Inspectiv

Strengthening Your Core Security Program with Comprehensive Testing

Executive Summary

Strengthening cybersecurity defenses is like building physical strength overall - it takes commitment, consistency, and focus. Bug bounty programs deliver the same opportunity to get stronger, helping organizations fortify their cybersecurity defenses in key areas such as vulnerability detection, prioritization, and remediation.

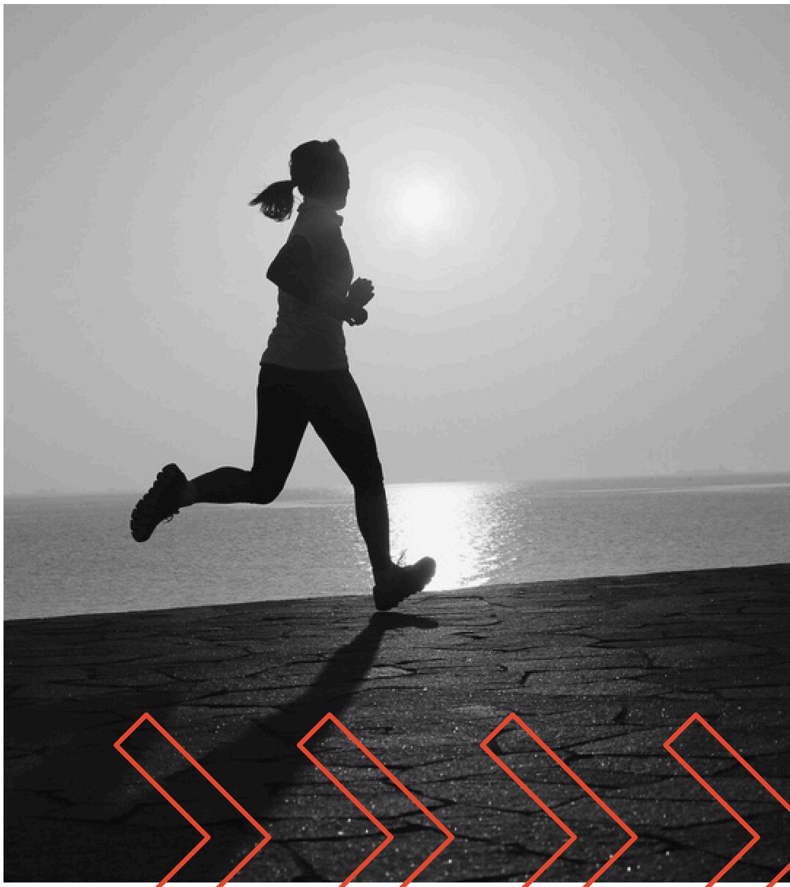
Similar to training different muscle groups, these programs uncover a wide range of vulnerabilities from low-severity findings to impactful, exploitable issues needing immediate attention.

The real progress, however, comes not just from the number of "reps" (# of opportunities to build strength) but from the discipline and speed with which validated vulnerabilities are prioritized and remediated ("form").

For most customers, "reps" are no problem. With the right scope, Inspectiv's bug bounty programs consistently uncover tangible evidence of security gaps by finding tangible evidence that uncover new vulnerabilities.

Inspectiv's Account Management team provides ongoing support to keep customer programs well-scoped and carefully triaged. Unlike static, company-wide bug bounty programs, this keeps researcher attention where you want, and when you want it. As a result, the vulnerabilities that are found are rapidly highlighted for the next remediation (or compensating control) step.

Remediation alone is not enough. By leveraging bug bounty as a core discipline, organizations can elevate their defenses and build enduring cybersecurity strength. Here's how.



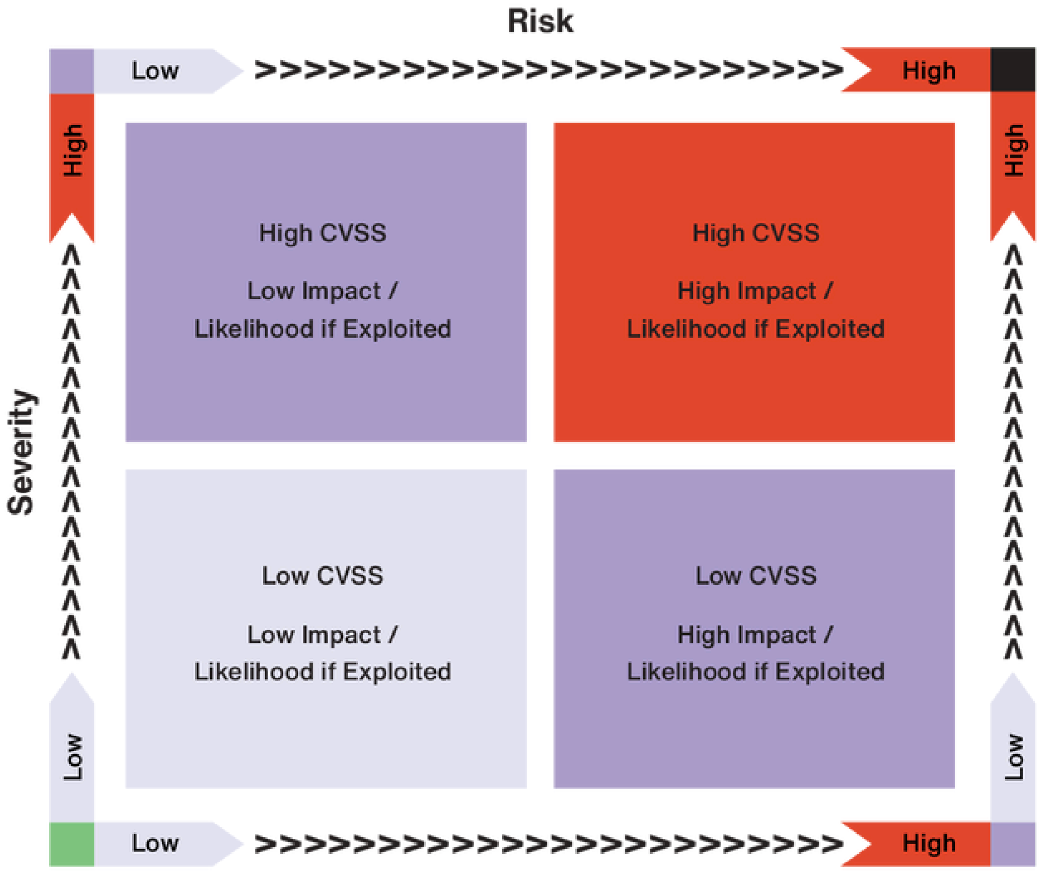
Plan your Workout; Workout your Plan

When bug bounty platforms launched early in the 2010s, it was a battle of humans vs. humans - adversaries vs. defenders - with the introduction of ethical hackers providing a new advantage for blue teams. However, the recent rise of AI-assisted coding, “vibe coding,” and low-code platforms has accelerated development cycles, with companies producing more applications, faster than ever before. The downside: they’re also shipping more vulnerabilities at a pace that strains most security teams.

Traditional vulnerability management programs often rely heavily on severity-based scoring models like CVSS or on raw counts of vulnerabilities. That’s like judging a workout based on one indicator (time, weight, or reps). No single metric tells the whole story. Progress can be measured by looking at them together.

While useful for categorization, measures like CVSS alone cannot be the sole guide for real-world remediation decisions. They provide a picture of risk but ignore exploitability, business exposure, and context.

Simple Risk Analysis Matrix



Risk

Low >>>>>>>>>>>>>>>>>>>>>>>>> High

Severity

High CVSS

Low Impact / Likelihood if Exploited

High CVSS

High Impact / Likelihood if Exploited

Low CVSS

Low Impact / Likelihood if Exploited

Low CVSS

High Impact / Likelihood if Exploited

Bug bounty programs help highlight this gap more than any other process. A substantial share of incoming reports are invalid, duplicative, or speculative. Without skilled triage, organizations would quickly be overwhelmed. Inspectiv triagers absorb this complexity and deliver only validated vulnerabilities to customers, shielding internal teams from wasted cycles and disputes with researchers. The contrast between pay-per-bug and flat-fee models is stark: where the former rewards noise and quantity, the latter emphasizes validated, prioritized, high-value results.

This document explores how a bug bounty program can be more than just another source of vulnerabilities. Vendors, CVE reports, and xAST scanners do an adequate job discovering security vulnerabilities. Most organizations do just fine with those findings before relying on bounties and pen tests to find even more issues.

Instead, a bug bounty program does something that the other vulnerability sources cannot - it can improve the overall defensive strength of an organization. It's a better workout regimen for building cybersecurity muscle. Let's see how.

From Strain to Strength: How Bug Bounty Programs Fortify Security

In a perfect world, an omniscient analyst could evaluate every reported vulnerability and instantly calculate its actual risk: the probability of exploitation multiplied by the impact to the business. In reality, organizations face fragmented information, multiple ways to measure risk (some mandated), and constant shifts in business priorities. As a result, most organizations continue to rely on severity scores as a proxy for risk,much like using a single fitness metric such as BMI to represent overall health. Two people, from elite athletes to the average individual could share the same BMI, but those numbers alone reveal nothing about their true fitness.

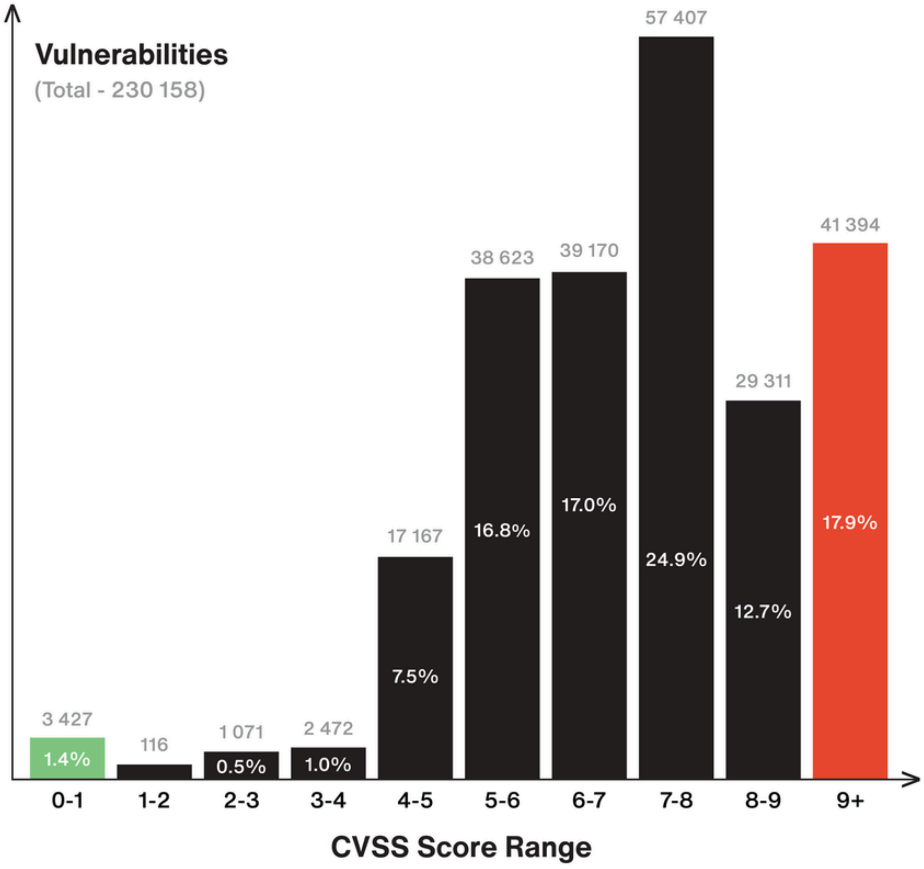
Typical vulnerabilities coming in from some sources, especially DAST scanners, are often low priority with low probability of exploitability. Compensating controls or even advances in general software engineering tend to make vulnerabilities less severe over time. A simple example would be an application subject to out-of-memory bounds exploitation in the world before and after virtualization and containerization. The same code running on bare iron could be highly concerning, but much less so in better memory-protected environments of today. Remember, DAST should find every vulnerability ever recorded, from any year.

Bug bounty programs generate vulnerabilities of any severity, and at unpredictable intervals. This requires thoughtful triage to happen at any time, to see if a vulnerability is risky or not. Further, this has to be tempered with risk, not just a passthrough of severity to determine how to prioritize a fix.

Inspectiv triagers understand this and do not rely on CVSS only to determine if and how to validate a report for a customer. Reports are evaluated based not just on severity, but also on exploitability (is an attack practical or theoretical?), exposure (is the system internet-facing or internal?), and business impact (what data or function would be compromised?). Researcher reputation also plays a role, with proven researchers who consistently deliver accurate and impactful reports receiving faster attention.

The industry is improving. Forward-looking tools like the Exploit Prediction Scoring System (EPSS) provide dynamic insights into the likelihood of exploitation. EPSS allows risk to be tracked over time, raising or lowering the urgency of a vulnerability based on active threat activity. Inspectiv incorporates these tools where appropriate but keeps prioritization anchored in real-world risk.

Distribution of Vulnerabilities by CVSS scores



The National Vulnerability Database (NVD) is the main tracker of reported security vulnerabilities. Organizations can expect bug bounty programs and penetration testing to find vulnerabilities across the severity range.

Many low- and medium-priority vulnerabilities are not reported to the NVD, whereas security testing will report all valid findings.

7.6 Weighted Average CVSS Score

*For CVEs published in the last 10 years
Source: www. cvedetails.com

Bug Bounty Programs Start with Built-In Warmups

Bug bounty programs exhibit recognizable patterns over their lifecycle. In the early stage, there is typically a surge of reports focused on surface-level, low-severity findings such as verbose error messages, information disclosures, or small misconfigurations. This is analogous to weight training leading to few tangible results in the first few days (or weeks) until...they do.

After this light jog of a program, where organizations can expect more clickjacking than RCE (just kidding - who puts clickjacking in scope?), more strenuous vulnerabilities often emerge.

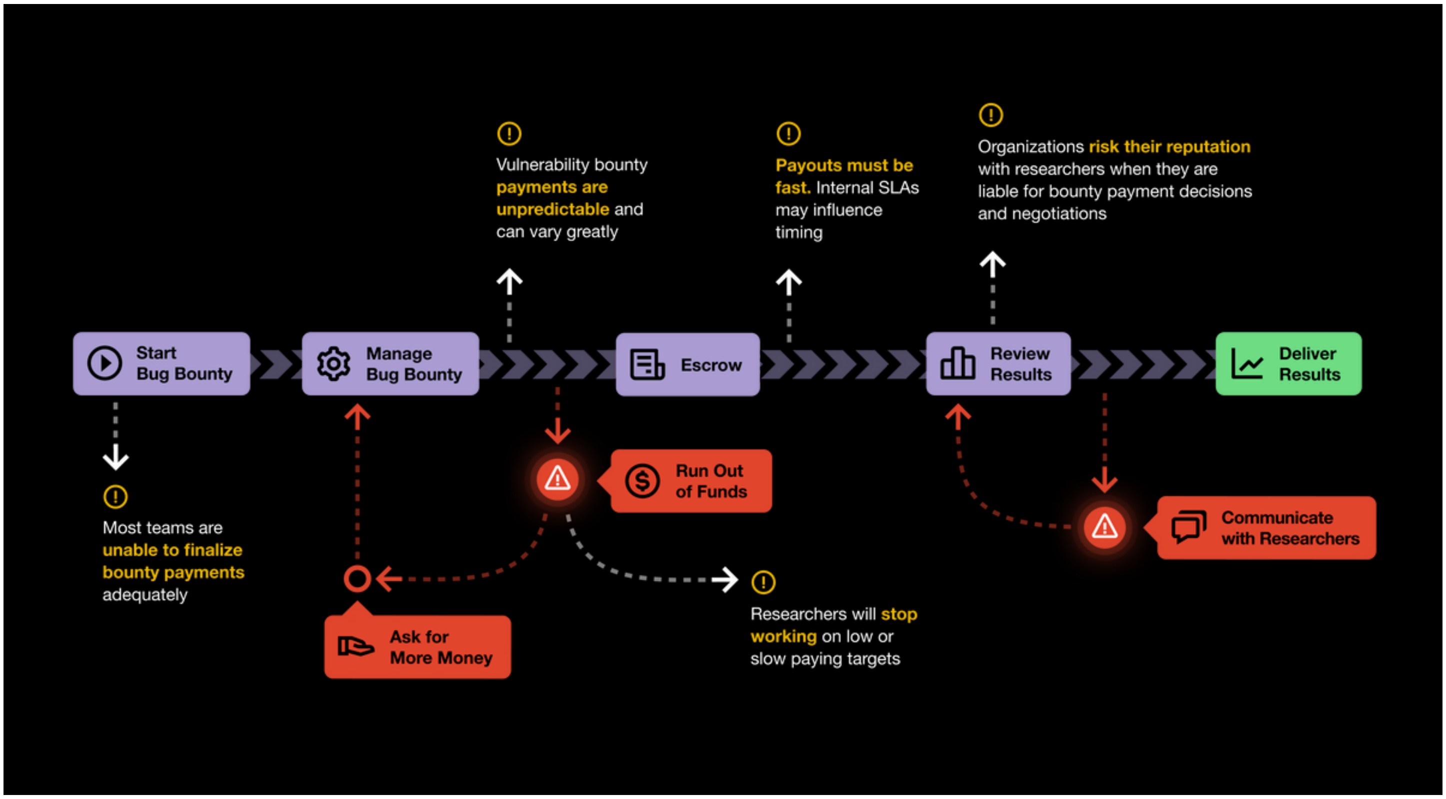
This is because researchers begin to conduct deeper reconnaissance. They have learned the idiosyncrasies of the attack surface and can make good decisions on where to devote their effort. This recon and transition phase can yield more impactful vulnerabilities such as authentication bypasses, RCE, or SQLi. While volume decreases compared to the early flood, the findings often increase in importance.

Over time, a steady state emerges where discoveries are less frequent but more technically sophisticated. At this point, most low-hanging fruit has been addressed. However, bug bounty programs have continued to deliver value due to their earned reputation of finding complex vulnerabilities that automated scanning tools miss.

Finally, spikes in vulnerability submissions occur whenever new attack surfaces are introduced. Examples include deploying a new feature, migrating infrastructure, or acquiring another company. These changes create opportunities for researchers to uncover fresh vulnerabilities.

Invalid and duplicate reports are present throughout all phases. Although they reduce perceived efficiency, they underscore the importance of triage: triagers filter out noise and transform raw submissions into actionable intelligence. For customers, this translates into predictable remediation needs and better resource planning.

Traditional Bug Bounty is Complex and Cumbersome



Vulnerability bounty payments are unpredictable and can vary greatly
Payouts must be fast. Internal SLAs may influence timing
Organizations risk their reputation with researchers when they are liable for bounty payment decisions and negotiations
Start Bug Bounty
Manage Bug Bounty
Escrow
Review Results
Deliver Results
Run Out of Funds
Communicate with Researchers
Most teams are unable to finalize bounty payments adequately
Ask for More Money
Researchers will stop working on low or slow paying targets

What You Do With a Vulnerability is the Real Workout

Now that we've addressed vulnerability finding, there is the true gain - remediation.

Service-level agreements (SLAs) set the pace of remediation much like the protein-fuelled recovery fuels muscle growth. Inspectiv provides numerous features to help organizations apply rigor to their SLAs and strengthen cybersecurity defense muscles. For example, it's easy to set SLAs with real time metrics in hours and days, rather than business hours. After all, attackers operate continuously, and vulnerabilities do not cease to exist over weekends or holidays off.

Inspectiv's default recommended expectations are straightforward: High-risk vulnerabilities remediated within 24 hours, Medium-risk within 7 days, and Low-risk (including informative findings) within 30 days. In reality, low-risk vulnerabilities are rarely addressed at all. Some vulnerabilities require larger teams or involve more coordination to remediate or institute a non-disruptive compensating control, such as microsegmentation.. In the worst-case scenario of an active breach, security teams may be partially diverted to legal and notification requirements. Even then, remediation must remain a priority.

Muscle growth comes from strain + recovery. The same applies for what an organization must do in the face of a potentially serious vulnerability that can literally come in at any time. For example, if a security vulnerability that is severe comes in at 5 PM on a Friday, how is that gonna mesh with your SLAs? If you have one technical expert who's really capable of understanding how to deal with a vulnerability on one part of your infrastructure and that person is on vacation, what do you do?

Because the answer is typically: "Whatever I have to!", organizations' cybersecurity strength gets built from the "regular irregularity" of bug bounty programs' typical output.

Common SLAs from Inspectiv Customers

24	72	07	30
HOURS	HOURS	DAYS	DAYS
CRITICAL	HIGH	MEDIUM	LOW

(Times are from customer-acceptance of a validated vulnerability to a deployed fix. Inspectiv offers free remediation validation, and routinely gets requests more than a year after reporting.)

Good Form and Sticking the Landing - Remediation Workflow Best Practices

Effective remediation starts with clarity. By the time vulnerabilities reach engineering teams, Inspectiv ensures they are processed, reviewed, clarified, and distilled into actionable descriptions. Where appropriate, supplemental graphics or videos are included to ensure understanding. Pen testing and bug bounty programs both offer a way to get vulnerability report clarification from researchers, though bug bounty programs typically offer a longer time period to do so. This matches well with the (dangerously long) year or more timelines sometimes seen for remediation in even the best-run organizations.

Each vulnerability is assigned to a responsible team with defined accountability and SLA deadlines. This prevents vulnerabilities from falling into limbo or being overlooked.

Automation also plays a critical role. Ticketing systems, orchestration, and validation tools reduce overhead while guaranteeing no step is skipped. Customers can therefore focus on applying fixes, knowing the process is streamlined.

The Cooldown — After-Action Reports

Incorporating lessons learned from remediated vulnerabilities is akin to a crucial "cooldown" in a fitness regimen, solidifying gains and preventing future regression.

Bug bounty-sourced vulnerabilities are particularly effective for this. When a program is scoped properly (and Inspectiv helps ensure that they are), they can produce a wide variety of security vulnerabilities that touch on numerous security controls and infrastructure/ software components.

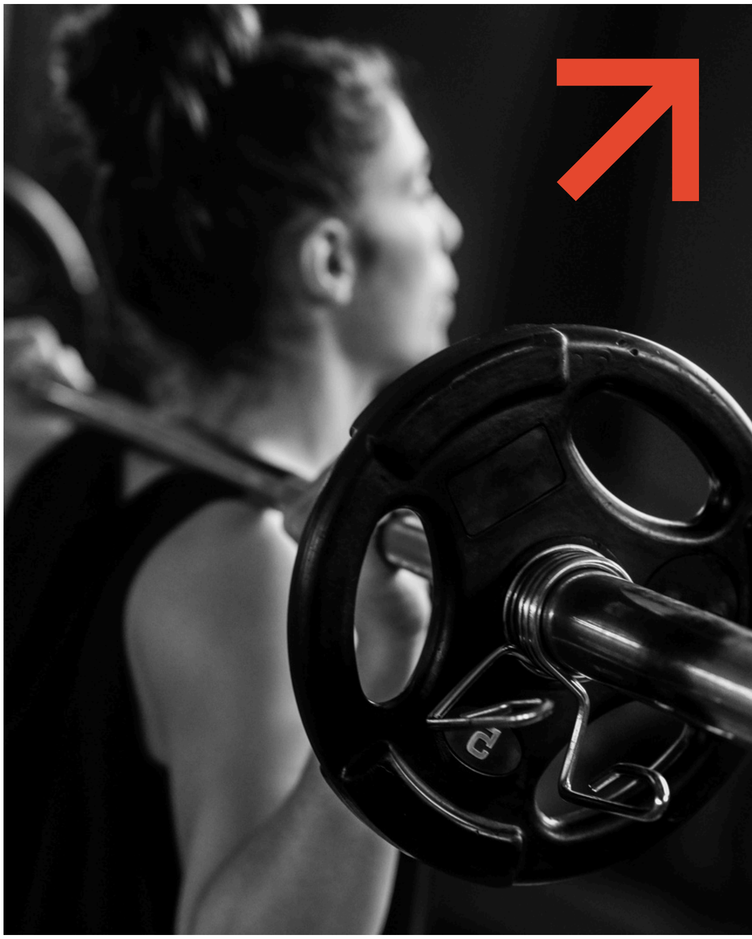
Unlike generic scans that might flag theoretical issues, bug bounty programs deliver real-world, exploitable vulnerabilities that have been validated by human researchers. This provides an organization with tangible evidence of how attackers could compromise their systems, offering invaluable insights into specific attack surfaces and defensive gaps.

Almost every vulnerability results in a thought like “If we only had done X, this vulnerability wouldn’t have been found.” It’s a perfect blueprint to incorporate those lessons into earlier processes - training, software development, DAST, etc. - that make your security stronger for longer.

Maintaining good form without overthinking it

The following rules serve as universal anchors for using bug bounties to strengthen your cybersecurity:

- Fix what's exploitable and public-facing first.
- Prioritize by data sensitivity and threat relevance.
- Assess remediation skillsets upfront — determine in-house vs. consultant support.
- Document exceptions and compliance impacts with ownership and deadlines.
- Track remediation timelines and measure outcomes to refine processes to improve SLAs.



Conclusion

Bug bounty programs provide organizations with continuous assurance against evolving threats. They provide real-world examples of how today's attackers could compromise customers' systems. To settle for “just” finding vulnerabilities is to lose sight of the complete, transformational advantages that can come with a well-developed program. Fixing a vulnerability is one outcome of a found vulnerability, but understanding how it was found can help harden defenses later. Mature organizations can take advantage of all the outputs - vulnerabilities, explanations, metrics - from a bug bounty platform like Inspectiv to drive security improvements throughout the entire organization.

Inspectiv

Inspectiv provides enterprise-class offensive security testing for busy security and software development leaders. Utilizing top researchers from around the world and an AI-enhanced software testing platform, Inspectiv helps organizations find and eliminate vulnerabilities, staying ahead of AI-enabled attackers. Their adaptive approach to pentesting and bug bounty programs helps organizations achieve harder defenses without burdening small teams with triage, payments or validation work. Fixed price options minimize risk and maximize predictability.