hackerone

Turning Security Inside Out

THE GITHUB BUG BOUNTY STORY

Learn how hacker-powered security illuminated GitHub's security blind spots, resulted in an ROI of "phenomenal," and became a permanent part of their security program.

CUSTOMER STORY

GitHub

Customer Data

GitHub

STATUS
Public

NO. REPORTS SUBMITTED
795+

HACKERS THANKED
70+

TOTAL BOUNTIES PAID
$125,000

AVERAGE BOUNTY
$200 - 500

TOP BOUNTY
$12,000

COMPANY SIZE
600 Employees

INDUSTRY
Software

COMMUNITY SIZE
20 Million +

GitHub supports a community where more than 20 million people learn, share, and work together to build software.

It's a massive undertaking, and it carries with it the responsibility of ensuring the 55 million projects created by those 20 million people are safe and secure. Earning that trust means working with hackers who find and report issues, then following through to show appreciation for their efforts.

But what began as a tedious, manual, and highly-customized bug bounty program quickly morphed into a streamlined, efficient, and highly-automated program hosted by HackerOne.

It Began With a Need for Easier Communication...

GitHub doesn't take their users' trust for granted. They not only want to earn that trust, they want to keep it by continuously improving the security of their services.

GitHub had been using all the typical approaches to identifying vulnerabilities: red teams, third-party assessments, and automated and manual bug detection. In 2014, GitHub launched their bug bounty program to "have more eyes" to track down vulnerabilities. They even integrated HackerOne's report data with an internal support tool to help accept reports and manage communications.

"By providing welcoming arms, we hoped that people who find issues in GitHub will turn to us before notifying anyone else," said Neil Matatall, Security Engineer at GitHub. "We're immensely appreciative of their work, and we want to make sure they know it."

In April of 2016, GitHub transitioned to the HackerOne platform to increase process control and have an easier means for finding, communicating with, and rewarding hackers. Now, after three years of bounties, the program has become a critical component of GitHub's overall security apparatus.

"We see the bug bounty program as a permanent part of our security program," Neil said. "To lose this program would be an enormous setback."

> There's no question it's a runaway success. There isn't a single person in the company who thinks the bounty (program) was not successful.
>
> **- NEIL MATATALL, SECURITY ENGINEER AT GITHUB**

YOUR SECURITY TEAM

VS

BUG BOUNTY TEAM + YOUR TEAM

...Then Quickly Illuminated Security Blind Spots

While a well-rounded approach to security is crucial, it's always difficult to cover everything. GitHub recognized their gaps and saw bounties as a way to fill those gaps.

"Bug bounties are really good at pointing out our blind spots," said Neil. "When we use third parties, we ask them to focus on a specific area. With bounties, researchers look at anything and everything."

And, they're looking in ways and places that may not be obvious, even to the people who built the products in question.

"Our developers are frequently blown away by the ingenuity of researchers," Neil added.

Security teams frequently focus on newly released features or products, so having a wide field of hackers to look at seldom used or older features helps find vulnerabilities regardless of where they live. But since the attention goes to new code, hackers tend to follow. That helps GitHub find bugs faster.

"Hackers also take a pass at new features as they're released," added Neil. "This ensures that the amount of time a bug exists will be shorter than it would've been without a bounty program."

A Bounty Process for Everyone

HackerOne's bug bounty platform doesn't just benefit the security team, and GitHub recognized that early on. Facilitating communications with hackers, paying in foreign currency to a hacker in a far-off location, or keeping records for eventual audits are all tiny slices of the larger HackerOne pie.

"Using HackerOne saves our security team a large amount of time, but more importantly, it also saves our finance team a lot of trouble," said Neil. "Moving to the HackerOne platform allowed us to automate away all of the financial burdens, which are significant."

It's also helped streamline GitHub's overall bug bounty process. In the past, the process of triaging bug reports was mostly manual, so they created a checklist with nearly 20 points to ensure each bug was properly vetted and the hacker was rewarded. Some involved copying and pasting, others required entering the same data in multiple systems.

Using HackerOne as a rallying point, the team automated most of the process. Now, their checklist is down to 4 items.

PRO TIP:

GitHub's advice for those considering a bounty program:

- **Plan!** Make sure your team is on the same page with timing, scope, and process.
- **Budget!** Be prepared to pay when the bugs start flowing in.
- **Voice!** Consider how you want to be perceived by hackers.
- **Humanity!** Don't be a jerk.

ARE YOU READY FOR A BUG BOUNTY PROGRAM?

Click here to see if you're ready.

Automation via HackerOne's API

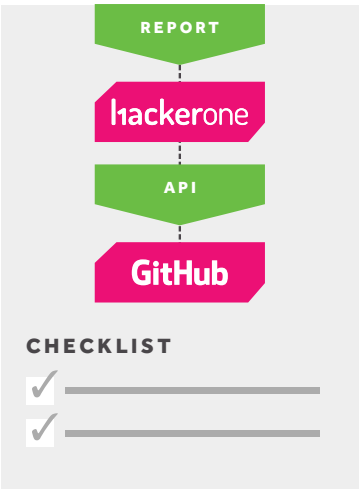
Much of the automation GitHub used to reduce their incoming bug checklist from 20 items to 4 was based on the HackerOne API.

"With the APIs, we create our internal tracking issue for the valid bug, fill out all information, apply a coupon, invite the researcher to the GitHub Bounty org, add them to the "bountyhunters" team, and creates a draft blog post for bounty.github.com," Neil explained. "In the future, we hope to add automatically awarding the researcher with money and swag."

Automating the process not only reduces workload, it turns a once tedious and dreadful task into a streamlined and fast process that eliminates missed items and speeds rewards to hackers. And keeping hackers happy is a great benefit of the automation enabled by the API.

GitHub also recently added automation that takes a HackerOne report and triages it, creating an internal issue for the responsible team with a copy of the report and marking the HackerOne report as triaged. It's just another way the API has helped GitHub eliminate more manual work.

BEFORE HACKERONE



AFTER HACKERONE

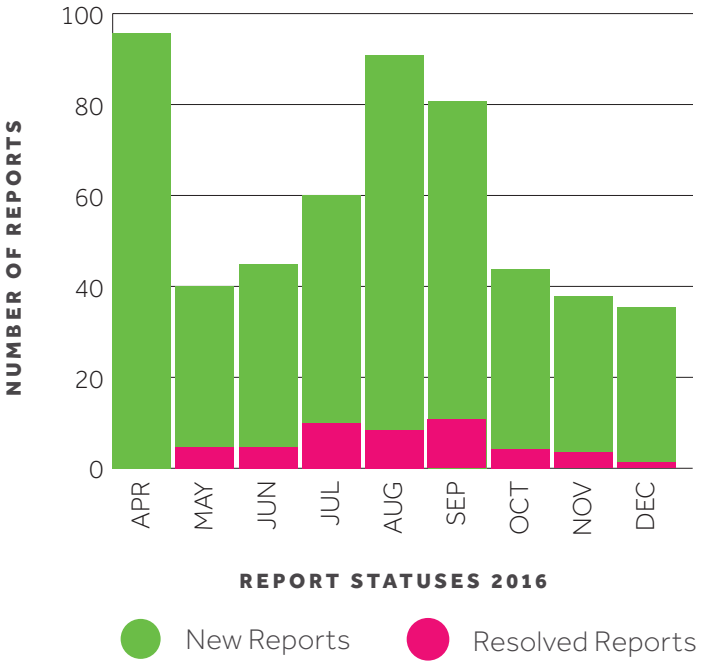
GET THE SCOOP ON TRIAGE →

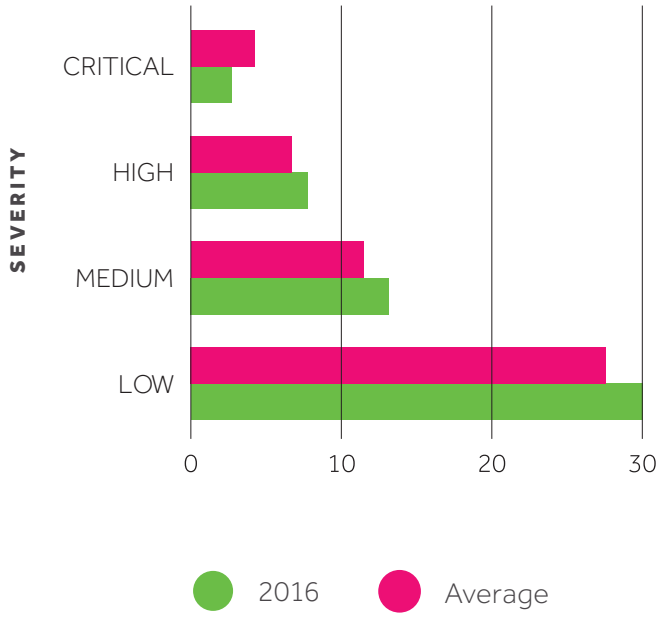
Success is Obvious

GitHub isn't afraid to share metrics around their bounty program. In fact, as of March, 2017, they've shared data on the more than $80,000 they've paid in the past year for 73 submissions. For valid bugs, they've found 48 out of 795 reports for a validity rate of 6%.

Using HackerOne has made it easier to track metrics, but metrics aren't the end all, be all of Github's program. Instead, they see their bounty program as a core part of their security program, and that's critical to scaling their team.

"I don't think quantifying this data is useful," said Neil. "We focus on preventing bugs from happening. This is where the bounty plays a crucial role. The amount of bug hunting we personally do is fairly low and does not scale well. This is why the bounty is a critical piece in scaling our team. It doesn't fully replace manual assessment work, but it certainly complements it nicely."





Pro Tip: Get Creative to Build Buzz and Guide Hackers

GitHub recently celebrated the third anniversary of their bug bounty program with their "More Bugs, Bigger Bounties" promotion designed to generate renewed interest in their program. For two months, they added bonus rewards up to $12,000 on top of the existing bounties! They also pulled in their marketing team to generate more buzz and increase the fun factor.

"Marketing designed limited-edition t-shirts and new graphics, and planned a stream of blog posts and tweets," Neil explained. "We tied it to our three-year anniversary and it just took off."

By including a $5,000 prize for "best report," the promotion also helped GitHub increase the quality and clarity of incoming bug reports.

"We hoped to send a signal to the broader community that well-written reports are appreciated," added Neil. "Dealing with reports that are hard to follow takes a lot of energy. Good reports are ones that provide step-by-step instructions on how to reproduce the issue. Reports with proofs-of-concept are even better."

The big winner of the $12,000 first prize was @jkakavas with their GitHub Enterprise SAML authentication bypass report. The $5,000 prize for "Best Report" went to @orangetw for their report of Server-Side Request Forgery. Additional winners and results of their promotion and overall program were highlighted in this blog post.

Reaping the Rewards

While GitHub considers their bounty program indispensable, the question of "value" eventually arises. Whether it's a finance team asking for justification, a CISO asking for results, or HackerOne asking for this case study, ROI makes the business world go 'round.

For GitHub, however, the return on their bounty program is so obvious, there's no need to even calculate it.

"I don't have anything quantitative but the ROI is phenomenal," Neil said. "Financially, a bounty program is cheaper than a full-time employee or a third-party consulting firm, so we'd be spending more money without it."

"There's no question it's a runaway success. There isn't a single person in the company who thinks the bounty was not successful. Sometimes impostor syndrome makes me question my own value when compared to the bounty if we think of the bounty as an employee."

When pressed, GitHub points to the fact that they've never had to formally justify their bounty program. That, quite possibly, is the best metric of all.

PRO TIP:

Create a "thank you" reward to keep hackers interested and happy!

GitHub gets reports for low-risk issues that may or may not get fixed, but they still pay a small bounty to the hacker. Before HackerOne, it was too cumbersome to pay rewards for every report. Now, it's easy...and for GitHub, it's the right thing to do.

"Hackers get rewarded sooner, have their reputation go up sooner, and know they won't have to follow up with us on resolution status," added Neil. "They can move along while we deal with the resolution. It's our way of saying, 'Thanks, we might not get to this, but we appreciate your work.'"

> The ROI is phenomenal!
>
> **- NEIL MATATALL, SECURITY ENGINEER AT GITHUB**

"There isn't a single person, policy, or practice other than a bug bounty that shows that sort of return," Neil concluded. "There's a general feeling amongst the team and company that the bounty is phenomenally successful while the cost is less than an FTE."

The Best Part? The Bugs!

When asked of the most surprising result of their bounty program, Neil had one, blatantly obvious word: "Bugs!"

Again, the ingenuity of the hackers makes this possible, and is why bounty programs are so successful. It would be financially untenable to outfit any security team with the breadth and depth of skills and experience present in the hacker community. Furthermore, hackers are just plain clever.

"It's been everything from blatant mistakes that are embarrassing to bugs that are so deep and difficult to exploit, they've been fascinating," said Neil. "Every once in awhile we get a really juicy one. Each bug is a learning opportunity and each postmortem reveals an improvement. These extra juicy bugs have led to changes in our processes, tooling, the frameworks themselves, or general understanding of our security posture."

And that's exactly what bounty programs are best at:
making your technology more secure!

hackerone

TRUSTED BY



...AND OVER 700 OTHER COMPANIES