MAPBOX: ONE YEAR PUBLIC ON HACKERONE

Written by Alex Ulsh, Information Security Engineer at Mapbox

Alex Ulsh runs the bug bounty program at Mapbox, the hot startup called a foundational infrastructure for the next stage of the Information Revolution by SoftBank. They've grown their security from a simple vulnerability disclosure policy and security@mapbox.com to a robust and competitive bug bounty program. Read their story here.

Last month was our one year anniversary of launching publicly on HackerOne and our **two year anniversary of running a bug bounty program**. In that time, we've learned a lot about running a bug bounty program and thought we'd share some best practices we've discovered, as well as a recap of the past year.

History of The Mapbox Bug Bounty Program

We first launched our bug bounty program with a single security@mapbox.com inbox in March 2015. That same week we also unveiled www.mapbox.com/security. At first, we managed and triaged reports via email then awarded bounties manually via Paypal. We quickly found this process burdensome and signed up for a private program on HackerOne, a hosted bug bounty service, in March 2015.

We chose HackerOne as it not only connected us to an existing community of seasoned security researchers but also offered productivity features that automated aspects of the bug bounty triage process. By allowing us to start off with a private program and slowly invite more researchers, HackerOne also made it possible to scale our bug bounty program as we ramped up our security team. After a year as a private program, we reached 100 invited security researchers and decided to go public on March 1st, 2016.

The Years in Numbers

Last year we blogged some metrics from our first year as a private program. Here's an update with this year's (March 1st, 2016-March 1st, 2017) numbers compared to last year's.

Metric First year (private) Second year (public) Total reports 408 311 Valid reports 46 48 Percentage of valid reports 11.27% 15.43% Money awarded $22,091 $19,800 Average bounty $458 $521 Largest bounty $2,000 $1,500 Average response time 5 days 2 days Average resolution time 16 days 2 months.

Looking at our numbers over the past two years, we found three trends—reduction in noise, increase in average bounty amount, and decrease in average response time.

Reducing Noise

Though we received 97 fewer reports our second year, the proportion of valid reports rose slightly from 11% to 15%. Invalid reports are considered noise, and we have three primary methods of decreasing them:

- a strict minimum signal requirement
- our HackerOne program page
- common responses linked to triggers with warning dialogues

Strict Signal Requirement

Our first line of defense is a strict minimum signal requirement, a new feature that HackerOne shipped shortly after we first went public. The strict signal requirement means that researchers must have 1.0 or greater signal points to submit reports to our program. Researchers who just signed up for HackerOne and haven't yet built up their reputation and signal are still allowed a few trial reports.

A strict signal requirement was invaluable in the first month after we went public. It allowed us to spend less time responding to noise and more time triaging and fixing valid reports. Now that our program has matured, we're considering lowering our signal requirement this year to take advantage of the work of less experienced researchers.

Detailed Program Page

In addition to stating our program rules and expectations, we maintain detailed lists of what's ineligible or out of scope on our HackerOne program page. These lists deter noisy reports and steer researchers towards high-quality eligible reports. We frequently update these lists whenever we receive new issues that are ineligible but not listed. We also don't penalize researchers who submit ineligible reports that are not already listed on our program page.

What should go on your list of ineligible issues? Anything that researchers think is a security issue but is actually a feature. You should also add any low priority security issues that generate a lot of duplicate reports and noise but may not be immediately actionable, such as missing security headers. You should **not** list any known and unresolved security issues that could be actively exploited—instead, a member of your security team should submit a HackerOne report to your own program. If a security researcher reports the same issue, you can then mark it as a duplicate and invite them as a participant to the original report.

For example, we used to get a lot of reports about HTML and CSS in map markers in mapbox.js. **This is actually a feature of mapbox.js!** We want people to be able to add rich HTML content to their map markers. We don't want them to be able to add JavaScript to their map markers though, and we use sanitize-caja to mitigate this. Reports about the ability to execute JavaScript in mapbox.js map markers **are eligible** and encouraged (though reports about the newer Mapbox GL JS library are even more encouraged)! To reduce noise, we added this issue and a brief explanation to our list of ineligible reports and known issues on our HackerOne program page.

Common Responses and Triggers

Our final defense against noise—if someone slips through the signal requirement and does not closely read our program page—are numerous common responses linked to triggers. We have created common responses with triggers for most of the ineligible issues on our program page as well as phrases from common automated reports.

For example, we have a response that status.mapbox.com is out of scope and that researchers should report those issues to StatusPage.io's bug bounty program instead. We then assign this saved response to a trigger that fires whenever a report contains the text "status.mapbox.com." This trigger shows a warning dialogue—HackerOne uses the term interstitial—if any researcher tries to submit a report about status.mapbox.com. Warning the researcher before they submit the report not only saves both parties time and frustration, but also helps the researcher protect their HackerOne reputation, as ineligible reports are marked as N/A after submission and result in a loss of five reputation points.

Higher Average Bounties

Though we paid $2,291 less in total bounties this year, our average bounty increased slightly from $458 to $521. This could be due to a higher quality or severity of reports, or it could be an indication that our security team is feeling more generous.

HackerOne released a new severity level feature last October that uses the CVSS (Common Vulnerability Scoring System) standard. As a result, we don't have complete data on severity levels for the past year. We also found that researchers tend to overestimate the severity levels of the reports they submit. We're hoping to adjust or assign severity levels for all 2017 HackerOne reports so that we'll have a complete data set for March 2017 through March 2018.

During our first year and for most of our second year, we awarded bounties after issue resolution and confirmation of the fix. Depending on the complexity of the report, this sometimes meant security researchers had to wait a long time between reporting the issue and receiving a bounty. This year we are moving towards awarding a bounty on issue triage. While this means we can accidentally under or over estimate the bounty amount (sometimes you don't know the true impact of a security report until it's 100% fixed), we believe the benefits in awarding bounties earlier outweigh the risks. Paying a bounty on triage rather than on fix leads to better researcher engagement and satisfaction within bug bounty programs.

Improvements to Response Time

We improved our average first response time for reports from 5 days to 2 days in the past year. In the past three months, we've been able to decrease our first response time even further down to 19 hours! This decrease is the direct result of the Mapbox security team overhauling our Incident Response Framework (IRF) for how we triage and handle security issues, including HackerOne reports. What we did to improve and overhaul our IRF process is worthy of its own dedicated blog post—expect another blog post about it next month!

Over the last year, our average resolution time has increased for various reasons, including greater average report complexity and higher average report impact. Our current 2 month resolution time puts us on par with other HackerOne programs like GitHub and Yelp. We're excited to work on improving our average resolution time in the coming months, and we have a few strategies in mind to achieve this. Our switch to rewarding on triage is an immediate way we can strengthen researcher engagement and satisfaction while we work to decrease average resolution time.

New Security Bulletins Page

We're also excited to announce our new Mapbox Security Bulletins page. We launched this page earlier in March 2017 as part of our security IRF overhaul. We'll be posting security announcements related to Mapbox software and our platform, including updates on the impact of future third party security incidents like Cloudbleed or Heartbleed.

What's Next?

Are you a security researcher? Sign up on HackerOne today and start hacking Mapbox Studio, our SDKs, our APIs, and our public website. We're particularly interested in security reports related to our SDKs and our APIs.

Do you run a bug bounty program or are interested in starting one? Connect with me on Twitter at @AlexUlsh and I'll be happy to chat bug bounty programs with you.

HackerOne Has Vetted Hackers For Hundreds of Organizations Including:

GM Lufthansa European Commission

Spotify HACK THE U.S. AIR FORCE airbnb Square LendingClub

HACK THE ARMY Google Play New Relic. Nintendo® Adobe™

Dropbox YAHOO! coinbase

shopify slack yelp

With Over 900 Customers, More Companies Trust HackerOne Than Any Other Vendor