《漏洞利用及渗透测试基础》实验报告

姓名：平世龙　学号：2012656　班级：1074

实验名称：

格式化字符串漏洞实验

实验要求：

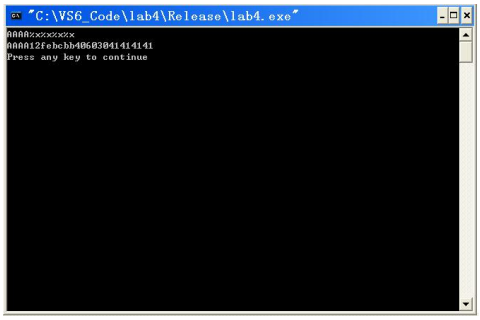
以第四章示例 4-7 代码，完成任意地址的数据获取，观察 Release 模式和 Debug 模式的差异，并进行总结。

实验过程：

1. 进入 VC 反汇编，编写代码如下：

```
#include <stdio.h>
int main(int argc,char *argv[])
{
    char str[200];
    fgets(str,200,stdin);
    printf(str);
    return 0;
}
```

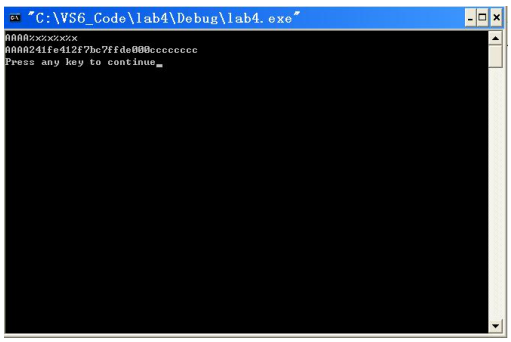
2. 在 Release 模式下编译运行并输入 AAAA%x%x%x%x 得到结果如下：



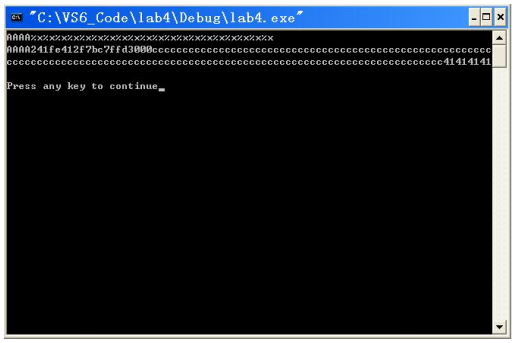
我们成功读到了 AAAA12febcbb40603041414141（0x41 为 ASCII 的字母 A 的值）。

为什么会得到 41414141 值呢？考虑栈帧状态，参数入栈后，通过%x 依次读取参数下面的内存数据时最终会读到局部变量 str 的数据。

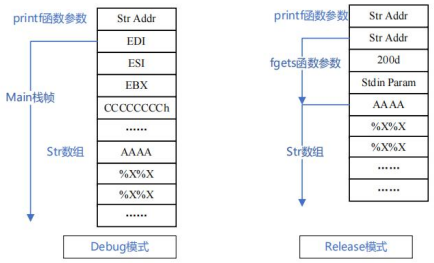
3. 而我们在 Debug 模式下编译运行并输入 AAAA%x%x%x%x 得到一下结果：



我们只有输入 AAAA%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x 才能得到以下结果：



在 Debug 模式下，因为开辟了足够大的栈帧并初始化，char str[200]是从靠近 EBP 的地址分配空间，如果要读到 str 的地址，需要很多的格式化字符；但是在 Release 模式下，并没有严格按照制式的栈帧分配，而是考虑运行性能，在执行到 printf(str)的时候，栈区自顶到底部分为存着“printf 函数参数|fgets 函数参数|str 数组”的内容，在 Main 函数的 retn 语句前，才有一个 add esp XX 的处理。



总结：Debug 模式与 Release 模式的差异为：Debug 模式下，增加了调试信息的输出，开辟了足够大的栈帧并初始化；而 Release 模式并没有严格按照制式的栈帧分配，而是考虑运行性能。

心得体会：

通过实验，熟悉了格式化字符串漏洞的相关内容；
通过实验，了解了格式化字符串漏洞及普通的栈溢出的相似与不同之处；
通过实验，了解了 VC6 的 Release 模式与 Debug 模式的不同。