Breaking into Vulnerability Research

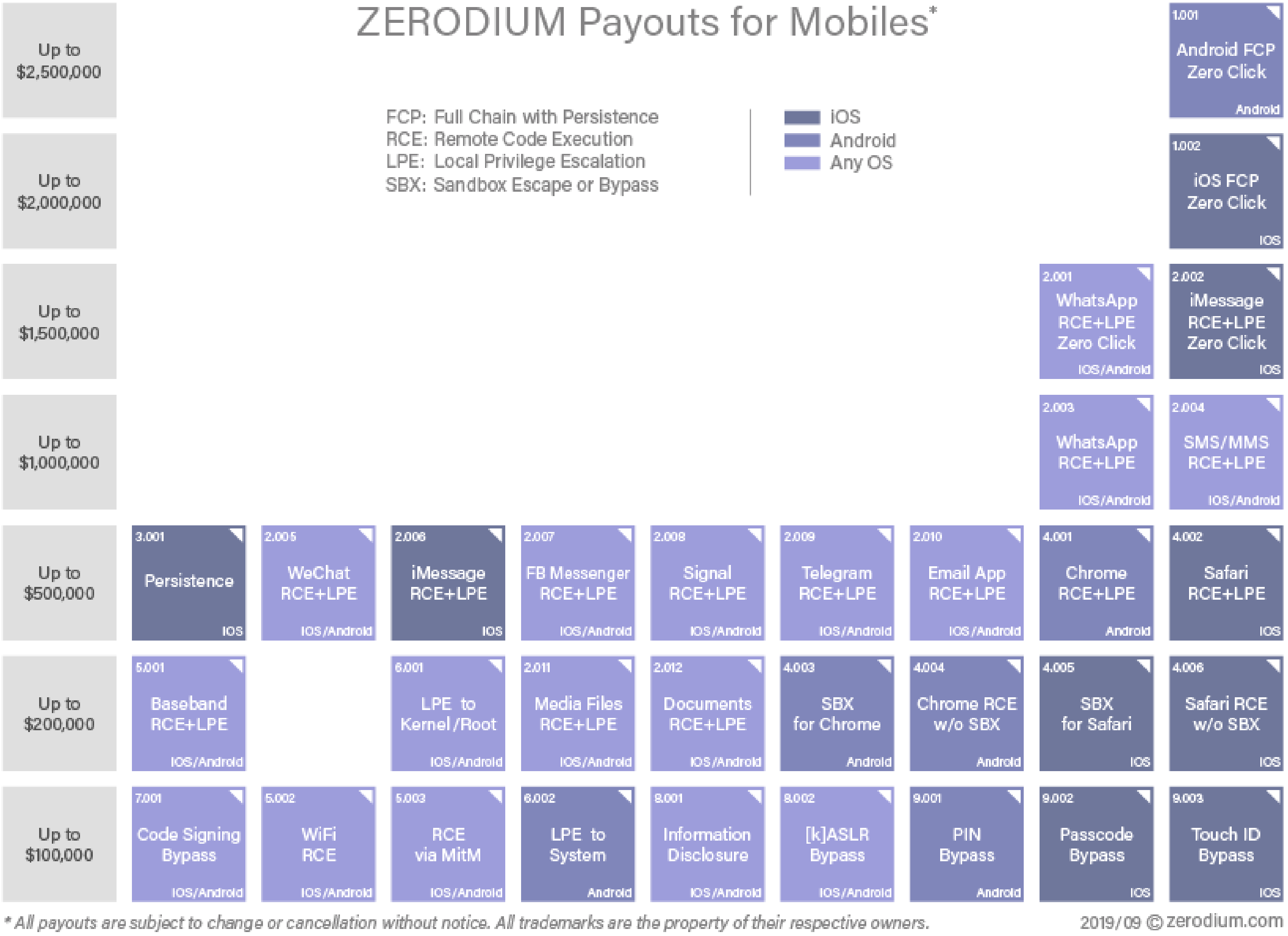
Dr Silvio Cesare
CTO, InfoSect

What is Vulnerability Research?

- The ability to reliably and repeatedly discover and exploit 0-days in hardware or software target or targets:
    - That is ubiquitous in nature and most likely essential systems software
    - That is in the default configuration
    - That is hardened and defended
    - With some strong effect such as remote code execution (RCE) or local privilege escalation (LPE)
- Any attack that achieves the objective (e.g., RCE or LPE) is valid
- In practice, this is likely through memory corruption and against native code like C or C++

Example Targets

- Operating System (OS) Kernels
- OS Network Services
- Hypervisors
- Browsers
- Instant Messaging Applications
- Mobile Handsets
- Embedded Devices
- Zerodium “exploit broker” bounty list is a good start …



ZERODIUM Payouts for Mobiles*

FCP: Full Chain with Persistence
RCE: Remote Code Execution
LPE: Local Privilege Escalation
SBX: Sandbox Escape or Bypass

■ iOS
■ Android
■ Any OS

Up to $2,500,000									1.001 Android FCP Zero Click Android
Up to $2,000,000									1.002 iOS FCP Zero Click iOS
Up to $1,500,000								2.001 WhatsApp RCE+LPE Zero Click iOS/Android	2.002 iMessage RCE+LPE Zero Click iOS
Up to $1,000,000								2.003 WhatsApp RCE+LPE iOS/Android	2.004 SMS/MMS RCE+LPE iOS/Android
Up to $500,000	3.001 Persistence iOS	2.005 WeChat RCE+LPE iOS/Android	2.006 iMessage RCE+LPE iOS	2.007 FB Messenger RCE+LPE iOS/Android	2.008 Signal RCE+LPE iOS/Android	2.009 Telegram RCE+LPE iOS/Android	2.010 Email App RCE+LPE iOS/Android	4.001 Chrome RCE+LPE Android	4.002 Safari RCE+LPE iOS
Up to $200,000	5.001 Baseband RCE+LPE iOS/Android		6.001 LPE to Kernel/Root iOS/Android	2.011 Media Files RCE+LPE iOS/Android	2.012 Documents RCE+LPE iOS/Android	4.003 SBX for Chrome Android	4.004 Chrome RCE w/o SBX Android	4.005 SBX for Safari iOS	4.006 Safari RCE w/o SBX iOS
Up to $100,000	7.001 Code Signing Bypass iOS/Android	5.002 WiFi RCE iOS/Android	5.003 RCE via MitM iOS/Android	6.002 LPE to System Android	8.001 Information Disclosure iOS/Android	8.002 [k]ASLR Bypass iOS/Android	9.001 PIN Bypass Android	9.002 Passcode Bypass iOS	9.003 Touch ID Bypass iOS

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

2019/09 © zerodium.com

Why do we have "standard" domains and targets?

- In pen-testing you might be given any software or hardware in an engagement
- In VR, there's often a 'set' of standard targets and domains
- The goal is to reliably, repeatedly, and consistently develop research outputs
  - Targets need to be complex to support a supply of high quality bugs
  - Targets need to support reliable exploitation
    - Most exploitation is multi-stage, requiring heap grooms, leaks, corruption, and so forth
    - Typically this means one-shot attacks are problematic
    - Programmatic feedback is desirable
  - Targets need to be ubiquitous and have desired privileges
    - Obscure packages that need a user to install them are not useful

A little about me

- In “hacking” for 25+ or so years
- Didn’t do bug hunting and exploit development for the entire duration of my career
- I did “hobbyist” bug hunting at various points in the first half of my career
  - Was one of the first public people to do large scale auditing of the Linux kernel in ~2002
- Most of the time working on low level native development, RE, or program analysis
- Did a PhD and research commercialization in program analysis based malware defense
- Was a university educator for a couple of years and still regularly run trainings at InfoSect
- Before transitioning to full time bug hunting and exploit development

InfoSect

- A 20 person VR company
- Team works on-premises
- Founded by myself (CTO) and Kylie McDevitt (CEO)
- This talk is about some of the challenges and lessons in building and running a VR company

The Founding Team

- Need business know how
  - But this is a talk about technical and engineering challenges, so we'll ignore this part
- Founders need to be generalist specialists
  - Founders will be setting up infrastructure
  - Doing research from finding bugs to writing exploits
  - Developing tooling
  - Etc.
- For InfoSect
  - My co-founder was a generalist, and a specialist at infrastructure
  - I started as the primary researcher

Sales or Engineering Focused?

- Most startups see the main problem of business as a sales problem
- They see that sales drives growth
- In a VR focused company, the main problem is engineering
- Better research output and an engineering focus drives growth

Some popular VR myths

- If you want to do VR..
- You’d already be doing it, but you aren’t …
- So if you can’t do VR against a target now …
- It’s only going to get harder
- And it’s only feasible, but very hard, just to maintain research outputs on hard targets
- So it’s impossible to catch up (since you’d already be doing it)
- TL;DR don’t even try

Breaking the Barriers

- CTF players often jump into hard problems
- VR newcomers enter the field all the time
- Existing VR companies can take on new targets or complexities as technology evolves

My own Transition

- There's a big difference between hobbyist bug hunting and VR
- VR is a large, deep, and complex field
- IMO it's basically impossible to find bugs reliably unless you've invested a large amount of time studying real-world code and targets
- I was good initially at finding bugs, but I lacked focus on finding relevant and impactful issues
- Over time I learnt target and domain specific knowledge and became better at identifying and focusing on relevant attack surfaces
- I then became faster at learning new targets and identifying target specific bug hunting strategies

Transitioning as a hobbyist bug hunter

- Most bugs are actually not that useful
- Non ubiquitous, non hardened nor defended software bases are generally extremely buggy
- Even large and flexible hardened code bases (e.g., OS kernels) have practically an unlimited number of worthless bugs
- It's great if you are a developer, but not great for VR
- Identifying attack surface and focusing on relevant default configurations is paramount

From 1 or 2 researchers to a team

- Leading by example is important
- Research is hard, don’t expect you can hire your way into a VR company without setting an example to the team you hire
- Hiring is challenging
- Do you hire VR veterans or seek new up and comers?

Where to hire?

- VR specialists and/or veterans
- CTF players good in pwn/RE etc.
- University graduates (after doing an internship)
- Pen testers and consultants looking to change careers
- Software developers (e.g., OS kernel or embedded)
- Forensics people (e.g., for firmware recovery in embedded work)
- Electronics technicians/engineers

Hiring Process

- We use a ‘trial’
- Given a task, indicative of the type of work we do, work on it
- Sit with the team for 1-3 days
- The task is not unrealistic for the time
- Normally take a fairly pointless 0-day from a real world target (e.g., pointless because it needs privileges to trigger)
- And verify that it exists with a PoC

Hiring Process Results

- You identify culture fit and if they can work in a team
- Recruit knows what type of work we do and if it meets expectations
- It’s very evident if the recruit knows basic skills around a terminal
- That they can write code in C etc.
- That they can read real-world code (to figure out how to trigger the bugs)
- If they complete the task quickly, we keep giving them further tasks..
  - Turn the PoC into some kind of more useful exploit (e.g., take an infoleak and turn it into a KASLR bypass)
  - Get given more bugs to work up, rinse and repeat

Selecting Target Domains

- The targets in VR are varied but not unlimited
- You need to aim for realistic targets if starting out
- This will almost certainly be dictated to what the founding team's specialties are
- And likely your early hires will support these specialties
- As the company grows, you can go more broad

Growing

- Typically you’ll start off with a relatively isolated component you’re targeting
- And you’ll add more components as your team improves
- These components might be useful together
- Depth and specialisation is the only way to be effective
- In general, more junior people focus on building enough depth on a narrow area to be a useful member of the team
- As they become more senior, greater depth slows down and is roughly the same, but they become broader

Knowledge Sharing

- Knowledge sharing is paramount
- The goal is to have a team that lifts each of its team members up
- Many approaches can be used
  - Internal documentation
  - Internal presentations
  - Having a collaborative team culture
  - Having staff feel comfortable in asking and receiving help
- Having knowledge trapped within in an individual and not having the opportunity to share with the rest of the team can limit how fast the team can grow and achieve complex objectives

Team Culture

- When the company is small, culture is built, and then becomes hard to change as the company grows
- VR is very different to pen testing and consulting – the culture is not the same
- Your engagement isn't to find a weakness with a person who implemented a sloppy misconfiguration
- But to work against a large team of expert developers who spend their days trying to write functional and hard to attack software
- You have a greater understanding of the skill that developers have versus the general security industry
- However, your job is to reliably find bugs in their software

New Starters

- Giving new starters large open ended problems may be effective with VR veterans
- But for people new to VR, it can be demoralizing
- Since output may be slow, inconsistent, non-existent, or weak
- I prefer progressing VR newbies from small tasks such as verifying a simple bug, say by writing a small PoC to prove that the bug exists (or doesn’t exist)
- And then progressively giving harder tasks
- The more real and consistent output people have at the beginning of their employment, the more confidence they build

Team Confidence

- New teams need to quickly find confidence
- And confidence is built by the founders and the rest of the team demonstrating competence
- And achieving seemingly impossible objectives
- Results do not always come quickly, but major findings must come at regular intervals
- This also promotes new people to join the company via referrals from existing staff
- If major findings aren’t coming, then a direction shift or change of strategy must occur to stay on track

Engineering

- Almost certainly engineering will take on a role
- Beyond purely doing research
- If you’re a small team, researchers will be required to do some engineering work
- This can be motivating in the sense that a problem is solved and a solution is seen working
- However, researchers in general prefer research

Engineering and Professionalisation

- At some point, the focus will be to make everything as professional as possible
- Coding and research output standardisation occurs
- Systems are in place to automate things as much as possible
- Internal design patterns and coding become solidified

Exploit Reliability

- The goal is as close to 100% reliable as possible
- Not always the case
- It’s generally not desirable to have long run times
  - A refcnt overflow that takes 4 hours to trigger with a core pegged at 100% CPU isn’t considered practical for many people
- In hobbyist approaches, ROP is seen as a reasonable exploitation strategy
- And sometimes, that’s all you can do
- It is sometimes considered to be quick and dirty
- Now your exploit is version specific and needs constant maintenance to support ROP chains for new software releases …

Infrastructure

- Your network will be vastly more complex than is reasonable for a comparably sized company
- Good infrastructure people can be somewhat challenging to hire
  - You need people very senior to understand your setup and requirements
  - But you don’t have a telco sized network
  - So the work, while interesting, isn’t classically desirable by infrastructure people
  - For InfoSect, my co-founder (and CEO) is a very senior infrastructure specialist
- If one of your domains is embedded (like InfoSect), then your lab is likely to grow
  - You might want an RF shielded room
  - You might want heavy and expensive machines

Is there a technical glass ceiling?

- Many traditional tech jobs have essentially a glass ceiling
- This is true regardless of what people advertise on the brochures
- Certainly in Australia at least
- Many techs reach their top level, their salary effectively maxes out, and the work they do becomes routine
  - They then struggle to decide if they want to transition to a non-tech role for more impact..
  - The same is true even in much academic research as senior professors tend to seek impact by becoming more administrative and supervisory in nature
- In VR, these things are simply not the case
- The more you learn, the deeper you go - the more success you achieve
- Proportionately, the higher your salary and bonuses become …

Is VR worth the hype?

- Is it a good job?
- TL;DR yes
- It’s the best job I’ve ever had (and not just because I’m a founder)
- It’s the most technically challenging career I’ve experienced in my 30 years of working
- I work with an amazing team
- We do amazing stuff

Conclusion

- There's lots of things about running a company
- And to run a company focusing in VR
- I've seen exploits referred to as "modern day magic"
- Which is an apt description