8 Tips for writing effective bug bounty reports

BY BLACKBIRD-EU · MARCH 25, 2025

So, you've found a valid security vulnerability in one of your bug bounty programs, now it's time to write the report. Finding the vulnerability was half the story. Writing effective reports is also an essential phase in bug bounty.

Clear, well-written, and to-the-point bug bounty reports often get triaged faster and have more chance of getting well received by companies. In this article, you'll learn how to write effective bug bounty reports.

Let's dive in!

Importance of writing clear bug bounty reports

Writing effective bug bounty reports has many advantages, from faster triage time to ensuring the team understands your finding and prevents it from being mishandled. Even more so, a proper bug bounty report is always well-received by bug bounty programs.

It sends a professional signal to the company, making them more inclined to invite you to other private programs! Besides all these benefits, some companies reward researchers for well-written reports with an additional bonus on top of their bounty!

This article will review tips and best practices for writing effective bug bounty reports. By implementing these strategies today, you'll increase your chances of:

- Getting your reports triaged faster and more accurately
- Making a positive impression on companies running bug bounty programs
- Improving your overall report acceptance rate!

Tip 1: Write clear and descriptive report titles

Let's tackle the most crucial aspect of your report and what the company and triager get to see first: your report title. Your report title must always be clear and as descriptive as possible. Avoid sending generic titles such as 'XSS in app.example.com' or 'IDOR vulnerability.'

Your report title must help triagers quickly understand and assess the nature of your vulnerability. Failing to do so will make it harder for them to prioritize your report and look for potential duplicates.

Examples of effective report titles

Let's take a look at a few examples:

POOR REPORT TITLE	EFFECTIVE REPORT TITLE
"XSS in app.example.com"	"Stored XSS in User Profile Comments Allowing Account Takeover"
"IDOR vulnerability"	"IDOR in Payment API Exposes Other Users' Transaction History"
"Authentication Bypass on app.example.com"	"Authentication Bypass via Password Reset Functionality"

Tip 2: Always include as much information as possible

As straightforward as it may seem, most reports still lack information for the triager to reproduce the issue quickly. Never assume that the triager is familiar with your target. Even if you've been hunting on this program for months, the person on the other side reviewing your report might be new to the team or unfamiliar with the specific target on which you've found a vulnerability.

Furthermore, when directly reporting to the company, your report may get reviewed by a non-technical employee. Explaining the weakness and why it matters in simple terms is crucial.

Document your findings as clearly as possible and always include essential information that will help the triager, such as:

- Precise location (e.g. IP address, hostname, domain or subdomain of the affected system)
- Request/response data (always mention all required request headers to reproduce the issue)
- Environment details (include information about your browser version, operating system, mobile device and/or network conditions whenever applicable)
- Account or other configuration information that's required to reproduce the vulnerability.

Add mitigation advice

Adding mitigation steps is recommended to help resolve the issue faster. Remediation advice is specifically helpful for the company when dealing with more complex security vulnerabilities such as web cache poisoning, business logic issues or HTTP request smuggling. Offering informed suggestions demonstrates your expertise and involvement and can speed up remediation.

TIP! When reporting vulnerabilities on bug bounty programs with wildcard scopes, always include proof of the vulnerable target that it is owned by the company maintaining the bug bounty program! Proof can consist of WHOIS data, SSL certificate logs or network information from where the IP address originates!

Tip 3: Write clear and detailed reproduction steps

Your reproduction steps must be clear and to the point. These steps will help the triager verify the underlying vulnerability quickly while also playing a vital role in getting your submissions triaged faster.

As mentioned, never assume the triager is familiar with your target. A rule of thumb is to write your reproduction steps as if the person reading has never seen the target.

Format for clarity

When writing reproduction steps, try to use a single line per instruction. Each step must help the triager to follow along and reproduce your reported issue quickly.

Make sure to mention any prerequisites upfront, such as test accounts with specific rights.

Include supporting materials

Try to include supporting materials that help reproduce the reported vulnerability. Clear images documenting each step or video recordings can help the triager save time when processing your report!

Tip 4: Provide a working proof of concept

Some researchers submit reports with non-working proof of concepts. Submissions like these create difficulties for the triager as they won't be able to reproduce your vulnerability. Try also to mention if there are any special requirements for your payload to work (e.g. XSS that only works on Firefox).

Include necessary material for the proof of concept to work, e.g. if you're reporting an XXE injection vulnerability, mention your malicious DTD file. Taking these extra steps while ensuring your proof of concept works consistently will help triagers validate your reported issue quickly.

TIP! Video proof of concepts can save a lot of time & effort needed in explaining a vulnerability to developers or non-technical people!

Tip 5: State a clear impact

The impact section must always state a clear impact caused by the vulnerability, especially when dealing with more complex vulnerabilities. A well-articulated impact will also help the triager and company to assess the severity of your finding correctly!

Most reports that fail to mention a clear impact are often closed as 'informative' or 'Not Applicable'. Try to be as specific and realistic as possible. Avoid overestimating the impact of a particular finding while also not downplaying serious risks.

Try also to state what the intended or expected result should be. For instance, mentioning the correct outcome would be helpful if you're reporting a business logic error or an insecure direct object reference (IDOR) vulnerability.

Tip 6: Use a proper report format

Most reporting forms, such as Intigriti's submission form, support Markdown. Take advantage of this and make use of Markdown to structure your report. Use headings to separate your report into sections, code blocks to highlight payloads, and image tags to include inline images. An organized and well-structured report is more straightforward to process and read, often leading to faster triage time.

Keep in mind that some bug bounty programs require you to follow a specific format, make sure you do so, as it can help speed up the triage time.

Tip 7: Review your submission

Before hitting that submit button, take time to review your report. A thorough review ensures that you're submitting a valid report.

Check that you've included a working proof of concept. Make sure also to cross-check if you've included all the necessary details. Reread your reproduction steps and ensure you did not miss out on documenting any essential instructions.

Common pitfalls to watch for

Missing context

Have you included all the necessary information about your findings? Some information that seems obvious to you might not be to the triager or the company. Try to be as straightforward as possible.

Scope confirmation

Always verify that the reported finding is in-scope. Double-check the IP or affected host, and include SSL logs or WHOIS data whenever applicable.

Proof of concept

Cross-check that the included proof of concept works and has no typos, incorrect URLs or any left-out details or instruction steps.

Taking the extra time to review your submission can help you spot minor issues before submitting the final version.

Tip 8: Maintain a professional tone throughout the triage process

Your submission isn't the end, how you communicate during the triage process is also essential. Always maintain a professional tone. Show patience and try to answer triager or company feedback whenever possible. Your professional tone also builds a positive relationship with the company.

At Intigriti, we value hackers like you and work closely to handle every incoming submission carefully. We've also integrated a 'Request Support' form to give your report a second consideration when you believe the triager or company handled your submission wrongfully.

Conclusion

Finding bugs is only half the story, writing clear and actionable bug bounty reports is crucial as it can help the company understand your submission. Writing effective bug bounty reports also comes with many benefits. From getting your reports triaged faster to sending a professional signal to the company to increasing your overall report validity rate!

You've just learned how to write compelling bug bounty reports that are effective, actionable and to the point... Right now, it's time to put your skills to the test! Why not browse through our 70+ public bug bounty programs on Intigriti and submit your following findings on our platform!

START HACKING ON INTIGRITI TODAY