What to consider when launching a bug bounty program [Part 3]

BY ANNA HAMMOND · MAY 11, 2022 · LAST UPDATED ON JULY 14, 2025

If you're not sure what the 3 key stages of setting up a bug bounty program are, or how best to prepare for launching one, take a look at those first two articles in this four-part series on the bug bounty process.

Today, we'll consider what happens when you've done your preparation. You should have run through your bug bounty program checklist. Now, it's likely that your finger is trembling over that big, shiny "launch program" button. Go ahead and click it! The following will help you successfully navigate what happens next.

Note: This is the third in a series of four articles on "What to expect from the bug bounty process, from setting up to post-launch."

Ready to click the 'launch' button on your bug bounty program?

7 things to consider when launching a bug bounty program

Your bug bounty program is live—congratulations! But remember, you're only at the beginning. Unlike pentests or other snapshot-in-time cybersecurity methods, what comes next will be an ongoing process.

Following these seven steps will guarantee your bug bounty process runs smoothly over time. (We'll get into the nitty gritty of bug bounty program optimization in article 4 in this series).

1. Consider the right person for the job of first contact

If you run your bug bounty program on a dedicated platform like Intigriti, operational overheads are low, allowing security teams to focus on other priorities.

Nonetheless, you'll need to assign a point of contact/first recipient for communications and vulnerability reports from the platform.

This person should have three qualities:

1. Time to process the vulnerability reports without long delays (see point 6 below)
2. A calm and systematic approach to processing things that scream "urgent"
3. Time to read the rest of this article.

Sound familiar? Read on!

2. Consider the value of the vulnerability reports you receive

When the first vulnerability reports come in from your bug bounty program, it is helpful to decide whether you could have caught these threats elsewhere.

For example, what if a security expert has noticed that your software is not up-to-date and patched, and has been able to exploit a security vulnerability that way? Kudos to that ethical hacker, but should you be paying bounties for something that could have been easily fixed *before* going live?

We will look closely at bug bounty program optimization in article 4 in this series, but for now, remember that article 2 in this series has tips on **how to prepare your infrastructure for a bug bounty program**. This includes patching all software and running vulnerability scanners and/or pentests.

You should also bear in mind that if you use Intigriti's bug bounty platform all the reports you receive will already have gone through **a stringent triage process**. This means they will be *in scope, non-duplicates* and *valid.* So if you are receiving reports that don't seem helpful, you should consider modifying your bug bounty program scope.

Intigriti's dashboard tools make monitoring the status and severity of vulnerabilities easy

3. Consider your business objectives

Imagine this scenario: it's your first week of running a bug bounty program, and three valid vulnerability reports come in. You're new to this. It's tempting to panic and start trying to fix everything immediately, but take a breath! Depending on how broad or narrow your bug bounty program's scope definition is, you might be seeing vulnerabilities of very different levels of severity (which you will have defined when you wrote your scope).

Before allocating resources, therefore, look again at the business priorities you defined after reading **article 2 on bug bounty program preparation**. Perhaps you'll discover that while one vulnerability is critical, well-known, and needs immediate developer time to fix, the other two have lower severity ratings. Could you perform your own internal triage based on your business objectives and risk severity to decide when you process them? Maybe your kanban board already has a "Fix Immediately" vs. "Fix this Week" list? Your own internal priorities should help decide where things go.

4. Consider your staff's time and resources

In preparing for your bug bounty program, one of the key steps you'll have taken is to make sure your teams are onboard. Now, as you approach those teams to fix discovered vulnerabilities, you'll see why this was a smart move!

Fixing bugs and vulnerabilities takes time and resources that shouldn't come as a nasty surprise. If you stick to your business priorities (see above) then it should be clear not only what needs to be done but *when* it *should* be done.

Maybe there really is something more important your development team needs to get done today than patching a low severity risk. But what about a high severity alert that arrives on a Friday afternoon? Do you have the right people in place who are ready to work over the weekend to fix this? Solid business priorities and established protocols are essential in making such difficult but important calls.

Thankfully, the triaging provided by a bug bounty platform team will help ensure there are no false flags. A reported high-severity risk really will meet your criterion and will need rapid attention.

5. Consider vulnerabilities as opportunities, not failures

Remember—and make clear to your teams—that even the very best cybersecurity defenses will see vulnerability reports when they are subjected to a bug bounty program. Companies as large and as well-protected as Intel use bug bounty programs.They do so because maintaining good cybersecurity is always an ongoing process.

Bug bounty programs are about galvanizing your attack surface and learning from the experience. As you run your program over time, your cybersecurity gets continuously tighter and your knowledge of how hackers think will grow.

6. Consider the security experts working on your program

The crowdsourced ethical hackers working on your bug bounty program are human. This means you can get to know them directly and build relationships. Communication tools in your bug bounty platform can make this frictionless.

One of the most important steps to take in building these relationships is to respond to vulnerability report submissions promptly. This not only validates the work done by the security researcher, but ensures they are paid in a timely manner. If there are delays, let the researchers know why. You'll see dividends if you treat researchers like you would your customers, including making your programs more attractive to the crowd. Fairly priced bounties will help in this respect too, obviously.

If you are not hosting your bug bounty program on a platform like Intigriti, you'll need to make sure you pay on time. With Intigriti, your researchers will receive automatic payment as soon as you accept the report. And, let's face it, do you know any humans who don't like to get paid promptly?

Keep up-to-date on what needs your attention with Intigriti's dashboard tools

7. Consider setting up a learning program

Many Intigriti customers have discovered that the vulnerability reports they receive can be used as excellent in-house training materials for development teams and IT/security teams alike. If your developers know about a vulnerability while they are coding, they can bake security directly into their apps.

As Intigriti customer, Showpad's Director of Security, Privacy & Compliance, Bram D'hooghe says:

"The examples we get out of Intigriti are examples we now use in our training towards our engineering team so that they get this information upfront in their development life cycle."

It's efficient, it provides better security, and it helps teams grasp the value of bug bounty programs.

Bug bounty programs are about continuous improvement

In summary, if you're new to bug bounty programs, stay calm as the reports come in and keep exploring the best ways to work with your team on fixes. There is no one size fits all, which is one reason why bug bounty programs are designed to be agile and adaptable.

Working through the seven things to consider above should prove valuable when launching a bug bounty program. But always try to keep in mind that cybersecurity is an ongoing process.

If you want to go deeper, we have a ton of resources available for bug bounty program managers of every level, from detailed articles like making the business case for a program to writing a vulnerability disclosure policy, or how to maximize hacker engagement, to more general content like the ways a bug bounty program can improve your IT security posture. And if you can't find the answer you're looking for in the documentation, Intigriti's real-person customer support is also always on-hand and easy to reach.

Ready to optimize your program?

In the first three articles in this series, we've shown how using a bug bounty platform, like Intigriti, makes setting up, launching and running a program simple, intuitive and highly effective for your cybersecurity. It doesn't require a high level of security expertise to get started and is very affordable when compared to recurring pentests.

There's one final step that we need to cover. Once you're up and running, there will be opportunities for continuously improving your bug bounty program. This fourth article in the series, '*How to optimize your bug bounty program for success*', covers precisely that.

But for now take a moment's pause and celebrate that your cybersecurity is already well on its way to becoming stronger all the time.

Learn more

Intrigued by what you have read? Want to know more about bug bounty programs? Get in touch to request a demo with a member of our team today.