The Path to Becoming a "1337 h4x0r": Pursuing a Career in Reverse Engineering







Description of RE/VR and Why it Matters



Career Paths and How to Learn the Trade



Resources and Additional Inspiration

About Me

whoami

Name:

• Brian Hornak

Role:

- Cybersecurity Researcher at Battelle Memorial Institute **Experience**:
- 7+ years in Cyber: Embedded Systems + Microelectronics Education:
- B.S. Electrical Engineering Univ. of Maryland
- M.S. Electrical Engineering George Washington Univ.



whoami

"Cybersecurity Researcher"? What does that really mean?



Breaking devices, voiding warranties, and (hopefully) putting them back together

How does it work? What's on it? What can I learn?

Hardware Hacker



What can I make it do?

whoami

"Cybersecurity Researcher"? What does that really mean?

Verifying a device's authenticity to stop malicious intent and activity

Is this device what is claims to be? Can I trust the components?

How can I be certain the device is "clean"?





My Journey to Reverse Engineering



But why?

Motivations

Every Device is a Challenge

Like Solving a Puzzle...

National Security Implications



Constantly Learning

Tip of the Spear Research

It's Fun (and rewarding)

What is "RE/VR"



Cyber Context



Hardware

Software

In Cyber systems, software and hardware are inherently linked



Examining HW



Investigate the PCB for all the components

Look for "points of interest"

Can be manual or automated

PCB Teardown – Source: voidstarsecurity.com with personal edits

Examining SW

💋 CodeBrowser: NSMBU:/red-pro2_v1.3.0 — 🗆 🗙			🗳 CodeBrowser(2): NSMBU:/i	red-pro2_v1.0.0	- 🗆 ×
File Edit Analysis Navigation Search Select Tools Window Help File Edit Analysis Navigation Search Select Tools Window Help					
🔲 (B A I D U L F K W B	- 👔 🛍 🗠 🖂 🥒 🕼 🖄 🛅 😋 🚠	🗐 (😓 • 🔿 •) 📑 🖻	PR JIDULFKWB	🔹 ǎ 🖆 ରାଜ ଜା 🏑 🕅 🖄 🛅 😋 🚠
Program Trees X	E Listing: red-pro2_v1.3.0	🗅 💼 💽 🗮 🔣 📾 🗐 • 🗙	Program Trees 🗙	E Listing: red-pro2_v1.0.0	🗅 🜔 🔖 🛱 🦌 💩 💷 • 🗙
🗔 🗁 🔁	red-pro2_v1.3.0 🗙		n 🔁 🔁	red-pro2_v1.0.0 🗙	
- 🔛 .fimport_gx2 🔺	48 a0 02 45 b1	FUN 02a0a9c4	🖃 🏹 red-pro2_v1.0.0	48 9a d1 99 b1	sead::ExnHean::relativeToAbsolute ^
.fimport_nsysne	7c 1f 18 40 cmplw	r31.r3	💽 .syscall	7c lf 18 40 cmplw	r31.r3
.fimport_nlibcurl	41 80 01 94 blt	LAB 0200a91c		41 80 01 94 blt	return
.fimport_nn_aoc			extern		
.fimport_nn_olv.	LAB 0200a78c	x	- 🗟 .rodata	LAB 0200a764	, I
.fimport_sysapp	3c 00 10 00 1is	r0.0x1000	🗟 .data	3c 00 10 00 11s	r0.offset DWORD 10000ce0
	7f c5 f3 78 or	r5.r30.r30	.module_id	7f c5 f3 78 or	r5.r30.r30
< >	38 60 00 00 11	r3.0x0	.bss	38 60 00 00 li	r3.0x0
Program Tree X	30 00 0c e0 addic	r0,r0,0xce0	Program Tree X	30 00 0c e0 addic	r0,r0,offset DWORD 10000ce0
Trogram free in	3d 80 10 00 lis	r12.0x1000	Trogram free in	3d 80 10 00 lis	rl2.offset aActorresloader
🚠 Symbol Tree 🛛 👌 🏲 🗙	38 81 00 08 addi	r4.r1.0x8	🔜 Symbol Tree 🛛 👌 🏲 🗙	38 81 00 08 addi	r4.r1.0x8
	90 01 00 0c stw	r0=>DAT 10000ce0.local 3c(r1)		90 01 00 0c stw	r0=>DWORD 10000ce0.0xc(r1)
Exports	39 8c 0c f8 addi	r12,r12,0xcf8	Exports	39 8c 0c f8 addi	r12,r12,offset aActorresloader
Exports	7c 67 1b 78 or	r7,r3,r3	Exports	7c 67 1b 78 or	r7,r3,r3
	38 c0 00 01 11	r6,0x1		38 c0 00 01 1i	r6,0x1
Classes	91 81 00 08 stw	r12=>s ActorResLoader 10000cf8.loc	E Classes	91 81 00 08 stw	rl2=>aActorresloader,0x8(rl)
Amespaces	48 9f ff f5 bl	sead::ExpHeap::tryCreate((uint,sea	Amespaces	48 9a cf 49 bl	<pre>sead::ExpHeap::tryCreate((uint,se</pre>
	7c 78 1b 79 or.	r24,r3,r3		7c 78 1b 79 or.	r24,r3,r3
	38 60 00 01 1i	r3,0x1		38 60 00 01 li	r3,0x1
	3f 20 10 1f 1is	r25,0x101f		3f 20 10 1d 1is	r25,offset HeapMgr
	1- 41 82 00 10 beq	LAB_0200a7d8		41 82 00 10 beq	LAB_0200a7b0
Filter:	80 79 80 40 1wz	r3,-0x7fc0(r25)=>DAT_101e8040	Filter:	80 79 82 e0 lwz	r3,-0x7d20(r25)=>HeapMgr
	7f 04 c3 78 or	r4,r24,r24		7f 04 c3 78 or	r4,r24,r24
🗊 Data Type Manager 🔻 🗙	48 a0 2d b1 b1	FUN_02a0d584	🗊 Data Type Manager 🔻 🗙	48 9a fc 85 bl	<pre>sead::HeapMgr::setCurrentHeap_((s</pre>
(= ː => ː 🌿 ː 📉	130 0200-240	ν 🗧	🚝 T 🔿 T 😘 T 📉	LAP 0000-750	. Ξ
📐 🖂		x26.0x1	🕅 🖃	2c la 00 01 cmmri	x26.0x1
A Data Types	7c 7f 1b 78 or	r31.r3.r3	Ala Types	7c 7f 1h 78 or	r31.r3.r3
🗄 🧉 BuiltInTypes	40.82.00 hc hre	LAB 0200a89c	🗄 🧉 BuiltInTypes	40.82.00 hc hpe	LAB 0200a874
⊕ i ored-pro2_v1.3.0	3f c0 10 1f lis	r30.0x101f		→ 3f c0 10 1d 1is	r30.offset spriteTokctorList
🗄 🧃 generic_dib	3e e0 10 le lis	r23.0x101e	💼 🧃 generic_dib	3e e0 10 1c 1is	r23.offset Level inst
	3b de cd 8c subi	r30.r30.0x3274		3b de cf 3c subi	r30.r30.offset spriteToActorList
	3a c0 00 04 1i	r22.0x4		3a c0 00 04 11	r22.0x4
	→ 3b 60 00 00 1i	r27.0x0		3b 60 00 00 1i	r27.0x0
	LAB_0200a7f8	x 🗸		LAB_0200a7d0	> v
Filter:	<	>	Filter:	<	>
Image: A start of the start	0200a7f4 FUN_0200a738	li r27,0x0	1	0200a7bc LoadActorRes	lis r30,0×101d
📲 🔎 💷 🧕 🔁 📴 🎒 🚍 🏶 🏶 🤜 🦣 👹 🖸 💷 🔍 🖻 🥲 🚱 🕫 🥵 🕸 👫 👼 👶 🥵					

Decompile binary data to "code"

Attempting to generate source

Use disassembler

Binary Disassembly in Ghidra – Source: Ghidra github repo

The RE/VR Mindset



Thinking Like an RE/VR

You are given a black box (literally) that you know nothing about other than what you can immediately see in front of you?

Your goal as a reverse engineer is to determine:

- What it is?
- What it does?
- How it works?
- What is is used for?



Your goal as a vulnerability researcher is to determine:

- Are there flaws in this system that I can fix or utilize?
- What hidden capabilities are available to me?

The Skills You Need

"Hard" Skills

- Programming languages
 - C, C++, Python, Assembly
- Electronics
 - PCB components
 - Voltage and Current
 - Data storage
- Communications
 - Wi-Fi, Bluetooth
 - Serial communication
 - Analog vs. Digital

"Soft" Skills

- Innate curiosity
- Attention to detail
- Patience
- Even MORE patience!

The RE/VR Tool Bag

Hardware Tools

Oscilloscope

Multimeter

Soldering Iron

Software Tools

Disassembler (Ghidra, IDA, etc.)

Virtual Machine

Debugger (GDB, etc.)

Tool	Purpose		
Oscilloscope	Measure system signals		
Multimeter	Detect voltages and connections		
Soldering Iron	Remove/add components or wires		
Disassembler	Turn binary data into readable code		
Virtual Machine	Research safely away from host		
Debugger	Investigate software behavior		

Example

Wi-Fi Connected (uh oh), "Smart" Toaster

Let's imagine we purchased this toaster but weren't satisfied with how well it toasted.

Could we pry it open to learn more about how it works, adjust something relevant to our problem, and put it back together to achieve the perfect toast?



Revolution R180 Smart Toaster: https://www.amazon.com/Revolution-High-Speed-Touchscreen-Technology-Connectivity/dp/B0D5MGPXG8?th=1

Career Path for RE/VR

Opportunities

Industries and Major Players

- USG:
 - NSA, DHS, DOJ, IC, DoD, Dept. of Commerce
- National Security:
 - Battelle (that's it, I promise)
- Private Industry:
 - Google Project Zero
 - FAANG (+ Microsoft)

But how do l get started?



Getting Started





Internet Write Ups and Blog Posts





Resources

Links and Fun Stuff

- <u>http://trailofbits.github.io/ctf/</u>
 - Overview of CTFs, problem categories, tool recommendations and some problem walkthroughs
- <u>https://microcorruption.com/login</u>
 - Embedded system CTF focused on binary exploitation/memory corruption that starts at absolute beginner and works up to Advanced problems
- <u>https://crackmes.one/</u>
 - repository of RE challenges that are rated by users for difficulty and quality
- <u>https://ctftime.org/</u>
 - Upcoming CTFs, pick one and dive in. Also is a great place to find writeups for CTF problems from past CTFs
- <u>ROP Emporium</u>
 - Series of challenges that help you learn and practice return-oriented programming
- <u>pwn.college</u>
 - Educational platform with variety of (free) challenges
- Void Star Security Blog
 - Blog on all things RE/VR with specific emphasis on hardware hacking

Questions?