Homework-6

Francesco Penasa

January 24, 2020

1. **Exploit Buffer Overflow vulnerability to login using an arbitrary password.**

   In order to login using an arbitrary password we overwrite the variable *secret* using the buffer overflow vulnerability. We write on the variable *password* 20 characters, the first ten characterd will actually be stored in the password variable and the other ten will be stored in the previous variable in the code. For instance if we answer to the question
   Please enter the password: with **password12password34**
   the variables used in the program will be

   ```
   char secret[10] = "password34";
   // char version[3] = "1.0";
   char password[10] = "password12";
   ```

   Thanks to this vulnerability we are able to overwrite the variable *secret* with an arbitrary string and make the condition *(strncmp(password, secret, 10) == 0)* true.
   In the case where other variables are declared in between *password* and *secret* we can still exploit this vulnerability by inserting some more random characters to overwrite the middle variables. For instance if we answer to the question
   Please enter the password: with **password12nulpassword12**
   the variables used for the login will be

   ```
   char secret[10] = "password12";
   char version[3] = "nul";
   char password[10] = "password12";
   ```

   and we will be able to exploit the vulnerability successfully.

2. **Fix Buffer Overflow vulnerability.**
   To fix such vulnerability we could do a check on the length of the input string, to enable such check we can use the function *fgets(...)* that limit the number of characters that can be specified in input instead of the function *gets(...)* that does not perform such check.
   So we simply substitute the function call *gets(password);* with *fgets(password, 10, stdin);*

   In the next page the screeshot of the mitigated C code is displayed.

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{

 char secret[10];
 char version[3] = "1.0";
 char password[10];

 memset(secret, 0, 10);

 printf("%x\n", &secret);
 printf("%x\n", &password);

// printf("%s\n", secret);
// printf("%s\n", password);

 strcpy(secret, "pass");

 printf("Please enter the password: \n");

 fgets(password, 10, stdin);

 if (strncmp(password, secret, 10) == 0) {
  printf("\nCorrect password!\n");
 } else {
  printf("\nWrong password!\n");
 }

 printf("%s\n", secret);
 printf("%s\n", password);


 return 0;
}

```