
Google Project Zero

ANDY GREENBERG SECURITY 07.15.14 06:30 AM

MEET 'PROJECT ZERO,' GOOGLE'S SECRET TEAM OF BUG-HUNTING HACKERS

Hacking wunderkind George Hotz's latest gig: An intern on Google's elite hacking team.

TRIBUNE REVIEW, ANDREW RUSSELL/AP

MAKE
0DAY
HARD

is

~~MAKE~~

0DAY

HARD

yet?

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

Good defense requires a detailed knowledge of offense.

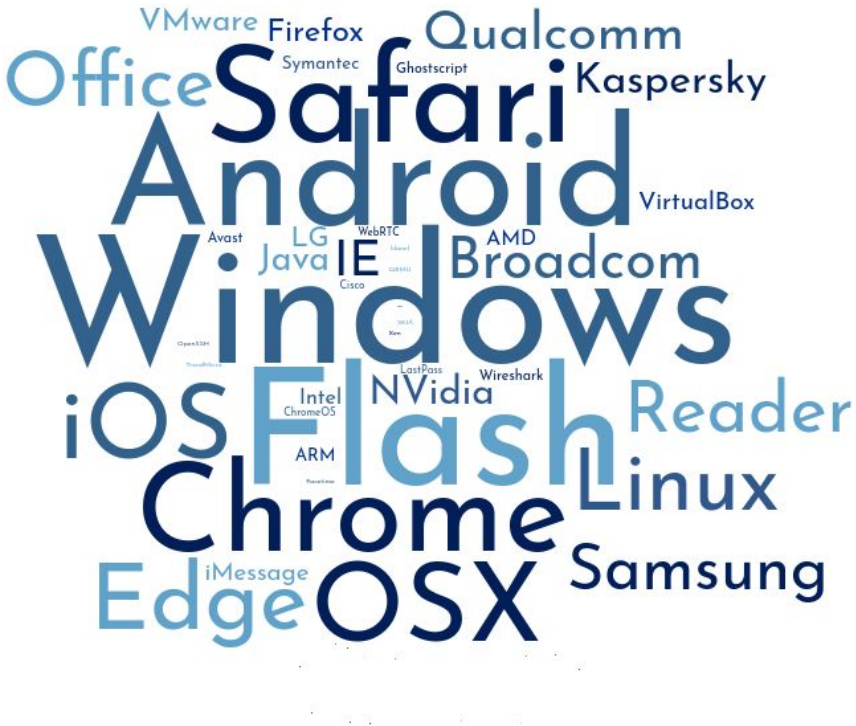
Attackers target the weakest link in the chain.

Openness benefits defenders more than it benefits attackers.

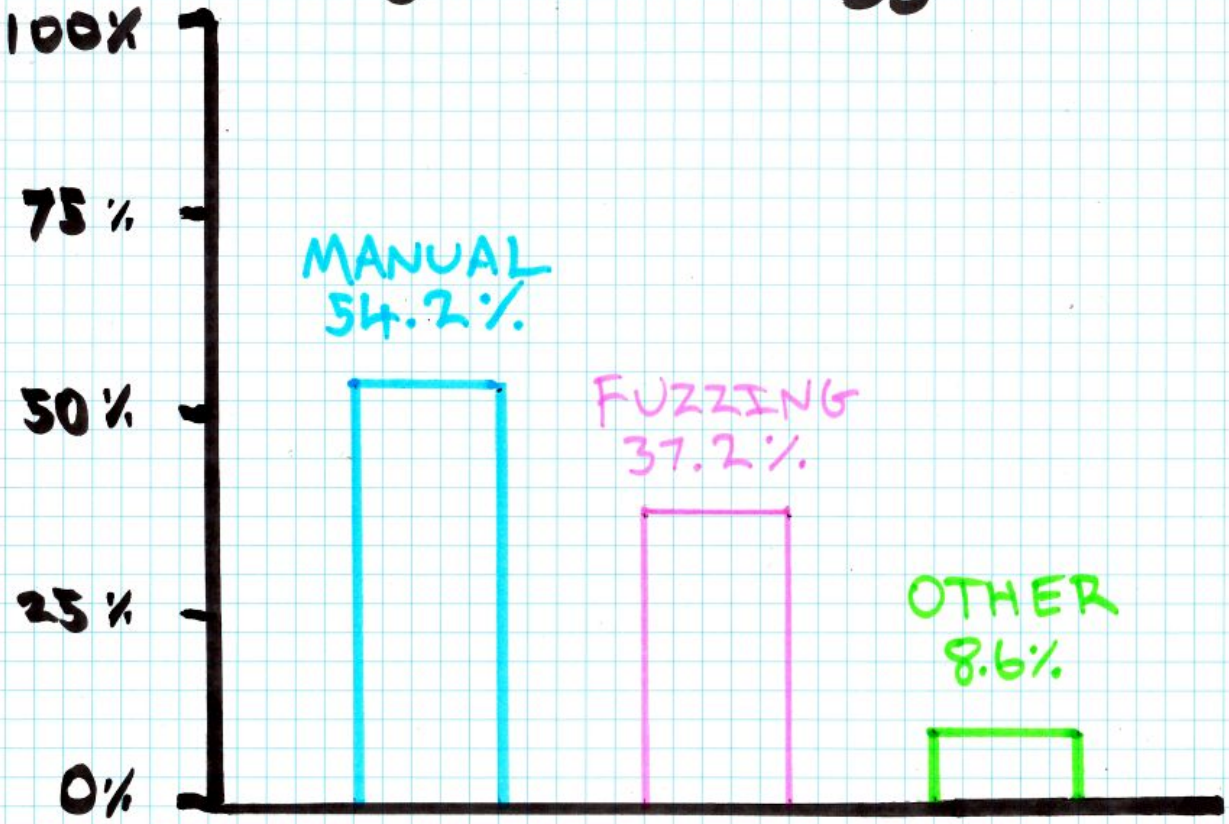
Challenging industry norms leads to improved security.

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

- Vulnerability research
- Exploit development
- Methodology building
- Technical writing
- Exploratory reading
- Chatting with peers
- Working with vendors/OSS projects
- Software engineering
- Presenting at conferences


VMware
Firefox
Qualcomm
Office
Symantec
Ghostscript
Kaspersky
Safari
Android
VirtualBox
Avast
LG
WebRTC
AMD
Java
IE
Broadcom
Windows
Cisco
LastPass
Wireshark
iOS
Intel
NVidia
Reader
Flash
ChromeOS
ARM
Linux
Chrome
Samsung
iMessage
Edge
OSX

PROJECT ZERO DISCOVERIES (by methodology)



New methodologies for vulnerability discovery should either:

a. Find bugs faster than we currently are, or
b. Find bugs that we can't currently surface.

Exploit Development

1. Ensures that the security impact of the bug is well understood
2. Establishes an equivalence class of similarly exploitable vulnerabilities
3. Surfaces new and improved exploit techniques
4. Generates appropriate amounts of urgency
5. Allows us to find areas of "fragility" in the exploit

Structural Change

- Attack surface reduction
- Better sandboxing
- Exploit mitigations
- Fixing bug classes
- Process improvements
- Improved documentation

advocates more than decision makers

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

Project Zero by the numbers

100+ technical blog posts

1500+ vulnerability reports

11,500,000+ blog page views

100,000,000+ disclosure debates

$$\frac{\text{cause}}{\text{effect}}$$

Spectre and Meltdown -- by Jann Horn et al.

- What happens when a Project Zero researcher tries to bypass a custom kernel mitigation? Spectre and Meltdown!
- Four variants found by Project Zero, many others in follow-up research.

- Industry-wide shift in our understanding of CPU security boundaries.
- Significant architectural changes in kernels, hypervisors, browsers.
- Hardware industry invested in security, building teams + processes.

task_t considered harmful -- by Ian Beer

- Design flaw in the XNU kernel (iOS and macOS); privilege escalation.
- "Every task_t pointer is a potential security bug"
- Multiple iterations required to fix the issue correctly.

- Fixing an entire bug class by refactoring how "execve" works!

Flash -- team effort

- Found and reported 200+ Flash UaF, OOB R/W and type confusion bugs
- Primarily by Natalie (manual review + fuzzing) and Mateusz (fuzzing)

- Worked with Adobe and Microsoft to implement exploit mitigations
- James implemented win32k lockdown for Chrome's plugin process
- Worked with Chrome to use these results to expedite Flash click-to-play

MPEngine and WPAD -- by Tavis, Ivan, Halvar

- MPEngine "hackathon", organized by Tavis: 9 bugs
- WPAD exposes jscript.dll, exploited by Ivan and Halvar: 12 bugs
- Interesting proof-of-concept exploits for both!

- Both of these attack surfaces are now sandboxed!

Interlude: a note on structural improvements

- Project Zero's vulnerability research and exploit development work is highly visible and typically very popular

- However our work on structural improvements is (at least) equally important, but typically happens in a relatively low key fashion

Broadcom WiFi -- by Gal Beniamini

- Broadcom WiFi SoC RCE + AP kernel code execution
- Full device takeover over Wi-Fi, requiring no user interaction.
- 16 distinct vulnerability reports in total, 5 part blog post series.
- Approximately 6.5 months of work.

Broadcom WiFi -- Exploit Development

- Two distinct exploit chains: Android and iOS.

- Primitive (Android): **25 byte linear overflow.**
  - Then used the total lack of IOMMU configuration on Android

- Primitive (iOS): **16-bit increments, up to 60 bytes past allocated buffer.**
  - Then bypassing DART IOMMU with a driver vulnerability

Broadcom WiFi -- Structural Improvements

- For Broadcom: improved their communication and PSIRT triage processes, and started pursuing exploit mitigations (e.g. a properly configured MPU, allocator hardening).

- For mobile vendors: Motivated adoption + configuration of IOMMUs

- For researchers: published tools for debugging/analysis of firmware.

Project Zero summary of results

50+ structural improvements

100+ technical blog posts

1500+ vulnerability reports

11,500,000+ blog page views

100,000,000+ disclosure debates

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

How do we measure the "hard" in "make 0day hard"?

Perhaps the price of exploits?

The number of vulnerabilities?

How many exploits are detected?

How do we measure the "hard" in "make 0day hard"?

~~*Perhaps the price of exploits?*~~

~~*The number of vulnerabilities?*~~

~~*How many exploits are detected?*~~

These aren't satisfactory at all!

Attempt #2 -- are better measures possible?

Time spent finding a "good" vulnerability?

Average bug lifetime -- presumably shorter is better?

Number of rumored or confirmed bug collisions?

Length of the average exploit chain?

Rate of "good" new attack surfaces being discovered?

Better, but still not entirely satisfactory.

Assessing "progress towards hard" instead

1. Is the **time spent** building an equivalent exploit chain to an equivalent level of reliability today more than it was last year?

2. Is the **rational shape** of an exploit today different from last year?

"Hard"

- Perhaps hard is in **relative terms**: *"with the same operational requirements, 0day is less cost effective than other available attacks."*
- Or perhaps hard is in **absolute terms**: *"only the top N% of all actors that want to maintain a 0day capability are actually able to do so."*

- Regardless, **the threshold between *not hard* and *hard* is unlikely to be precisely measured**. Progress towards hard *is* measurable however.

is

~~MAKE~~

0DAY

HARD

yet?

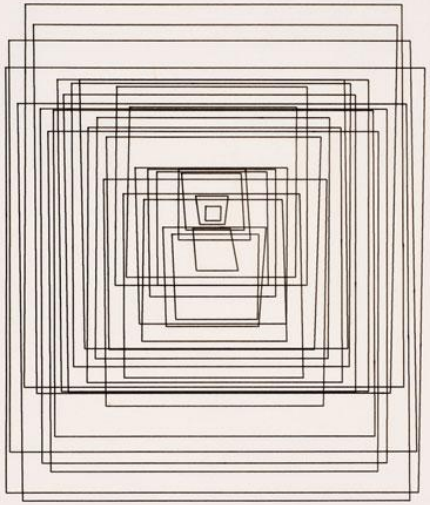
It's harder.

It's harder.

But not hard.

74.311 15.38.44

JOB FROM MOLNAR



1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

Key Message

 # 1 - There is a risk that the public state-of-the-art in attack research diverges from the private state-of-the-art.

2 - The "Project Zero model" is a viable approach to countering this type of divergence.

3 - And finally, open attack research provides the best path forward for "make 0day hard".

Key Message

1 - There is a risk that the public state-of-the-art in attack research diverges from the private state-of-the-art.

 # 2 - The "Project Zero model" is a viable approach to countering this type of divergence.

3 - And finally, open attack research provides the best path forward for "make 0day hard".

Key Message

1 - There is a risk that the public state-of-the-art in attack research diverges from the private state-of-the-art.

2 - The "Project Zero model" is a viable approach to countering this type of divergence.

 # 3 - And finally, open attack research provides the best path forward for "make 0day hard".

Lightning Round #1 -- Data Is the Beginning, Not the End

- The data you see is not the data you need.
- Attackers are incentivized to withhold and mislead.

- We need to model attacker behavior in order to make optimal decisions.
- ... but attackers don't want to be modeled.

- The consequence: we easily believe what we want to believe.

Lightning Round #2 -- Immortal Disclosure Debates

- Also a result of #1: we will **never** finish debating vulnerability disclosure.

- To arrive at "optimal", there are key unknowns that need to be modeled, and lots of ways that you can model them.

- There's a very wide range of justifiable disclosure policies, and this range evolves over time.

Lightning Round #3 -- The Comfortable Model

- Danger! When a chosen model for attacker behavior results in policies that align with business goals, the model itself becomes sacrosanct.

- How do you incentivize companies/organizations/groups/individuals to improve their models for attacker behavior when the result is less desirable for their immediate goals?

Lightning Round #4 -- Vulnerability Triage Blues

- The work of responding to external bug reports is incredibly undervalued
- Done well, it can be profoundly impactful work. Done poorly, it's a systemic risk.
- We should *urgently* find ways to increase the prestige and enjoyment levels of the people doing this work, and aim to retain them in this role.

Lightning Round #5 -- Opportunism is Magic

- Work with strategic impact doesn't need to start with a strategic plan.
- Results from curiosity-based prioritization lead to strategic impact at higher than expected levels.
- The ability to pull a promising thread, even "off topic" to the current project, generates a huge amount of energy and momentum.

Lightning Round #6 -- Vulnerabilities Matter

- We can't engineer our way out of the problem with mitigations alone. If vulnerabilities are plentiful, "edge case" bugs for the mitigations are too.

- Getting on the right side of the ratio of bugs fixed vs bugs introduced is a valid goal to have.

- The combination of bug fixing and structural improvements works best!

Lightning Round #7 -- A Small Aside on Bug Collision

- Publishing research gives you a global perspective on bug collision.
- We learn of bug collisions through private conversations made possible by being public figures.
- This is not a perspective that attackers are typically able to see.

Lightning Round #8 -- Greatness

- Like all pursuits, there's a difference between "good" and "great" in security research. What is it though?

- In objective terms, it remains a mystery. But we know it when we see it!

- Pursuing greatness involves taking risks. Managing the "rebound" from failure is seemingly key. **Embrace and celebrate negative results.**

1. Founding Principles
2. Team Operation
3. Classic Hits
4. Measuring Success
5. Lessons Learned
6. Next Steps

Plan A

Vulnerability Research

Exploit Development

Structural Improvements

"pop shells, ship sandboxes!"

Some Old Ideas and Some New Ideas

- Cross-disciplinary hybrid roles, like 0day "in the wild" analysis.
- Fully explore the tension between partnerships and independent research. What are the trade-offs, and which works better?
- Provide subject matter expertise in relevant policy discussions.
- Share our results with new audiences (other than researchers).
- Help vendors/open source to improve the "patch gap".
- Learn how to support more structured collaboration.

Revisiting the "Key Message"

1 - There is a risk that the public state-of-the-art in attack research diverges from the private state-of-the-art.

2 - The "Project Zero model" is a viable approach to countering this type of divergence.

3 - And not only that, **open attack research provides the best path forward** for "make 0day hard".

More open attack research, so perhaps...

more Project Zeros?

Building an Open Attack Research Coalition

- A coalition of open attack research teams has always been an implicit hope, but never an explicit goal.

- Industry, academic institutions, non-profit/NGOs, and government all have a role in ensuring that "make 0day hard" becomes a reality.

- Focusing on common mission + principles gives reasons for optimism.

A New Goal

A coalition of open attack research teams within the next five years.

A New Goal

A coalition of open attack research teams within the next five years.

One possible sketch:

1. *Website: organize open attack research results in an accessible way.*
2. *Mailing list: enable lightweight collaboration and technical discussions.*
3. *Summit: discuss trends in attacker behavior in a structured environment.*

Google Project Zero

Thank you! Questions?

googleprojectzero.blogspot.com

hawkes@google.com