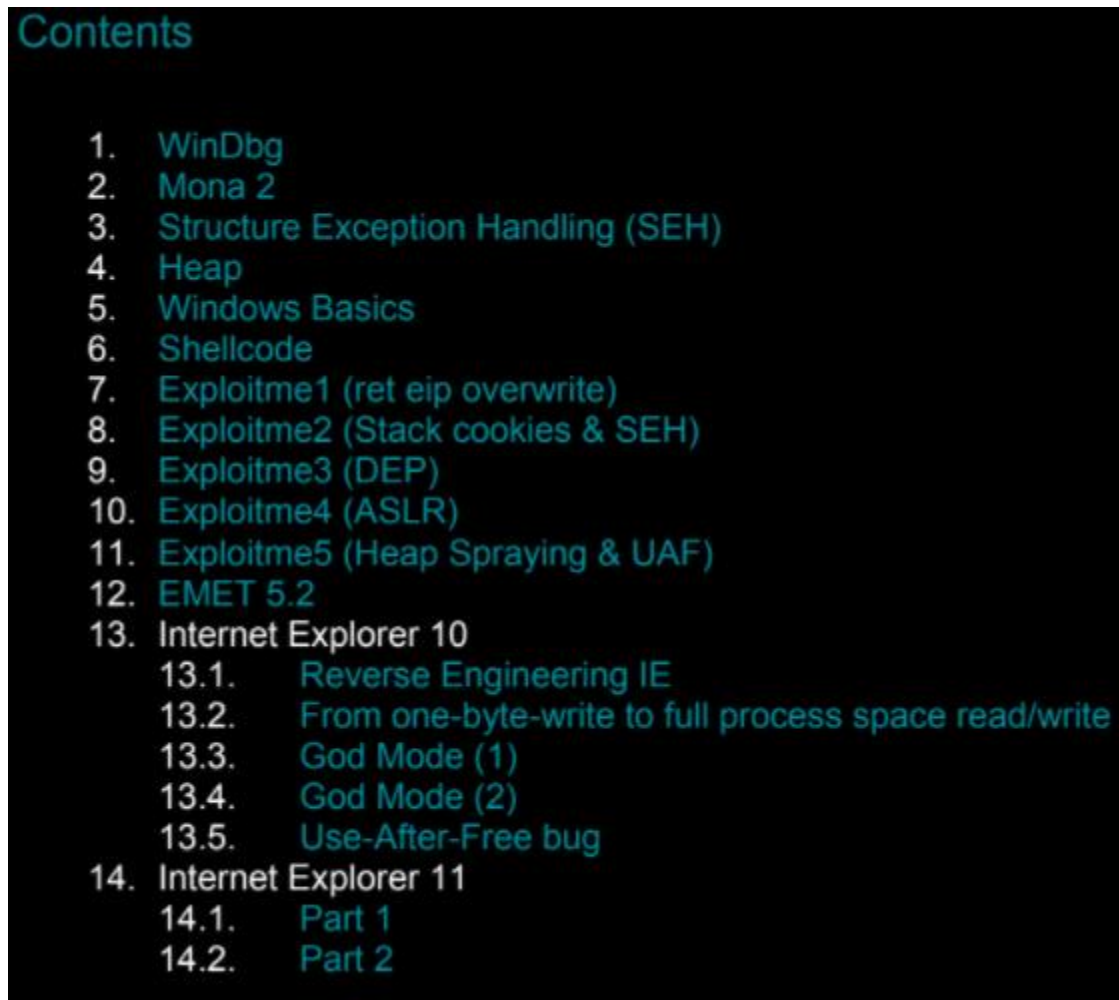


Windows Exploit Development

Books

- Modern Windows Exploit Development

16-week Study plan



| | |
|----------|--|
| Contents | |
| 1. | WinDbg |
| 2. | Mona 2 |
| 3. | Structure Exception Handling (SEH) |
| 4. | Heap |
| 5. | Windows Basics |
| 6. | Shellcode |
| 7. | Exploitme1 (ret eip overwrite) |
| 8. | Exploitme2 (Stack cookies & SEH) |
| 9. | Exploitme3 (DEP) |
| 10. | Exploitme4 (ASLR) |
| 11. | Exploitme5 (Heap Spraying & UAF) |
| 12. | EMET 5.2 |
| 13. | Internet Explorer 10 |
| 13.1. | Reverse Engineering IE |
| 13.2. | From one-byte-write to full process space read/write |
| 13.3. | God Mode (1) |
| 13.4. | God Mode (2) |
| 13.5. | Use-After-Free bug |
| 14. | Internet Explorer 11 |
| 14.1. | Part 1 |
| 14.2. | Part 2 |

Basically, rush through the first 6 chapters in 6 weeks. Then the next 10 weeks are doing the rest of the chapters. I will produce each type of exploit: stack, heap, and ones bypassing DEP&ASLR with detailed procedures to all of them. I can give an oral presentation on how you might go about doing these things at the end of the year. I will work on this course 2 hours per day during the week and have bi-weekly status reports with you. I'll explain: what i've learned, accomplished, and what I could be doing better.

Outcomes

- Proficient debugging skills in WinDBG and mona
- The ability to find and exploit following vulnerabilities: stack overflow, heap overflow, string, integer, and file vulnerabilities while bypassing DEP, ASLR, and SEH
- Use Sonarlint and sonarqube to see if these catch all vulnerabilities
- Strong oral/verbal communication skills in doing the above processes

Week 3- Project 1

- Receive some C/C++ code. Tell the instructor what compiler/linker was used
- Tell him all of the variables
- Write a description of what is going on
- Bypass the authentication by changing the binary
- Bypass the authentication by finding the key

Week 4 - Homework 2

- Fill out chapter 1 in the reverse engineering book

Week 6 - Project 2

- Find the buffer overflow
- Exploit the buffer overflow by popping up the calculator

Week 8 - Homework 3

- Fill out chapter 2 in the reverse engineering book

Week 11 - Project 3

- Exploit a SEH vulnerability

Week 15 - Project 4

- Exploit a heap based buffer overflow that pypasses ASLR