
Exploiting and Defense

Dobin Rutishauser

2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023

About Me

Dobin Rutishauser

- Penetration Tester at Compass
- SOC Analyst at Infoguard
- RedTeam at Raiffeisen

Interested in ~~Hacking~~ Security since a young age (1998+)

I got a bit overboard when I was young



Content

Content

Exploiting & Defense

We will write **exploits** to **exploit buffer-overflows**

We will analyze what **defenses** exist to make writing exploits harder

Lecture



Lecture - Online

<https://exploit.courses>

- ★ Online exploit development website
- ★ Access to your own Linux via JavaScript terminal
- ★ Solve challenges online
 - ★ Write exploits
 - ★ Debug them
- ★ Slides

Lecture - Online

If you wanna try it by yourself on your own machine (not recommended):

The writeup of the challenges: <https://github.com/dobin/yookiterm-challenges>

Source code of challenges: <https://github.com/dobin/yookiterm-challenges-files>



Important slides are marked with  in top right corner

Sometimes slides have helpful comments in "notes" section

Recap slides at end of chapters point you to which things are important, and should be understood

Motivation

Motivation for Exploiting & Defense

Motivation

For the hacker:

- ✦ Developing exploits
- ✦ Debugging of C/C++ code
- ✦ Disassembly & reversing of assembler code
- ✦ Being 31337

For the Sysadmin

- ✦ Judge security level of operating systems, and applications
- ✦ Harden and protect servers, clients

For the CISO:

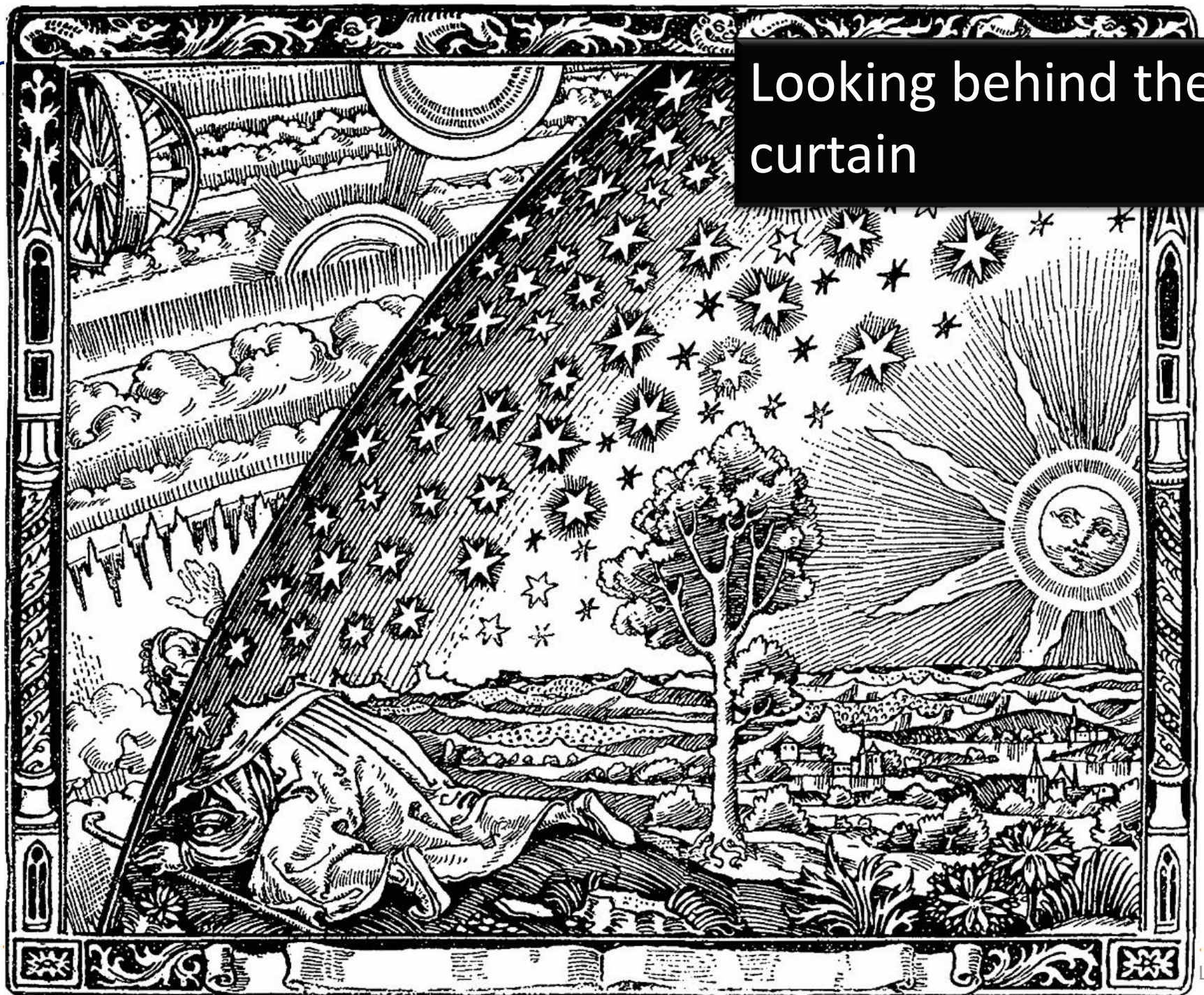
- ✦ Assess CVSS scores
- ✦ Assess (new) security mitigations
- ✦ Better risk analysis

Motivation

For everyone:

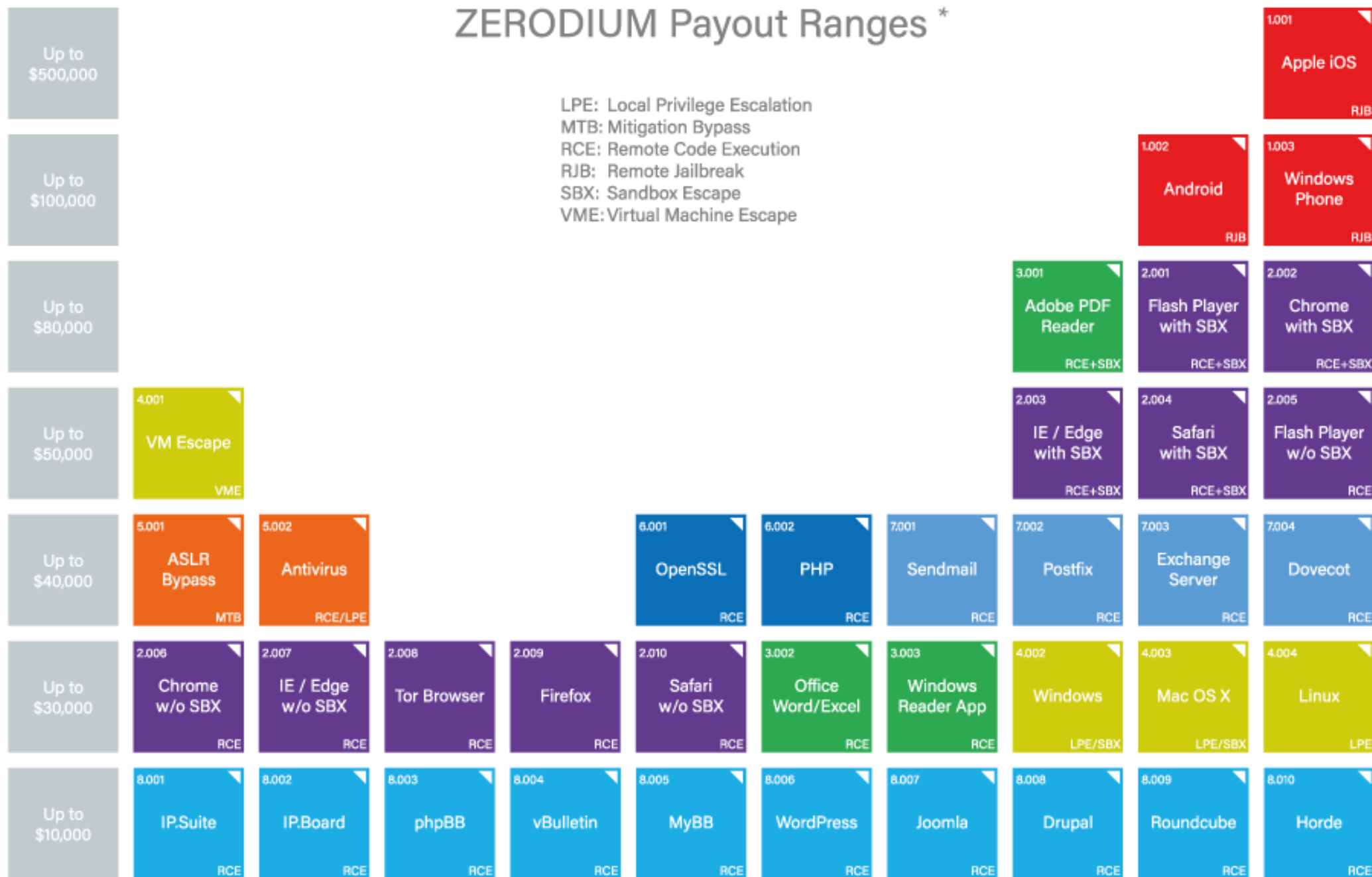
- ✦ How do functions work?
- ✦ How does computer work?!
- ✦ Dance the exploit / mitigation tango
- ✦ Breach abstraction layers to get what you want
- ✦ Get a sense of long term security development

Looking behind the curtain



ZERODIUM Payout Ranges *

LPE: Local Privilege Escalation
MTB: Mitigation Bypass
RCE: Remote Code Execution
RJB: Remote Jailbreak
SBX: Sandbox Escape
VME: Virtual Machine Escape

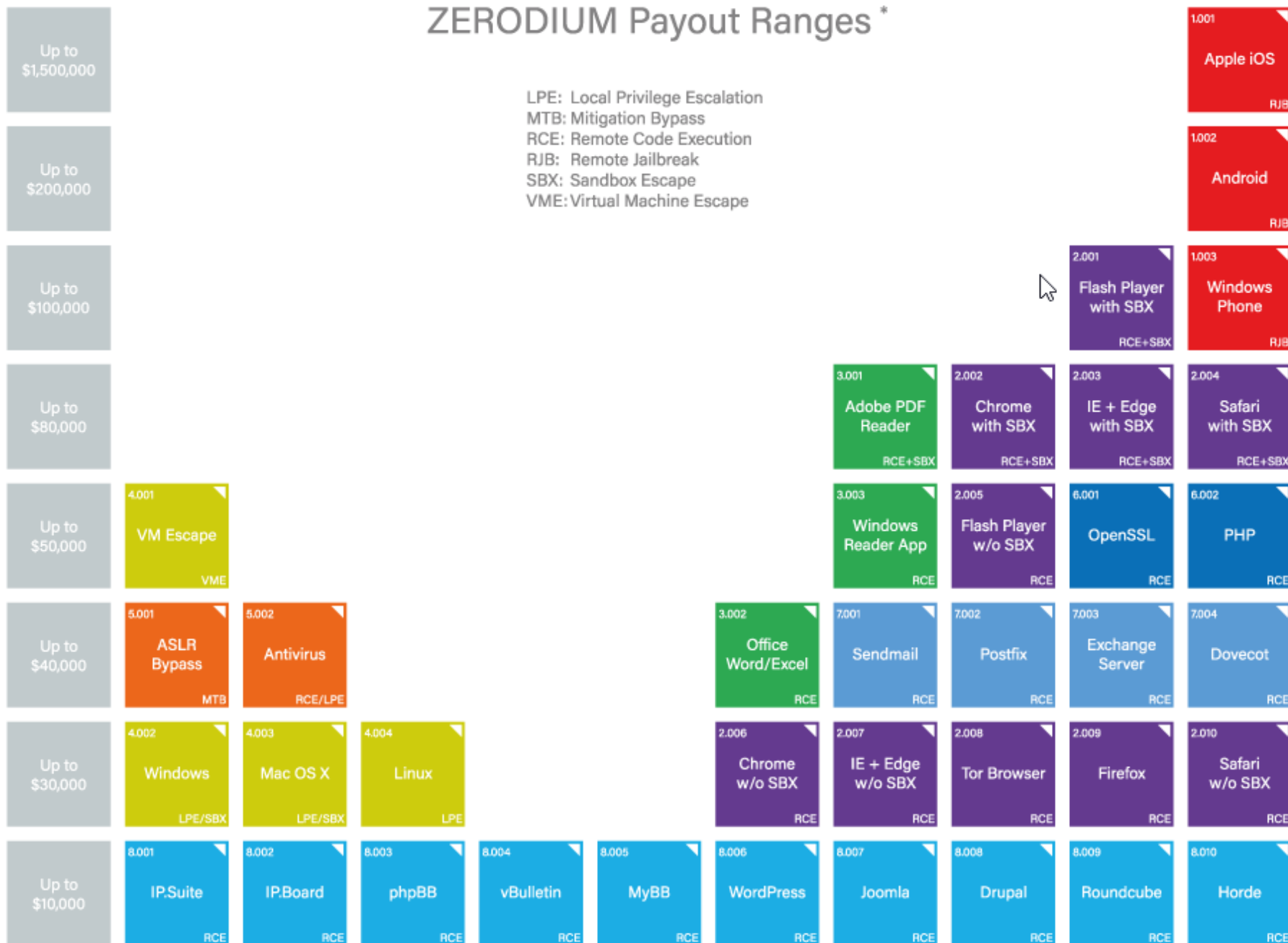


* All payout amounts are chosen at the discretion of ZERODIUM and are subject to change or cancellation without notice.

2016/01 © zerodium.com

ZERODIUM Payout Ranges *

LPE: Local Privilege Escalation
 MTB: Mitigation Bypass
 RCE: Remote Code Execution
 RJB: Remote Jailbreak
 SBX: Sandbox Escape
 VME: Virtual Machine Escape



* All payout amounts are chosen at the discretion of ZERODIUM and are subject to change or cancellation without notice.

ZERODIUM Payouts for Desktops/Servers*

Up to \$1,000,000											1.001 Win RCE Zero Click Win
Up to \$500,000								3.001 Chrome RCE+LPE Win	2.001 Apache RCE Linux	2.002 MS IIS RCE Win	
Up to \$250,000							5.001 MS Outlook RCE Win	4.001 MS Exchange RCE Win	2.003 OpenSSL RCE Linux	2.004 PHP RCE Linux	
Up to \$200,000	6.001 VMware ESXi VME Win/Linux	5.002 Thunderbird RCE Win/Linux			4.002 Sendmail RCE Linux	4.003 Postfix RCE Linux	4.004 Dovecot RCE Linux	4.005 Exim RCE Linux	2.005 nginx RCE Linux		
Up to \$100,000		3.002 Safari RCE+LPE Mac	3.003 Edge RCE+LPE Win	3.004 Firefox RCE+LPE Win	5.003 Word/Excel RCE Win	7.001 WordPress RCE Linux	7.002 cPanel/WHM RCE Linux	7.003 Plesk RCE Linux	7.004 Webmin RCE Linux		
Up to \$80,000	6.002 VMware WS VME Win/Linux					5.004 Adobe PDF RCE+SBX Win	5.005 WinRAR RCE Win	5.006 7-Zip RCE Win	6.003 Windows LPE/SBX Win		
Up to \$50,000	6.004 USB LPE Win/Mac	8.001 Antivirus RCE Win			5.007 WinZip RCE Win	5.008 tar RCE Linux	6.005 macOS LPE/SBX Mac	6.006 Linux LPE Linux	6.007 BSD LPE BSD		
Up to \$10,000	9.001 Routers RCE	8.002 Antivirus LPE Win	7.005 phpBB RCE Linux	7.006 vBulletin RCE Linux	7.007 MyBB RCE Linux	7.008 Joomla RCE Linux	7.009 Drupal RCE Linux	7.010 Roundcube RCE Linux	7.011 Horde RCE Linux		

Windows
 macOS
 Linux/BSD
 Any OS

RCE: Remote Code Execution
 LPE: Local Privilege Escalation
 SBX: Sandbox Escape or Bypass
 VME: Virtual Machine Escape

* All payouts are subject to change or cancellation without notice. All trademarks are the property of their respective owners.

2019/01 © zerodium.com

Slide 19

ZERODIUM Payouts for Mobiles*

RJB: Remote Jailbreak with Persistence
RCE: Remote Code Execution
LPE: Local Privilege Escalation
SBX: Sandbox Escape or Bypass

■ iOS
■ Android
■ Any OS

Up to
\$2,000,000

1.001

iPhone RJB
Zero Click

iOS

Up to
\$1,500,000

1.002

iPhone RJB

iOS

Up to
\$1,000,000

2.001

WhatsApp
RCE+LPE

iOS / Android

2.002

SMS/MMS
RCE+LPE

iOS / Android

2.003

iMessage
RCE+LPE

iOS

Up to
\$500,000

2.004

WeChat
RCE+LPE

iOS / Android

2.005

FB Messenger
RCE+LPE

iOS / Android

2.006

Signal
RCE+LPE

iOS / Android

2.007

Telegram
RCE+LPE

iOS / Android

2.008

Email App
RCE+LPE

iOS / Android

3.001

Chrome
RCE+LPE

Android

3.002

Safari
RCE+LPE

iOS

Up to
\$200,000

4.001

Baseband
RCE+LPE

iOS / Android

5.001

LPE to
Kernel / Root

iOS / Android

2.009

Media Files
RCE+LPE

iOS / Android

2.010

Documents
RCE+LPE

iOS / Android

3.003

SBX
for Chrome

Android

3.004

Chrome RCE
w/o SBX

Android

3.005

SBX
for Safari

iOS

3.006

Safari RCE
w/o SBX

iOS

Up to
\$100,000

6.001

Code Signing
Bypass

iOS / Android

4.002

WiFi
RCE

iOS / Android

4.003

RCE
via MitM

iOS / Android

5.002

LPE to
System

Android

7.001

Information
Disclosure

iOS / Android

7.002

[k]ASLR
Bypass

iOS / Android

8.001

PIN
Bypass

Android

8.002

Passcode
Bypass

iOS

8.003

Touch ID
Bypass

iOS

* All payouts are subject to change or cancellation without notice. All trademarks are the property of their respective owners.

2019/01 © zerodium.com

Slide 20

Linux Vulnerabilities in 2022

CVE-2021-4034: Linux Polkit

- <https://blog.qualys.com/vulnerabilities-threat-research/2022/01/25/pwnkit-local-privilege-escalation-vulnerability-discovered-in-polkits-pkexec-cve-2021-4034>

CVE-2021-44142: Samba

- <https://blog.malwarebytes.com/exploits-and-vulnerabilities/2022/02/samba-patches-critical-vulnerability-that-allows-remote-code-execution-as-root/>

CVE-2022-0185: Linux Kernel

- <https://jfrog.com/blog/the-impact-of-cve-2022-0185-linux-kernel-vulnerability-on-popular-kubernetes-engines/>

What is the impact of these vulns? Risk? Whats the vulnerability? Whats the mitigation?

Content of the next 7 Friday afternoons

You want to learn:

- ✦ What memory corruptions are
- ✦ What buffer overflows are
- ✦ What exploits are
- ✦ How exploits are being created
- ✦ To exploit a local application
- ✦ To exploit a remote application
- ✦ Learn about anti-exploiting technologies
- ✦ To circumvent all common anti-exploiting technologies for Linux
- ✦ See how Windows does it
- ✦ Use Use-After-Free Heap overflows
- ✦ See next generation attacks and defenses
- ✦ ~~Hack facebook “for a friend”~~

What you first learn:

- ★ Intel x86
 - ★ Architecture
 - ★ CPU
 - ★ Registers
- ★ Linux
 - ★ Userspace memory layout, stacks, heap
 - ★ Syscalls
 - ★ Sockets
- ★ Programming Languages
 - ★ Assembler
 - ★ C
 - ★ Python
 - ★ Bash

Plan

Plan

21.04.2023

Theory:

- ✦ 0x01 Intro (this)
- ✦ 0x02 Intro Technical
- ✦ 0x10 Intel Architecture
- ✦ 0x11 Memory Layout

Challenges:

- ✦ 0: Introduction to memory layout - basic
- ✦ 1: Introduction to memory layout - advanced

Plan

28.04.2023

Theory:

- ✦ 0x12 C Array and Data Structures
- ✦ 0x30 Assembler Intro
- ✦ 0x31 Shellcode
- ✦ 0x32 Function Call Convention
- ✦ 0x33 Debugging

Challenges:

- ✦ 2: C buffer analysis - simple
- ✦ 3: Introduction to shellcode development
- ✦ 7: Function Call Convention in x86 (32bit)
- ✦ 8: C buffer analysis - with debugging
- ✦ 9: Simple Buffer overflow - variable overwrite

Plan

05.05.2023

Theory:

- ✦ 0x40 Arrays
- ✦ 0x41 Buffer Overflow
- ✦ 0x42 Exploit
- ✦ 0x44 Remote Exploit

Challenges:

- ✦ 11: Development of a buffer overflow exploit - 32 bit
- ✦ 12: Development of a buffer overflow exploit - 64 bit
- ✦ 13: Development of a remote buffer overflow exploit - 64 bit

Plan

12.05.2023

Theory:

- ✦ 0x51 Exploit Mitigation
- ✦ 0x52 Defeat Exploit Mitigation
- ✦ 0x70 Secure Coding

Challenges:

- ✦ 14: Stack canary brute force
- ✦ 15: Simple remote buffer overflow exploit - ASLR/DEP/64bit
- ✦ 16: Remote buffer overflow with ROP - DEP/64bit
- ✦ 17: Remote buffer overflow with ROP - DEP/ASLR/64bit

Plan

19.05.2023

- Nüt

Plan

26.05.2023

Theory:

- ✦ 0x52: Defeat Exploit Mitigations
- ✦ 0x56: Defeat Exploit Mitigations – PIE
- ✦ stuff

Challenges:

- ✦ 31: Heap use-after-free analysis

Plan

02.06.2023

Theory:

- ✦ 0x54: Defeat Exploit Mitigations ROP
- ✦ 0x60: Windows Exploiting
- ✦ 0x74: Hardware Hacking
- ✦ 0xA0: Browser Security
- ✦ stuff

Plan

09.06.2023

Theory:

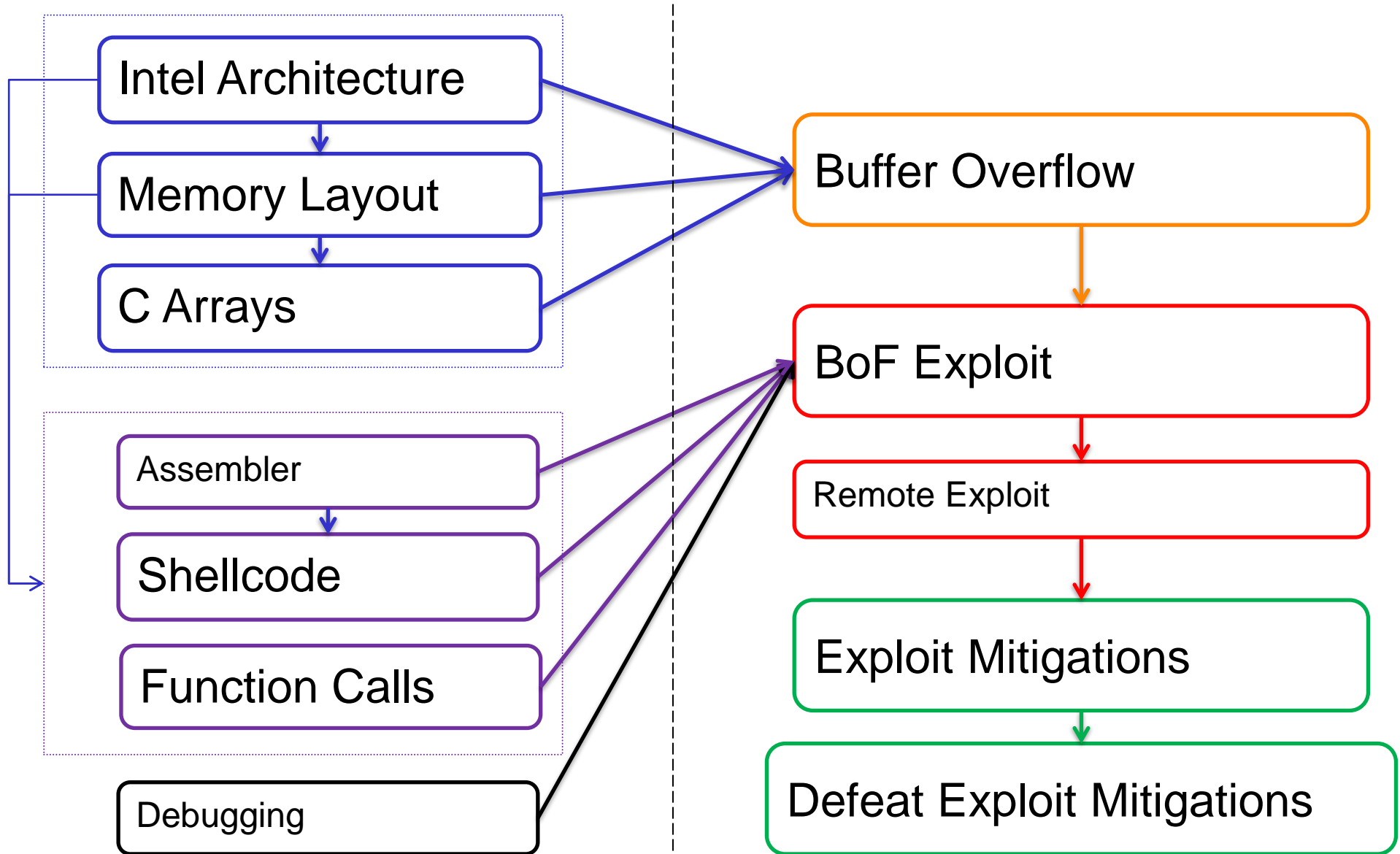
- ✦ 0x71: Fuzzing
- ✦ 0x75: CFI

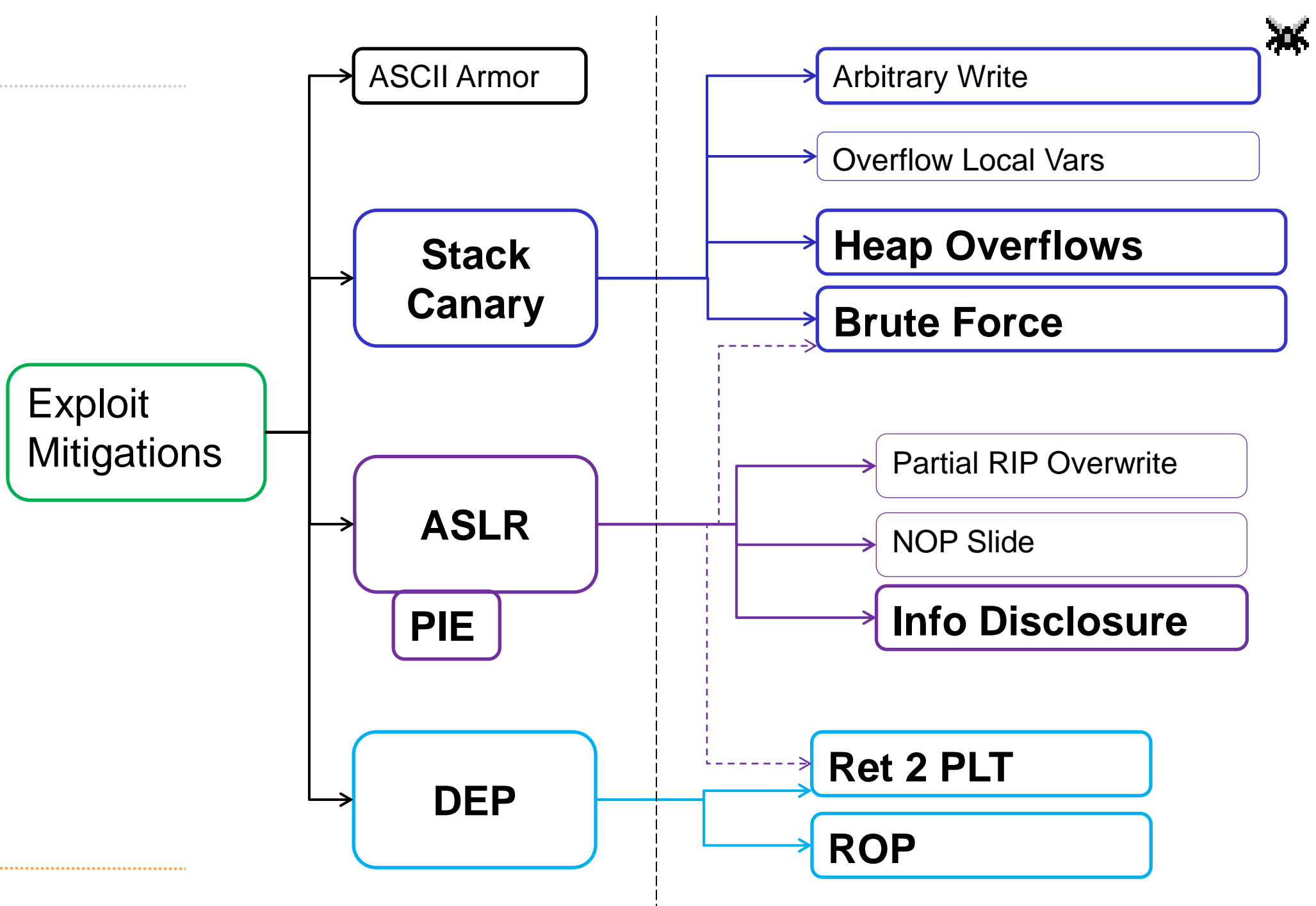
Challenges:

- ✦ 60: Linux Hardening



Content





And some...

Windows Exploiting

Fuzzing

Browser Security

Kernel Exploits

Secure Coding

Linux Hardening

Case Studies



What is (mainly) relevant for the oral exam:

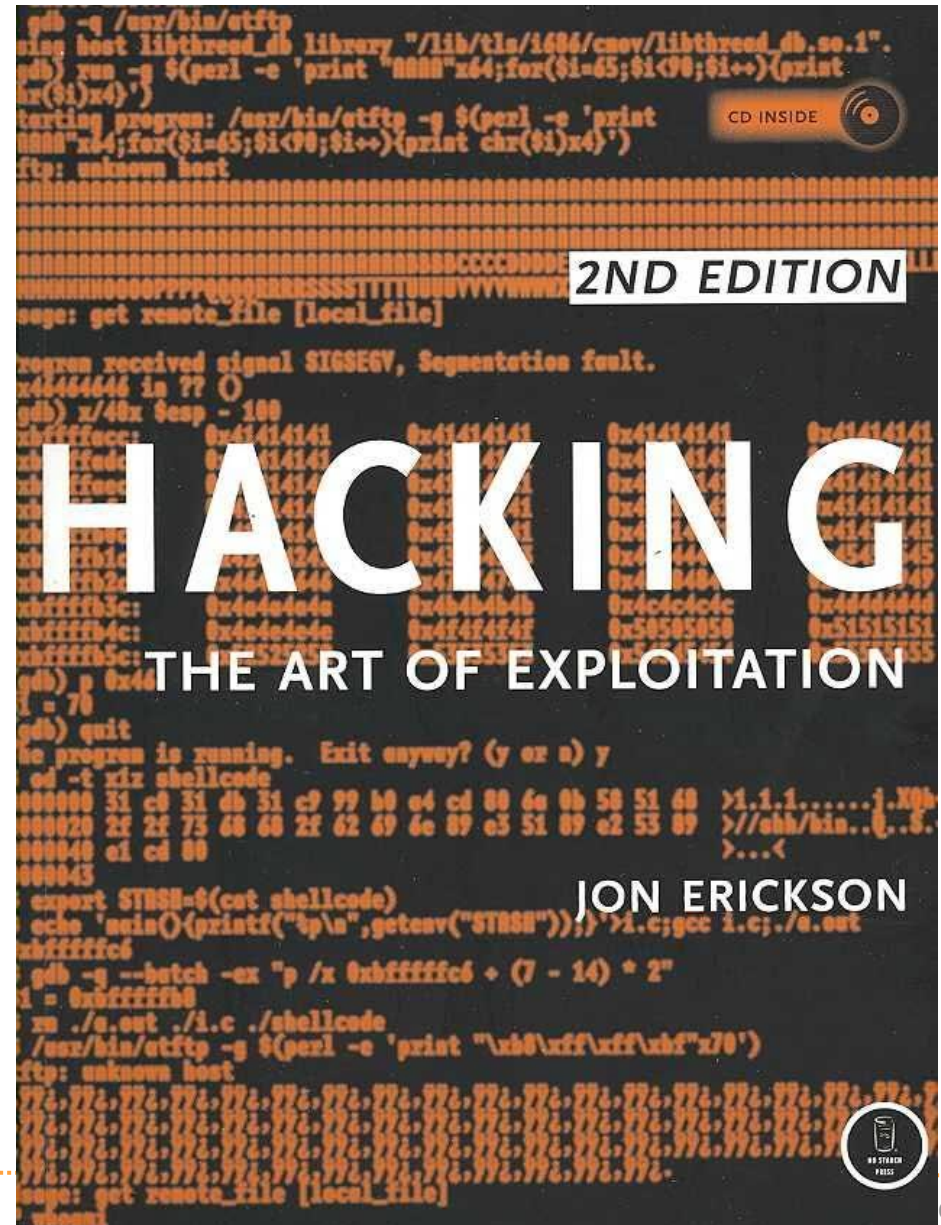
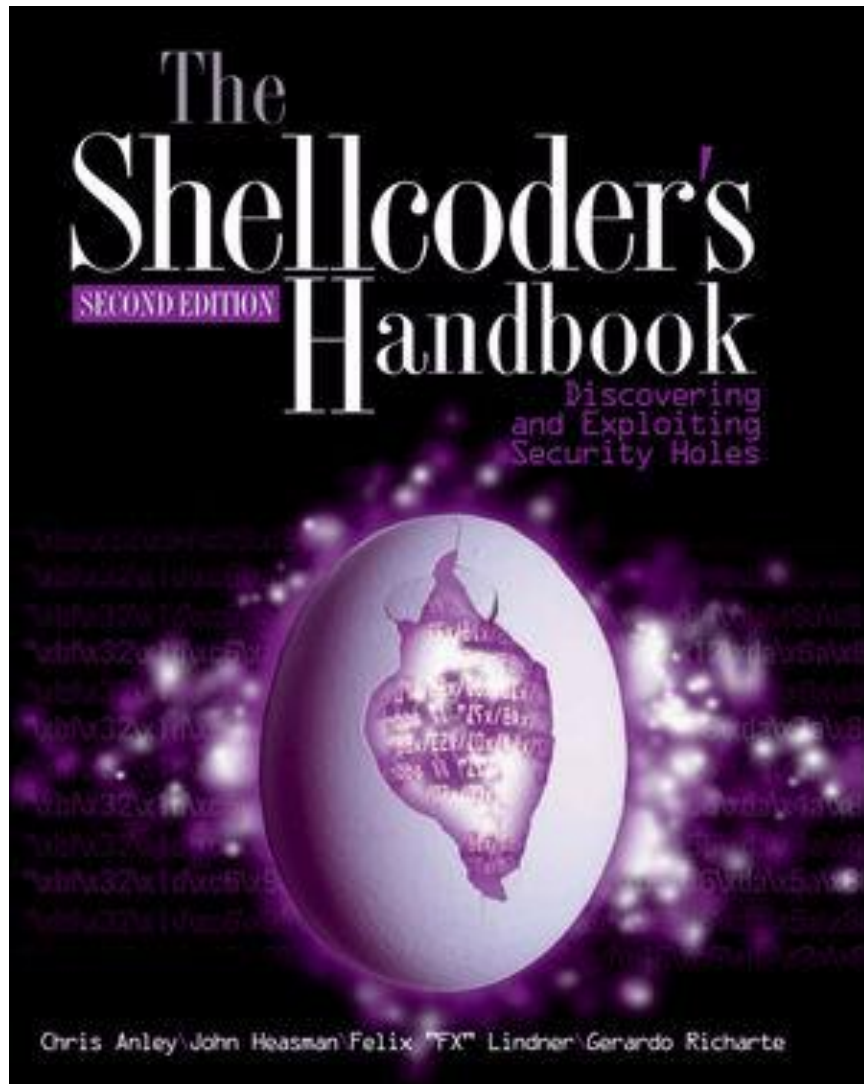
- ✦ How does memory corruption work?
- ✦ How does an exploit work?
- ✦ What exploit mitigations exist?
- ✦ How can these exploit mitigations be circumvented?

More theoretical, not so much the nitty gritty details

Typical question:

- ✦ Explain me how a buffer overflow exploit works
- ✦ Now we introduce ASLR. What do you need to change?

Books



Legal Issues

Don't hack other people's systems

«Damit der Tatbestand des **strafbaren Hackens** erfüllt ist, müssen **folgende Voraussetzungen kumulativ** erfüllt sein:

- ✦ **Eindringen** in das **Datenverarbeitungssystem**;
- ✦ **fremdes Datenverarbeitungssystem**;
- ✦ Eindringen auf dem Weg der von **Datenübertragungseinrichtungen**;
- ✦ **besondere Sicherung** gegen Zugriff.

<https://www.lexwiki.ch/hacken/>

Legal International

Wassenaar

- ★ Arms Control Treaty
 - ★ Anti-proliferation of Nukes and stuff
- ★ Includes now (?):
 - ★ Intrusion malware
 - ★ Intrusion exploits
 - ★ IP surveillance
- ★ -> Exploits are now weapons...
 - ★ Not allowed to transport over the border
 - ★ Exception: If they are open source
 - ★ (stop selling 0-days to Chinese gov!)



<http://blog.erratasec.com/2015/05/some-notes-about-wassenaar.html>