Crafting your bug bounty methodology: A complete guide for beginners

BY BLACKBIRD-EU · NOVEMBER 25, 2024 · LAST UPDATED ON MARCH 6, 2025

Bug bounty hunting can seem overwhelming when you're just starting, especially when you are coming from a non-technical background. And even then, bug bounty (or web security in general) is a vast topic with so much to grasp.

Participating in bug bounties often also means competing along on bug bounty programs where thousands of other hunters are also actively hacking, with some participants having years of experience. In this article, we are going to cover how you as a beginner bug bounty hunter can navigate around the bug bounty world, and craft your methodology to still be able to find unique and undiscovered vulnerabilities.

Let's dive in!

What is a bug bounty methodology?

A bug bounty methodology is your unique approach to a target. This approach is a step-by-step process that should help you find the most number of vulnerabilities. Everyone has his or her unique approach to bug bounty targets.

Having a unique bug bounty methodology is important as it will provide you with an edge over other competing hunters. A unique methodology will also reduce your number of duplicate submissions often because competing bug bounty hunters on the same program will not be able to find the same bugs as you.

Let's take a look at 2 common and simple examples to better understand what a bug bounty methodology is. Hunter A is skilled in reconnaissance and can find interesting targets by leveraging internet search engines like Google & Bing or Shodan & Censys. So he/she prefers programs with wildcard scopes and spends days just gathering useful data.

Hunter B takes an opposite approach that works for him/her. He/she has a background in application development and actually prefers to dive deep into the primary application, read JavaScript files and test every possible feature and functionality that the target provides.

So let's help you provide you with ideas to build your unique bug bounty methodology. But before that, we need to have the right fundamentals first.

Grabbing the fundamentals

Your fundamentals are everything. If you are a beginner, we highly recommend you master the basics. Learn how the internet works, especially HTTP. Most experienced bug bounty hunters also recommend you learn front-end web development, especially HTML & JavaScript.

Next up is understanding web security fundamentals, we recommend you grasp the basics of OWASP's Top 10 vulnerabilities, such as SQL Injections, Cross-Site Scripting (XSS) vulnerabilities, etc. Dive into multiple resources, watch instructional videos, try to solve CTF challenges and practice in labs that mimic real-world scenarios.

We have an official YouTube channel where we publish videos for bug bounty beginners, such as XSS in 100 seconds:

We also have written guides that you can learn from on our blog, such as Exploiting advanced SSRF vulnerabilities.

You should also not neglect your tooling, learn how to use your proxy intercepting tool and web browser to your advantage. You will be spending a lot of time with these tools, and learning the ins and outs will certainly help you in your testing.

Identifying your strongest points

This is by far the most important part as it will define your starting point. Identifying your strongest points will also help you to double down on them and set you apart from other competing hunters.

So you must identify your strongest points to dive deeper into a certain web vulnerability class, technology stack or target type. Most experienced hunters often only put their focus on a small set of vulnerability classes or a certain type of target (such as e-commerce websites), this approach ensures that you can easily build experience to find certain types of vulnerabilities and also help you master them more quickly.

For example, if you are proficient in front-end web development, it puts you in a unique advantage to focus on client-side web security vulnerabilities (such as XSS). If you have previously worked with databases, you might already have experience in detecting SQL injection vulnerabilities.

If you are comfortable with e-commerce websites, you can put your focus on a subset of vulnerabilities that are commonly present in these types of targets (such as price manipulation vulnerabilities).

Other bug bounty hunters discover that they are skilled in reconnaissance and can easily find various information about a target that often leads to vulnerabilities so they only spend their time hunting on bug bounty programs with huge scopes.

You must identify and select your strongest sides and double down on it, that's by far the best way to start and book quick results.

In case you are unable to identify any of your strongest points, for example, if you come from a non-technical background, we recommend you learn about several vulnerability classes and select the most interesting ones you to double down on.

Continuous practice & refining

This is where you are going to have to hack on several targets and actively search for vulnerabilities in your selected target type. The key here is to spend as much time looking for security vulnerabilities as possible. Ultimately, you'd want to gain experience to quickly catch and recognize patterns that help you

find vulnerabilities. This allows you to easily spot certain types of vulnerabilities on other similar bug bounty programs and targets.

For example, after spending enough time on a certain bug bounty program (such as an e-commerce site), almost only you will easily notice components within your target that are primarily vulnerable to a vulnerability class that others miss. Or you will notice how subdomains often do not have the same amount of security focus as the primary application or target, this will lead you to look for untouched subdomains in lucrative methods that few other competing hunters know of.

If you've primarily targeted software-as-a-service (SaaS) targets, you'll quickly notice that most of your vulnerabilities are found in recently released features. This would allow you to watch closely to change logs and API or product documentation changes.

Continuously practicing and spending time on targets will help you notice several of these small indicators that can help find you vulnerabilities. Indicators that others often miss or simply aren't looking for. As a result, you will also start generating unique ideas with which you approach a target.

Creating an automated and reusable system

Now it's time to create a reusable system that you can deploy on each new target and use at scale. This can be in the form of a checklist that you follow but it can also be an automated tool or script.

If you find yourself repeating the same fixed steps over and over again for a certain task, you can easily develop an automated tool to take back your valuable time.

Checklists can ensure consistency in your testing and help you prevent forgetting about looking for vulnerabilities in certain areas where you'd normally look for them.

8 Additional tips for beginners

Here are 8 additional tips that we can give to beginners to help with crafting a bug bounty methodology that finds you more vulnerabilities:

1) Hunt on a single target for an extended amount of time.

Hunt on a single target for an extended amount of time. Aim for at least 2 weeks before moving to another target. The more time you spend, the more knowledge and experience you gain on your target. This is because you'll start coming across features and functionalities that haven't been extensively tested before.

2) Map out targets on wide-scope programs

Map out targets on wide-scope programs. Especially subdomains and subsidiary assets (provided that they are included in the scope of the program). These assets often receive less security attention and are more prone to being vulnerable.

3) Read the product documentation

Read the product documentation. Learn what is allowed and what behavior should not be possible. This can help you find vulnerabilities such as access control issues or business logic errors.

4) Keep an eye on changes

Keep a close eye on changes. If the target maintains a change log, keep a close eye on it and follow it up daily. Most of the times, new features have most likely not been thoroughly tested for security issues.

5) JavaScript files are goldmines

JavaScript files are goldmines for bug bounty hunters. We recommend you read them and look up for explicit references to interesting links. Keep a close eye on hard-coded credentials as well!

6) Save notes on interesting behavior

Save your notes on interesting behavior. Have you identified a weird response after sending a malicious request? Note it down in your note-taking app for later reference. You never when it can come in handy!

7) Select a program that matches your skill set

Select a program that matches your skill set. If you are more proficient in mobile applications, find programs that explicitly mention these in scope. In case you like performing recon for wide-scope targets, find a program that offers a wildcard scope.

At Intigriti, we have bug bounty programs of all types, sign up today and start hunting on your new favorite program!

8) Test services running on non-standard HTTP ports

Discover and test services running on non-standard HTTP ports (open HTTP ports that are not on port 80 or 443). These often indicate that the service is used for development or administrative purposes!

TIP! You can use tools like Shodan & Censys for finding interesting assets!

Conclusion

Building a solid methodology takes time and practice. Start with the basics and gradually refine and build up your approach based on experience. You can also form a small team and collaborate with other hunters. Learning from each other can help you gain more experience more quickly. For collaboration, we recommend you join our Discord Community, a dedicated place to find like-minded hunters!

If you'd like to hunt on wide-scope programs, or dive deep into SaaS applications, check out our vast amount of programs on our bug bounty platform and who knows, maybe your next bounty is earned with us!

https://intigriti.com/programs

REQUEST A DEMO

intigriti.com/demo

VISIT THE WEBSITE

intigriti.com

GET IN TOUCH

hello@intigriti.com