漏洞解析之 3：使用释放了的缓冲区

1 概述

如果当前使用的缓冲区是一个释放后的缓冲区，那么就会发生“使用释放了的缓冲区”。

2 原因

- 错误的条件或者其他异常情况
- 混淆了负责释放内存代码

3 后果

- 系统崩溃
- 内存被修改
- 执行任何代码或指令

4 示例

- 示例 1

```
void CWE416_Use_After_Free__malloc_free_char_01_bad()
{
    char * data;
    /* Initialize data */
    data = NULL;
    data = (char *)malloc(100*sizeof(char));
    if (data == NULL) {exit(-1);}
    memset(data, 'A', 100-1);
    data[100-1] = '\0';
    /* POTENTIAL FLAW: Free data in the source - the bad sink attempts to
use data */
    free(data);
    /* POTENTIAL FLAW: Use of data that may have been freed */
    printLine(data);
    /* POTENTIAL INCIDENTAL - Possible memory leak here if data was not
freed */
}
```

- 示例 2

```
void CWE416_Use_After_Free__malloc_free_int_01_bad()
{
    int * data;
    /* Initialize data */
    data = NULL;
    data = (int *)malloc(100*sizeof(int));
    if (data == NULL) {exit(-1);}
    {
        size_t i;
```

```
            for(i = 0; i < 100; i++)
            {
                data[i] = 5;
            }
        }
        /* POTENTIAL FLAW: Free data in the source - the bad sink attempts to
use data */
        free(data);
        /* POTENTIAL FLAW: Use of data that may have been freed */
        printIntLine(data[0]);
        /* POTENTIAL INCIDENTAL - Possible memory leak here if data was not
freed */
}
```

- 示例 3

```
void CWE416_Use_After_Free__malloc_free_struct_03_bad()
{
    twoIntsStruct * data;
    /* Initialize data */
    data = NULL;
    if(5==5)
    {
        data = (twoIntsStruct *)malloc(100*sizeof(twoIntsStruct));
        if (data == NULL) {exit(-1);}
        {
            size_t i;
            for(i = 0; i < 100; i++)
            {
                data[i].intOne = 1;
                data[i].intTwo = 2;
            }
        }
        /* POTENTIAL FLAW: Free data in the source - the bad sink attempts to
use data */
        free(data);
    }
    if(5==5)
    {
        /* POTENTIAL FLAW: Use of data that may have been freed */
        printStructLine(&data[0]);
        /* POTENTIAL INCIDENTAL - Possible memory leak here if data was not
freed */
    }
}
```

5 应对

在每次释放后将指针置为 NULL，释放前检查指针是否为 NULL。