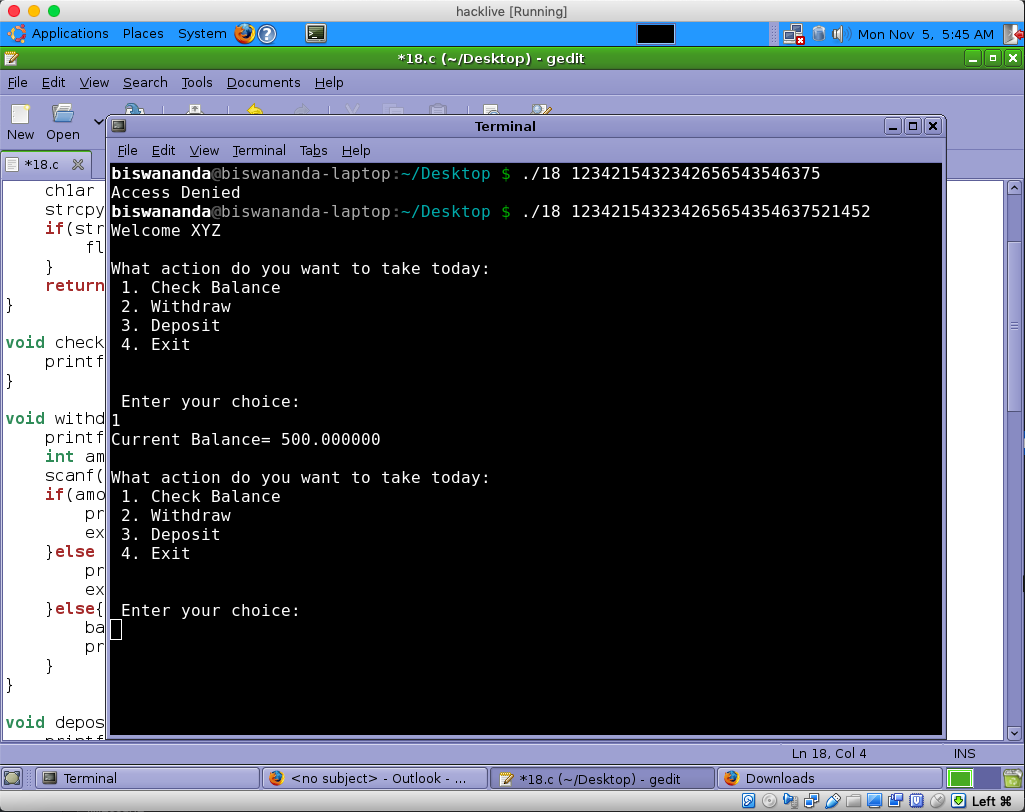


**Program5: 18.c**

**Understanding/Description:** ATM Service program with 3 options i.e. Check balance, Withdraw money, Deposit Money. The Pin is preset to 1234. So, while putting different pin didn’t work

**Exploitation:** it can be seen from the program the use of char pin\_buff [10] because of which it is vulnerable. So, while trying to a long string for pin more than 10, we can see overflow and successfully login to the application. Screenshot mentioned below:

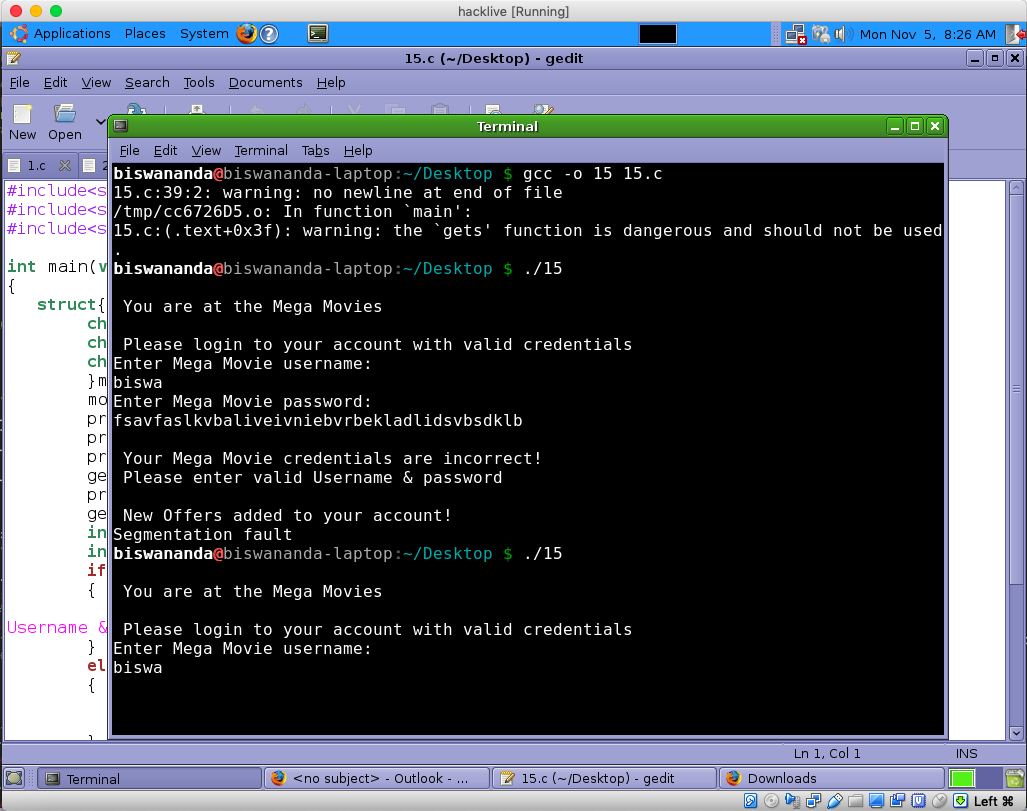




**Program6: 15.c**

**Understanding/ Description:** Mega movie program. Preset username and password are “sanju” and “1993” are already set with a fixed array size of 5.

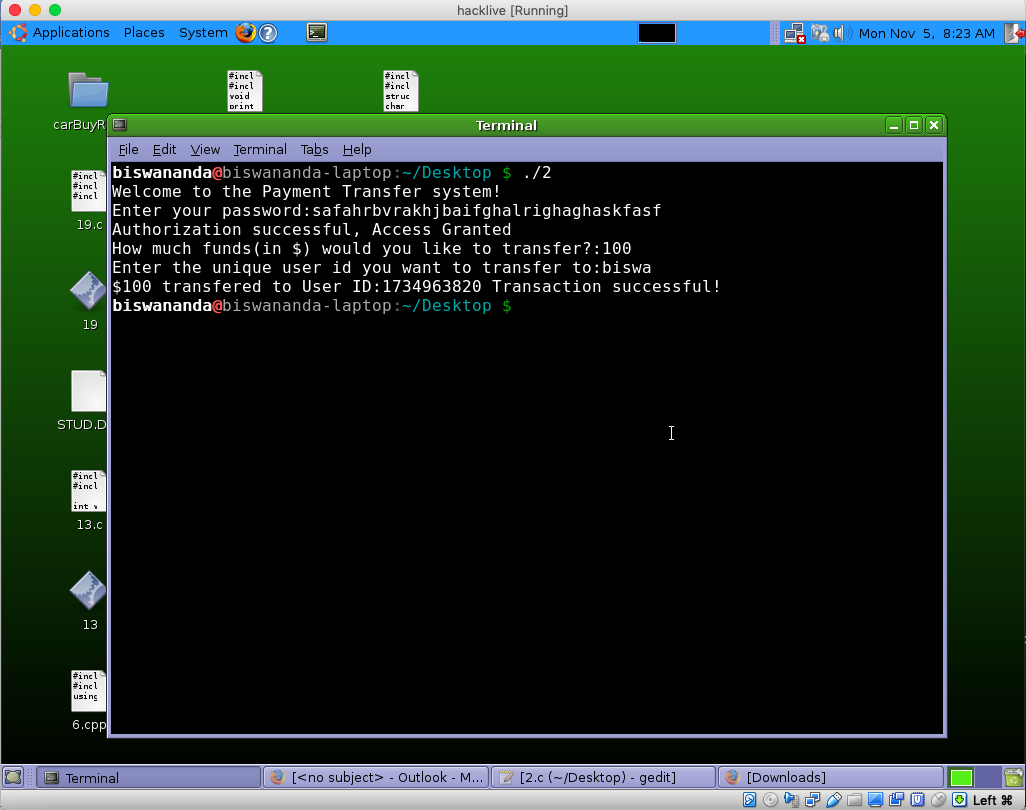
**Exploitation:** If we input a password greater than the allocated size, the output is obtained is segmentation fault cause of the overflow.



**Program7:** 2.c

**Understanding/ Description:** Payment Transfer System. Preset char storedpass[16], is where the vulnerability lies. Use of strcmp is also a bad choice for password comparison. Entering the correct password, you will be able to enter the system.

**Exploitation:** If we input a password greater than the allocated size, we can gain access of the payment system, as shown in the below screenshots.

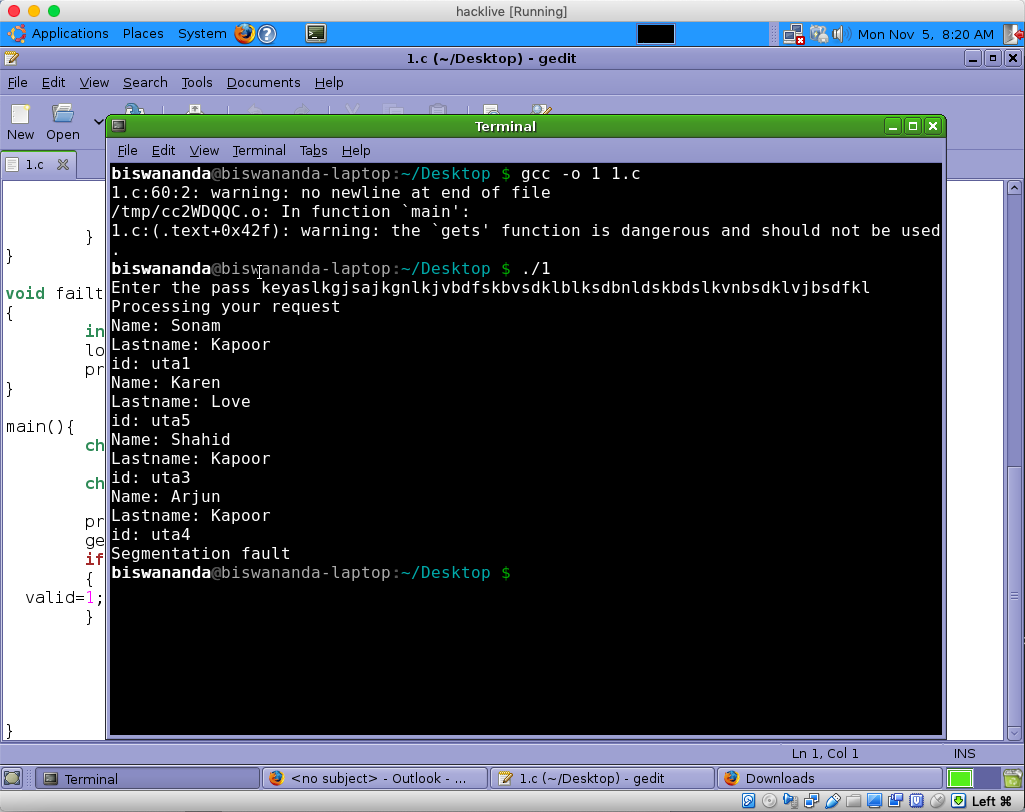


Extra 5 Programs for Bonus

**Program1: 1.c**

**Understanding/ Description:** Employee management system. It displays employee data if correct password is given.

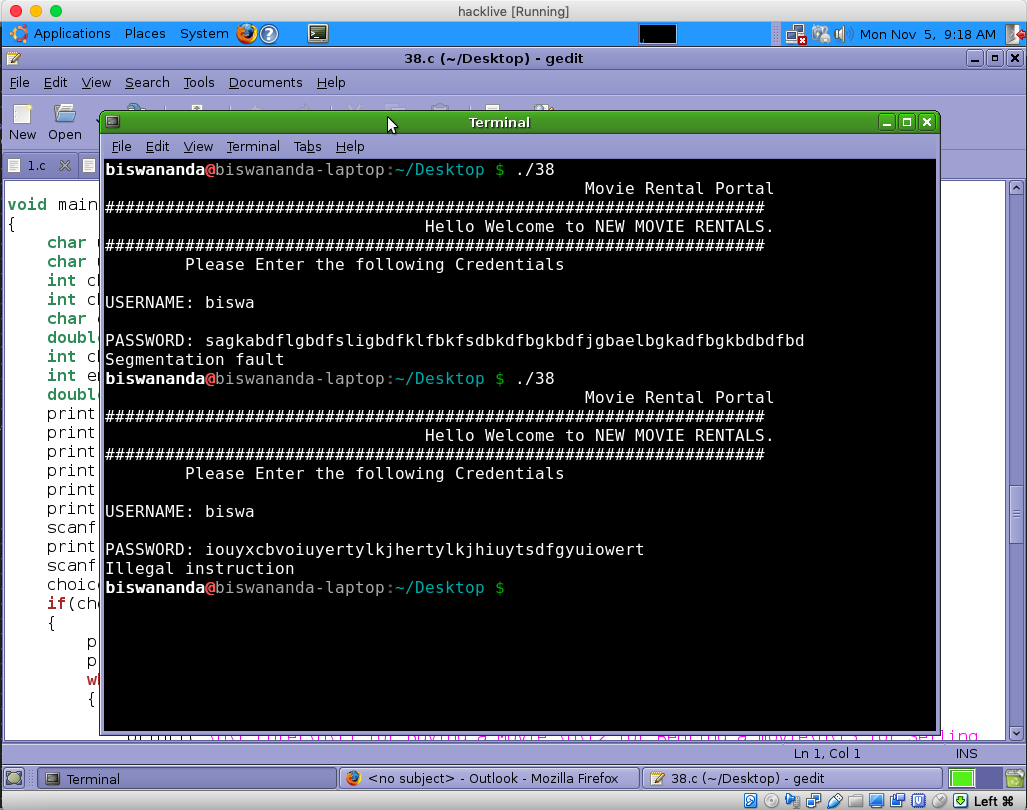
**Exploitation:** If we input a password greater than the length of the buffer, we can gain access of the system, as shown in the below screenshots.



**Program2:** 38.c

**Understanding/ Description:** Movie rental Portal. Vulnerabilities can be seen in the program like: char userI1[16]; char userI2[16]; char choice2[16];

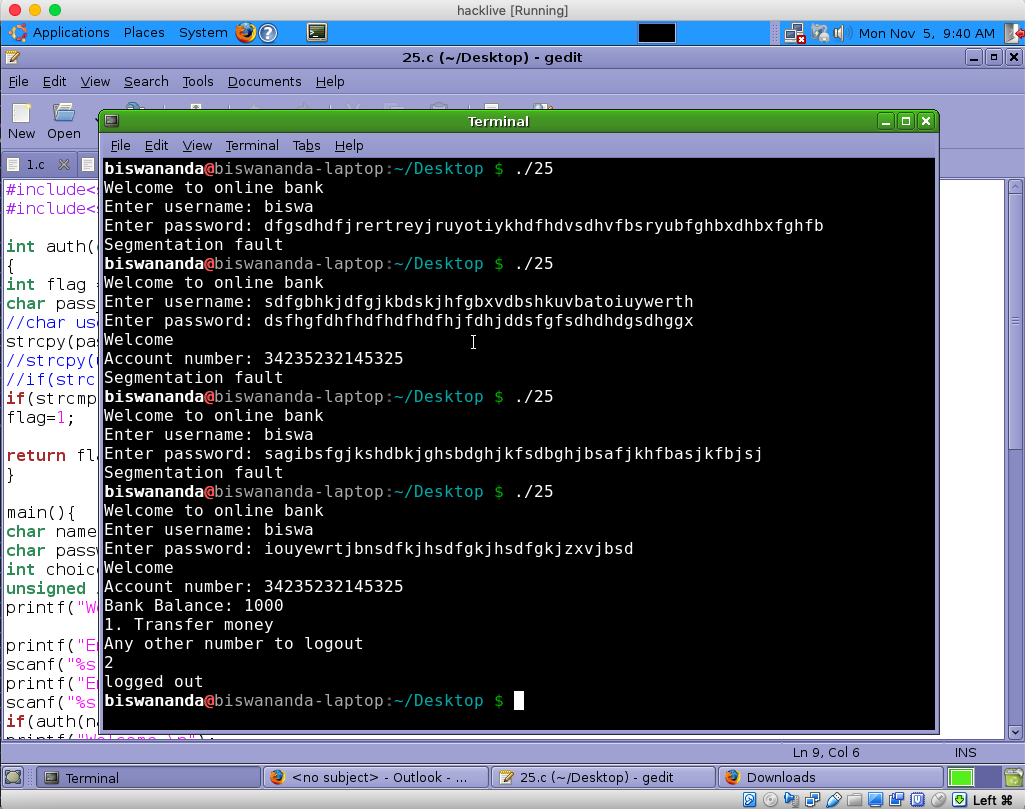
**Exploitation:** If we input a password as 44 we get an illegal instruction, greater than that we get a segmentation fault as shown in the below screenshots.



**Program3: 25.c**

**Understanding/ Description:** Banking System. When the correct credentials, you can get the access to the system to perform balance transfer.

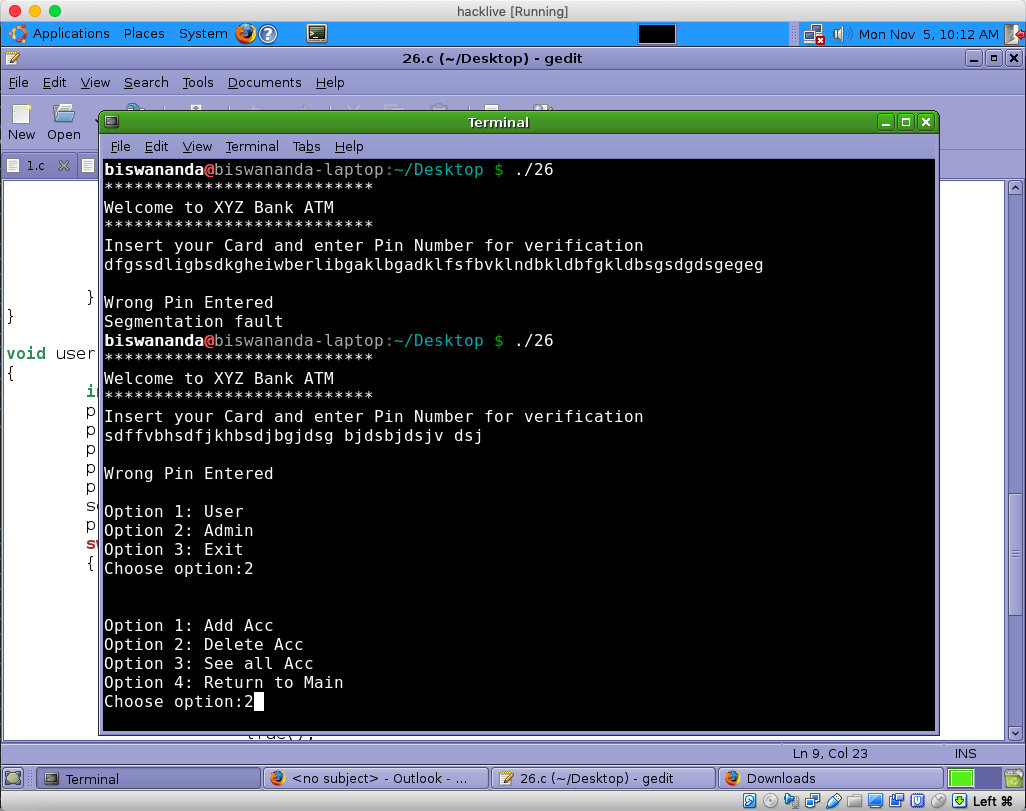
**Exploitation:** simply smashing the stack with the password with the long password that will overwrite stack. We can see the segmentation fault and eventually you will see the buffer overflow and segmentation fault.



**Program4: 26.c**

**Understanding/ Description:** Banking System. When the correct credentials, you can get the access to the system to check balance, withdraw, deposit money.

**Exploitation:** Entering long password will result in segmentation fault and also clear prompts for system activity.

****

**Program5: 8.c**

**Understanding/ Description:** ATM System. When the correct credentials, you can get the access to the system to check balance, withdraw, deposit money.

**Exploitation:** Entering long password will result stack smash.

