1. **案例分析目标**

熟悉漏洞分析的一般流程，

*（软件异常--动静态调试--异常定位--POC编写）*

掌握基本的辅助工具使用和反汇编调试技巧，

*（OD/Immunity\_Debugger、Windbg、IDA）*

对不同类型漏洞的成因、表征和利用有直观的认识，

进而启发漏洞挖掘的思路。

1. **围绕目标培养的能力/知识技能**
   1. **软件异常分析**
      1. 针对模糊测试或其他手段发现的软件异常，通过动静态分析、调试等方法，分析出异常的根源，定位异常点；
      2. 针对二进制程序，逆向分析调试技术
         1. 工具使用：OD、IDA、WinDbg、Immunity\_Debugger
            1. 熟悉OD、WinDbg进行动态跟踪调试的基本操作、下断点的技巧
            2. 熟悉IDA的交叉引用、重命名、标记等方便代码阅读分析的操作
            3. 熟悉WinDbg进行内核调试的基本方法
         2. 汇编代码的阅读能力，能够通过阅读汇编代码来理解代码的功能
            1. 熟悉汇编语言
            2. 对正向编程的理解，写过或见过类似功能的程序
            3. 对高级语言反汇编后架构的理解，能够从汇编代码看出是什么语言编写的
            4. 学习方法：自己编写一些小程序，反汇编生成的可执行程序，对照阅读汇编代码
      3. Windows API熟悉
         1. 从一个函数代码块中使用的API来推测该函数的功能
   2. **漏洞利用**
      1. 漏洞模式
         1. 熟悉常见安全漏洞的机理，源码和汇编代码级表现形式
         2. 不同安全漏洞在内存中的形态，漏洞利用原理以及利用代码的构造、内存布置
         3. 学习方法：通过案例或自己编写漏洞，进行调试分析，从最底层来理解漏洞
      2. POC编写
         1. Windows API熟悉、汇编语言编程
         2. 具体的编码技术
   3. **真实攻击exploit+shellcode设计编码**
      1. OS保护机制对抗：ASLR、DEP
      2. 免杀
      3. 进程隐藏技术：dll inject、rootkit、bootkit
      4. 无痕触发
      5. “通用”exploit+shellcode编写：API动态定位、Heap Spray、egg hunting、ROP
      6. APT设计
   4. **自动化分析、利用工具的开发**
2. **学习路线建议**
   1. 通过书籍，比如Oday安全漏洞挖掘，来了解安全漏洞挖掘、利用的整个过程、涉及到的一些环节、技术，有个完整的整体的概念（短时间内）
   2. 动手实践一些案例（书籍上的、网站上的）的分析、利用过程，从最底层加深对漏洞利用的整个过程的理解（短时间内）
   3. 尝试从软件异常开始自己分析一些漏洞，同时围绕这个分析过程，来补充自己在某些环节中的不足（较长时间内）
      1. 可以按漏洞类型，集中一段时间分析同一类型漏洞，学习总结该类漏洞的调试方法、下断点技巧、利用方法，悟出一些“规律”
      2. 该过程比较传统，较依赖人的经验能力，过程也比较长，
      3. 对于我们来说，这个过程可以持续的来进行积累，但是应该同时
         1. 分析总结后，尽可能地从“规律”中发现漏洞的一些特征，针对这些特征来编写模糊测试的变异规则，使发现的异常更有可能成为一个安全漏洞
         2. 跟踪研究一些实用小工具、新的技术、新的方法，尽可能地能够缩小人工阅读汇编代码的量，让人能够集中精力阅读分析关键的代码块，或者给出更多的漏洞提示信息，让人有头绪来找到分析的重点和方向

**具体的案例分析：**

**第一阶段：2014.10.13 ~ 2015.1.20 （3个月）**

案例分析为主，尽可能独立完成整个分析过程，入手顺序：

老员工以一个栈溢出的案例进行一次分析过程的讲解，让新人知道整个过程如何开展；

然后按照以下顺序独立开展分析，（参考0day书籍）

10.27 ~ 11.28

stack overflow （新员工）

*（ReturenEIP: Photoshop、SEH: ProShow、Off-by-one: winamp）*

New loopholes （老员工）

*网络组（Heartbleed(WNS)、工控组BangShell(ICSS)）*

ICSS virus （老员工）

*Stuxnet*

12.1 ~ 12.5

交流讨论

12.8 ~ 1.9（新老员工一起2周+1周+1周+1周每周末交流讨论）

heap overflow

*（SSReader、ACDSee)*

Use-afer-free

*（Photoshop）*

DDOS

*（FoxitReader）*

Integer Underflow

*（IE）*

1.10 ~ 1.20 案例分析总结

**第二阶段：2015.2.16 ~ 2015.3.31 （1.5月）**

分析漏洞挖掘的案例、学习漏洞挖掘的方法（静态、模糊测试、污点分析）和工具，结合第一阶段对漏洞的分析，思考和总结一些漏洞挖掘的思路，将漏洞挖掘和漏洞分析整个过程完整起来。

团队成员分工开展单点技术的实践，如API Hook、DLL inject、PE解析、内存映射、进程隐藏、ROP、 Egg hunting、免杀技术、无痕触发方法、开机启动、病毒扩展方法、虚拟机开发机制/技术等。

**第二阶段：2015.2.16 ~ 2015.3.31 （1.5月） -- 漏洞挖掘方向成员**

1. 模糊测试系统Peach fuzzer的学习
2. Intel Pin的学习
3. 轨迹记录模块的深入分析--重点工作在代码阅读、分析、理解、应用--具备重写轨迹记录模块与二次开发的能力
4. 漏洞挖掘的并行化研发