bugcrowd

Ultimate Guide to Vulnerability Disclosure

Table of Contents

Everything you need to know about vulnerability disclosure programs

In software development, errors are inevitable—with one report estimating up to **50 errors per 1,000 lines of code.** That means that security vulnerabilities are, quite simply, a fact of life. Taking into consideration that code will be recycled and reused across potentially millions of deployment points, the potential size of the attack surface is staggering.

In this environment, all organizations have to adopt strategies to maintain the trust of stakeholders—organizations must prove that they do everything possible to secure their systems and data.

Vulnerability disclosure programs (VDPs) are now an **industry standard** (and often a required one for compliance reasons) for proving an organization's public commitment to a strong security posture. A complement to bug bounties and penetration testing, VDPs allow anyone on the internet to altruistically report any vulnerability they've found.

All data in this guide are based on a survey of Bugcrowd VDP customers.

This report examines:

- The nuances associated with vulnerabilities discovered "in the wild."
- The basics of VDPs, including key benefits.
- Why the NIST Cybersecurity Framework lists vulnerability disclosure as a requirement for every organization.
- Best practices for implementing and managing a VDP.
- How to combine a VDP with bug bounty programs or penetration testing.

The basics of vulnerabilities

Before diving into the world of VDPs, it’s helpful to understand the **basic characteristics** of the vulnerabilities reported in these programs. Below are some frequently asked questions.

What are vulnerabilities?
Vulnerabilities are components of code that can be exploited to negatively impact the security of data, systems, people, or intellectual property (IP). Vulnerabilities are often referred to as “bugs.”

What causes vulnerabilities?
Vulnerabilities can be the result of erroneous scripting or can arise from changes in the deployment environment or from several seemingly intentional commands combined in unintentional ways.

How common are vulnerabilities?
The average software application reportedly has 15–50 bugs per thousand lines of code.

How are vulnerabilities surfaced?
Most internally developed software progresses through similar development life cycles, which include several phases of targeted testing prior to and throughout production. Unfortunately, it’s impossible to simulate every possible use case, permutation, or potential interaction in such controlled settings. Additionally, software is always evolving—expanding and contracting like a living organism to adapt to new operating environments and an ever-growing list of connected tools and services. This causes vulnerabilities to surface constantly.

Who finds vulnerabilities?

Internal software developers

End users

Hackers

Threat actors

What is a VDP?

Vulnerabilities are inevitable and not a sign of weakness. It’s all about how an organization responds to these vulnerabilities. VDPs may be best described as **the internet’s “neighborhood watch.”** Neighborhood watches, of course, rely on volunteers to monitor their communities for suspicious activity and to report incidents to the police when warranted. In other words, **"if you see something, say something"** applies equally to a VDP.

Like neighborhood watches, VDPs encourage anyone on the internet to be vigilant for the benefit of all. Specifically, they offer a framework for publicly reporting vulnerabilities discovered outside typical testing cycles. As VDPs usually cover all publicly accessible internet-facing assets, anyone with an internet connection can participate in the surfacing of vulnerabilities.

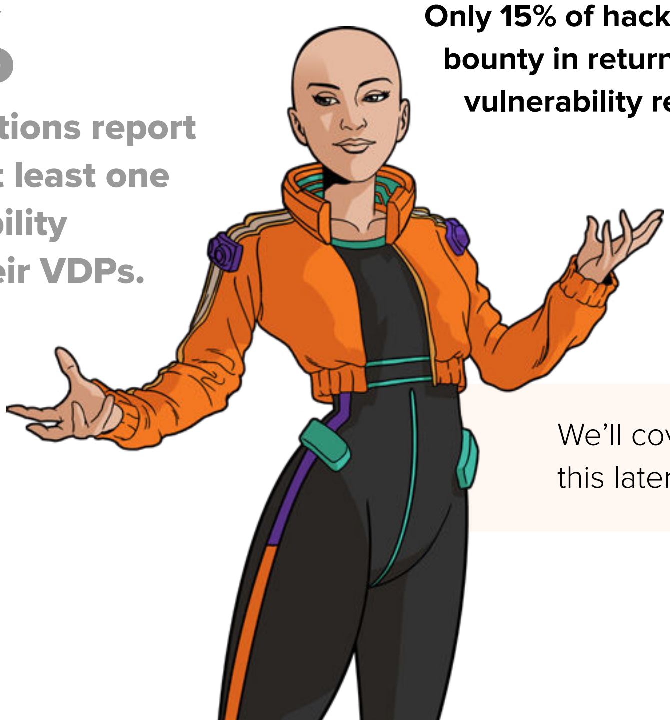
The adoption of a VDP is an acknowledgment that the organization understands the inevitability of vulnerabilities and is committed to security transparency.

It’s worth noting that VDPs are different from bug bounty programs and penetration testing, which provide monetary incentives for discovering critical, in-scope vulnerabilities.

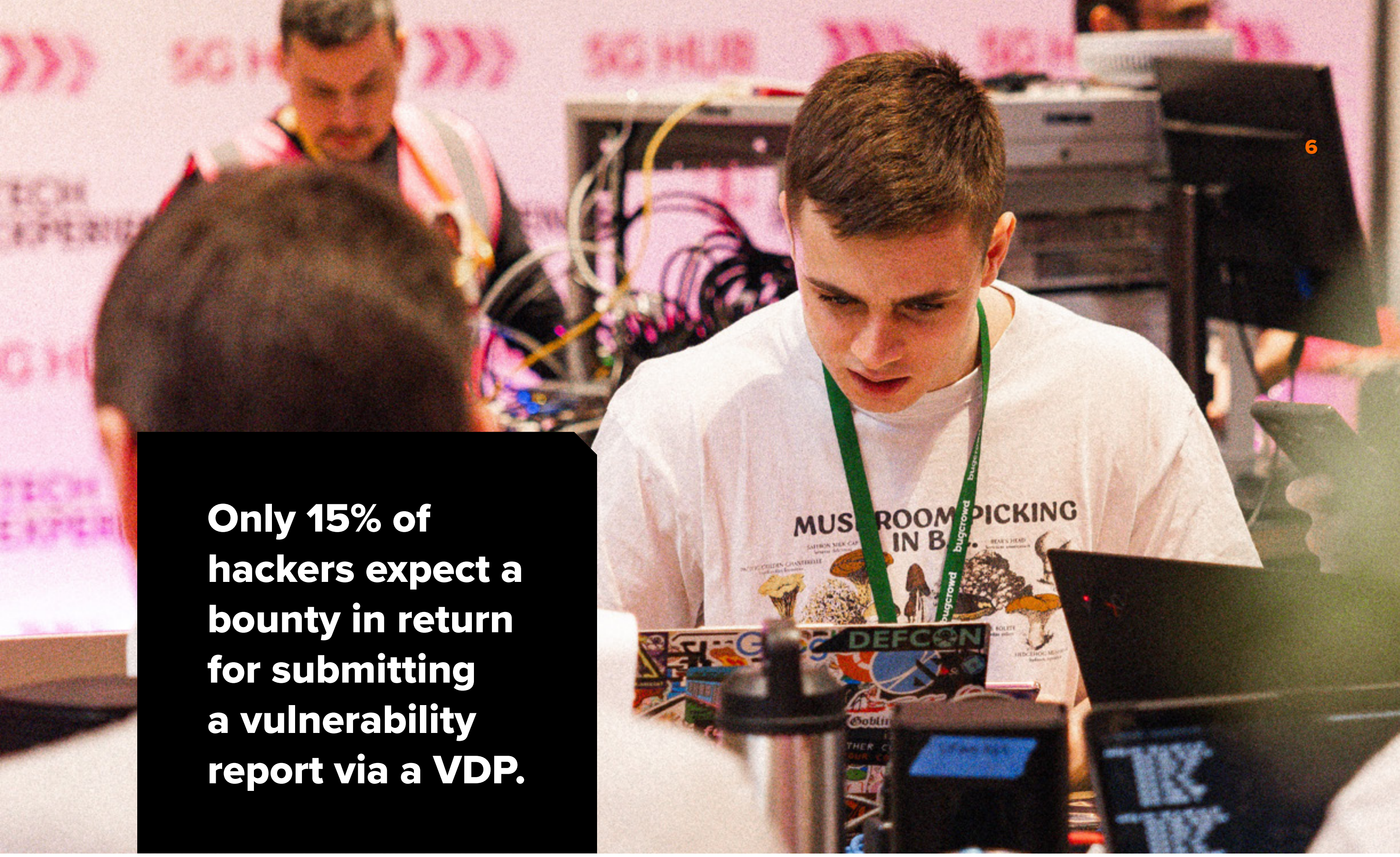
87%

of organizations report receiving at least one P1 vulnerability through their VDPs.

Only 15% of hackers expect a bounty in return for submitting a vulnerability report via a VDP.



We’ll cover more of this later in the guide.



Only 15% of hackers expect a bounty in return for submitting a vulnerability report via a VDP.

The **methods used to manage VDPs** differ by organization and are often dependent on their goals, resources, and bandwidth. While some choose self-management to get started, most rely on third parties like Bugcrowd to monitor intake channels, triage findings, and provide feedback to the submitting party at a speed and scale that most self-managing companies would find difficult.

By enabling the reporting of vulnerabilities found through the routine use or testing of externally-facing products and services, VDPs help organizations **reduce risk** with minimal disruption to existing security and production life cycles.

While many organizations derive great value from highly active VDPs in this fashion, the purpose of a VDP is first and foremost to **provide a secure channel for altruistic, externally sourced security feedback.** Therefore, they complement, but don't replace, bug bounties and pen tests, which are tightly focused on uncovering critical vulnerabilities.

Instead, by offering recognition to well-intentioned hackers who abide by a defined process, VDPs simultaneously build and **enhance an organization's reputation for security,** aka its "security brand."

Key benefits of a VDP

Countless vulnerabilities are being written into new and existing software every day, and organizations need to maximize their ability to discover them. However, per Bugcrowd's research,

58%

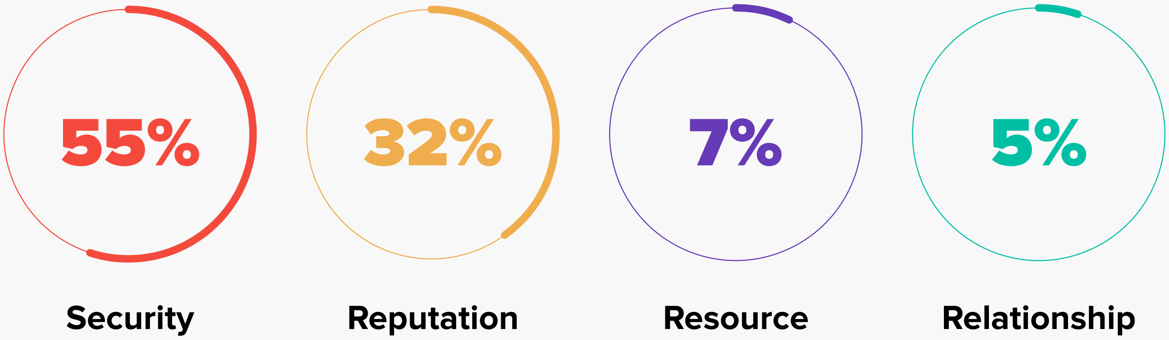
of ethical hackers won't report a vulnerability if the owner of that vulnerability doesn't provide a clear way for doing so.

Let's take a closer look at the idea of **"taking security seriously"** and discuss what this actually entails. This statement can usually be boiled down to a few common goals and priorities:

- ✓ Reduce risk
- ✓ Improve security ROI
- ✓ Accelerate digital transformation
- ✓ Make better decisions on security initiatives
- ✓ Improve security transparency and customer confidence

VDPs help organizations achieve these goals in many different ways. We'll discuss how VDPs on the Bugcrowd Platform specifically support the above security goals later on in this guide.

What do you believe is the main value of your VDP?



Beyond building a stronger security posture, a VDP offers several **key benefits,** according to an organization's customers, partners, investors, and employees, as well as the hacker community.

Customers

Vulnerabilities are an externality that affects end users much more than owners. This means organizations should not only prioritize the security of their users' data for users' sake but also for the prevention of the reputational, and ultimately financial, damage organizations will incur if they fail to do so.

> **A VDP allows companies to reduce risk while publicly showcasing their commitment to security in a way that is both easily understood and easily verified.**

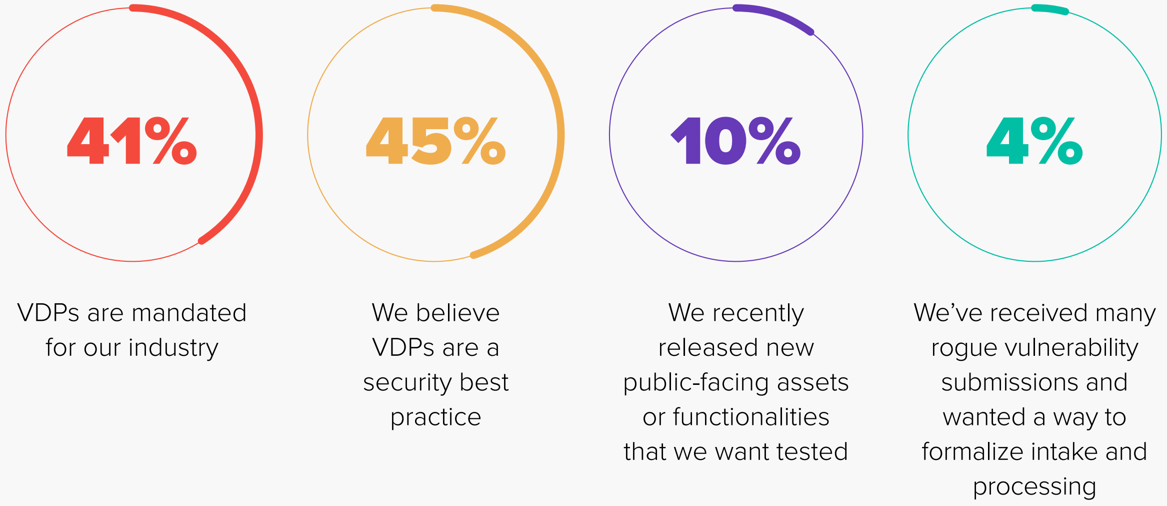
Security Researchers

The VDP halo extends to an organization's overall security brand, acting as a **strong indicator of security posture** for external stakeholders like prospective investors, partners, and other collaborators. These programs are public evidence of an organization's culture of remediation, recognition, respect, and commitment to rapid response. For potential security hires, the presence of a VDP often signifies the influence wielded by security leadership among executive peers in Marketing, Legal, and Sales.

Partners, Investors, and Employees

Any discussion on the impact of VDPs would be incomplete without due attention to the finders of vulnerabilities themselves. VDPs provide emerging hackers with the **opportunity to hone their skills,** while established hackers can build and extend relationships with organizations that may result in private, invite-only engagements like bug bounties. Moreover, both groups benefit from the knowledge that they are incrementally improving an organization's security.

For organizations with a VDP, what is the main reason your organization has implemented a VDP?



Most hackers are motivated by a combination of education, rewards, and recognition. Unfortunately, "recognition" is all too often lumped in with "reward." **Rewards and recognition are both gestures of appreciation but are each rooted in different measures of value. VDP rewards may come in the form of kudos points or swag.**

Recognition provided by a VDP program goes beyond an organization's acknowledgment of a hacker's contributions and instead refers to the ability of a hacker to have their contributions recognized by the broader security community. It is global recognition through disclosure.

79%

of organizations with a VDP have gone beyond awarding points for exceptional findings.

What is Disclosure?

Sharing security vulnerabilities with the world enables organizations to get ahead of threats before they become larger problems. Communicating how and when vulnerabilities were uncovered can drastically reduce the frequency of their creation while improving the ability of hackers to more readily spot related issues. Additionally, according to Bugcrowd's research,

> **Organizations that adopt disclosure terms see 30% more vulnerabilities than organizations that don't.**

"Disclosure" has several meanings, referring to the communication of a vulnerability to the organization within which it was discovered and to external parties, usually in a public forum. While the first definition benefits an organization and, by extension, its direct customers, partners, and other stakeholders, the second, when done right, benefits the entire digitally connected world.

Coordinated disclosure terms emphasize Bugcrowd's definition of good faith in the context of finding and reporting vulnerabilities; they encourage rapid remediation while demonstrating commitment to and appreciation of the hacker community.

However, the term "disclosure" carries an unfortunate and misplaced stigma, which is holding back security standards globally. Many organizations see the disclosure of a vulnerability as an unnecessary admission of weakness that harms their reputation, but this is a short-sighted outlook. A quick exploration of the varying types of disclosure can help clarify terms and alleviate unfounded concerns.

The Spectrum Of Public Disclosure

DISCRETIONARY DISCLOSURE

When organizations opt to enable coordinated disclosure, they signal their openness to considering the public disclosure of remediated vulnerabilities, in full or in redacted form, on a case-by-case basis. Ultimately, while disclosure may be requested by the finder of the vulnerability, this decision remains the sole discretion of the organization. Removing a vulnerability from consideration for coordinated disclosure is sometimes necessary when disclosing it would result in significant risk to customers. This is the case with pacemakers, vehicles, and other IoT devices that are difficult to recall quickly or update remotely.

COORDINATED DISCLOSURE

For more mature organizations, setting a “timer” for resolving and publishing every vulnerability can further encourage more active discovery, although this protocol often requires a dedicated team responsible for rapid remediation and communication. This approach is often taken by organizations that deem security to be a strategic priority and need to invest in building the best possible relationship with the security community.

Coordinated disclosure is based on good faith and is considered a best practice for all parties involved, as it encourages rapid remediation while demonstrating commitment to and appreciation of the hacker community.

FULL DISCLOSURE

Unlike the other approaches, full disclosure is not a program policy. Rather, it is an individual instance of public communication wherein a finder discloses a vulnerability before it has been fixed. Bruce Schneier defended the merits of full disclosure in 2007, suggesting that the threat of this act is sometimes necessary to force owners to fix vulnerabilities when they are unresponsive to hackers’ well-intended communications.

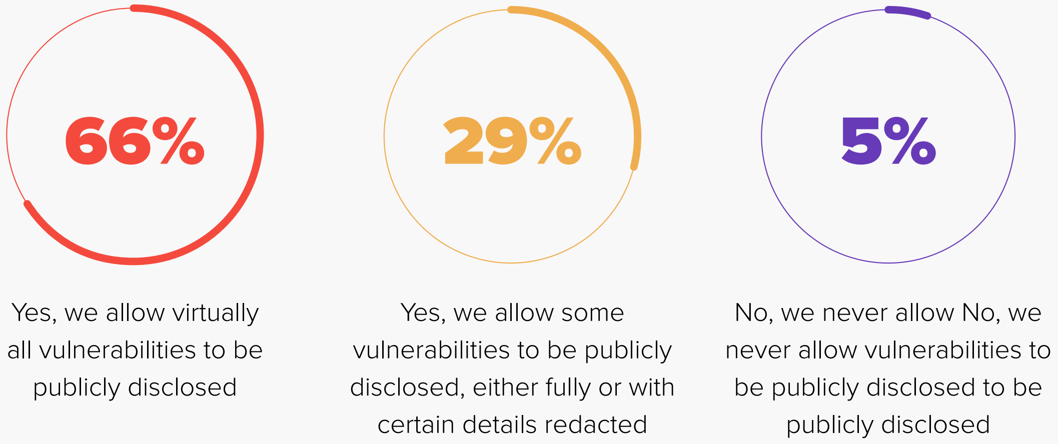
However, both hackers and organizations often prefer to avoid this type of disclosure at all costs.

In fact, both nondisclosure and full disclosure are discouraged because of the asymmetric cost to only one party; either the finder is not given recognition for their effort to improve security, or the owner is not given an opportunity to fix a vulnerability before it becomes public in a way that makes it more likely to be maliciously exploited. Disclosure should be undertaken in a way that protects the owner, rewards the finder, incentivizes further research, and enhances relationships between owners and the security community.

NON-DISCLOSURE

When programs are marked as “nondisclosure,” it is understood that the finder is not permitted to communicate any portion of a vulnerability beyond the confines of the organization itself, even after it has been resolved. For nondisclosure programs, no vulnerability, regardless of type or severity, can be shared. While these programs still receive submissions, they do not encourage them.

Does your organization allow coordinated disclosure?



OBSTACLES TO DISCLOSURE

In addition to improving the security posture of other organizations, coordinated and discretionary disclosure policies strengthen the relationship between an organization and the hacker community.

> **For hackers, their reputations are their brands, and receiving acknowledgment for identifying an exceptionally complex vulnerability enhances their reputation and increases their market value. Organizations that clearly state their willingness to collaborate on disclosing vulnerabilities in advance can expect better relationships with the security community, and often, greater program activity.**

While the rationale seems straightforward enough for both parties, disclosure decisions are not quite that simple for many organizations.

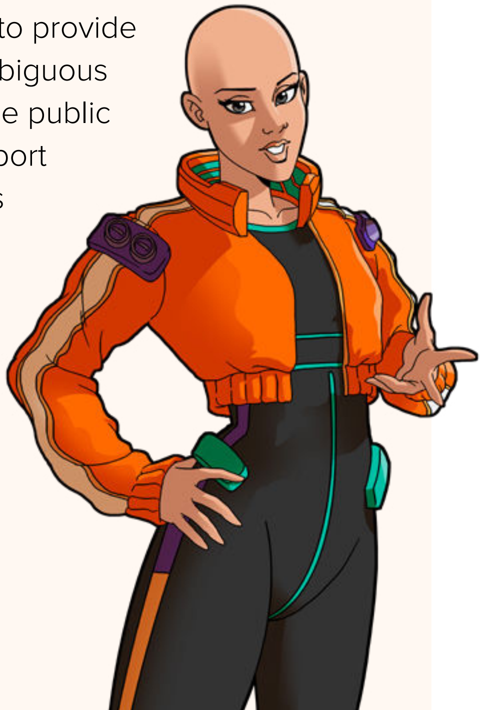
It is sometimes the case that perceived duties to stakeholders and the board can negatively impact an owner's disclosure decisions. Embracing vulnerability disclosure creates a security-first mentality, builds an organization's reputation within the security community, and educates a board in the process. That way, if there is ever a breach, the standard line "we take our security seriously" will carry far more weight.

Some security activists argue that the threat of full disclosure is necessary to keep owners honest and incentivize them to fix vulnerabilities. However, many owners argue that legal protections are necessary to prevent the threat of full disclosure becoming a vector for blackmail. A solid legal framework that recognizes the motivations of all parties is the best basis for facilitating vulnerability reporting and remediation.

Legal implications of VDPs

In 2020, the U.S. Federal Trade Commission (FTC), Department of Justice (DOJ), and Cybersecurity and Infrastructure Security Agency (CISA) released directives outlining the need for VDPs. With support from major legislative bodies like the **National Institute of Standards and Technology (NIST),** widespread adoption of VDPs is expected and necessary in the coming years.

In July 2023, the U.S. Securities and Exchange Commission (SEC) adopted new rules for **Cybersecurity Risk Management, Governance, and Incident Disclosure.** The ruling requires organizations to disclose material cyber incidents within four days of determining the criticality of the incident. To be in a position to responsibly comply, organizations must have in place the processes to meet the four-day requirement. One thing organizations can do to facilitate these processes is to provide a clear, unambiguous method for the public at large to report vulnerabilities under safe harbor, aka, a VDP.



The Binding Operational Directive 20-01 issued by the CISA requires all 100+ Federal Civilian Executive Branch agencies to develop and implement a VDP. This means that vulnerability disclosure policies are now a federal mandate.

There are just two examples of the increasingly popular belief that VDPs are a must for compliance and establish **a baseline for security best practices.**

Vulnerability disclosure legal status

Aligning the interests, incentives, and expectations of both hackers and host organizations primarily involves frequent and clear communication, but there is also a need to provide unambiguous legal clarity and assurance. Hacking involves testing, stressing, and sometimes even breaking software to rebuild and improve it. This creates problems, given a legal system that defaults to ownership as a starting point and presumes malice to be the motive for any party who uses and abuses software outside its supposed scope. As a result, the default legal status of vulnerability discovery and disclosure excludes good faith hacking.

The Computer Fraud and Abuse Act (CFAA) prohibits the access of a computer without authorization or the exceeding of authorized access. This renders good faith testing of assets illegal where robust VDPs are not in place, and while the number of hackers convicted for related offenses is low, it has nevertheless had a chilling effect on the community; 60% of hackers do not submit vulnerabilities due to fear of legal retribution.

The Digital Millennium Copyright Act (DMCA) makes it illegal to circumvent controls that prevent access to copyrighted material, defined to include software. This applies even to the legal owners of the products in question.

These laws were passed during a time when hacking was mostly done maliciously, before the advent of bug bounties, good faith hacking, and a thriving community of professional hackers. While the DMCA was amended in 2016 to allow hackers to work on owned consumer devices in good faith, there are still legal gaps that need to be resolved before organizations can fully benefit from VDPs.

Organizations must draft terms for VDPs to allow and incentivize **good faith testing** and the submission of vulnerabilities in a way that keeps lawyers happy by ruling out backdoor entry points or loopholes for malicious actors. These agreements create a legally robust “safe harbor” for well-intentioned hackers, which considerably increases the number and quality of vulnerabilities submitted.

One starting point to consider is **Disclose.io,** an open source standardization project that offers a **boilerplate VDP framework,** instilling a safe harbor and enabling good faith hacking. This provides an accessible legal agreement for the research and disclosure of vulnerabilities and standardizes terms and policies to create a more welcoming space for hackers, many of whom do not speak English as a first language and have minimal legal knowledge (keeping in mind that legal frameworks also differ between countries). The safe harbor terms from Disclose.io were adopted in 2020 by CISA DHS, voting machine manufacturers, and a number of U.S. states to encourage transparency and reporting of cybersecurity issues that could potentially impact elections.

VDP best practices

VDPs often serve as an organization's first foray into the world of crowdsourced cybersecurity. For many organizations, a VDP is the f**irst opportunity to work with the hacker community.** Both of these can be a little overwhelming to launch but have massive benefits.

How to launch a VDP

Having a VDP is quickly becoming industry standard and is, in fact, no longer optional for some. The CISA issued a binding directive requiring all federal agencies to publish a VDP.

There are five key steps that every organization can follow to build a strong VDP:

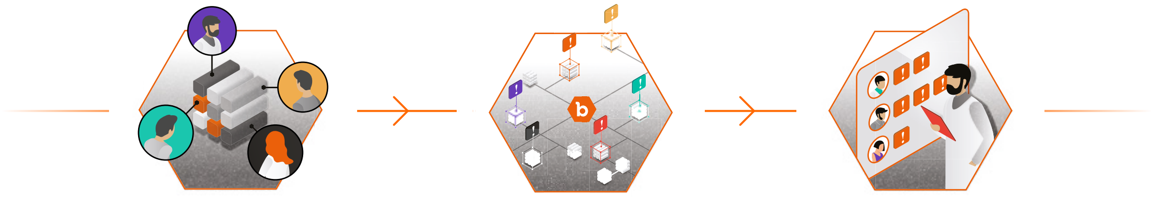
Decide Hosted Or On Self-Managed

Bugcrowd offers managed VDPs to help alleviate the burden of the time and effort required to construct and run an effective disclosure program. The Bugcrowd Platform is a data-driven SaaS framework that enables individuals to submit security feedback from anywhere in the world. The fully managed process includes the design and management of email and website-embedded submission forms; validation, categorization, and prioritization of vulnerabilities; integration with an organization's software development tools for faster remediation; and hacker communication, points-based remuneration, and support. Additionally, leveraging Bugcrowd for program management and enabling the option of having the program listed on Bugcrowd's hacker homepage brings the program to the attention of registered hackers for the increased likelihood of additional activity and submission volumes.

Companies with few internet-facing assets, limited resources, or still-maturing processes for accepting and remediating vulnerabilities may instead choose self-management, which usually equates to a more manageable flow of vulnerabilities. Of course, it's possible that incoming submissions may outpace the ability of a thinly resourced team to respond in time, which can lead to tension between hackers and an organization if communications are not prioritized. This tends to expedite the transition to a managed model, especially as evidence of urgency is usually quite easy to demonstrate to superiors.

Codify expectations

Organizations initiating a VDP should adhere to principles that make the program scalable and robust. This includes providing clear authorization of access to good faith hackers. This should include broad indications regarding acceptable conduct, as well as techniques that could be considered out of scope, such as DDoS or social engineering. Organizations should also determine the scope of assets covered by the policy, with restrictions for third-party data or personal information or a requirement that hackers use test accounts and dummy data when testing for vulnerabilities. Organizations with limited resources may also want to restrict the assets covered by the program to start with to ensure that they have the resources to deal with vulnerabilities submitted.



Anyone can submit security feedback

Any asset that is internet-connected

Submissions triaged and results shared

Expect to iterate

Starting a VDP can be overwhelming, so commit to a phased timeline that allows plenty of room for gathering data and making adjustments. The scope should be revised in line with this data.

No organization will land on their ideal scope, preferred disclosure policy, and the most efficient communication process with their first attempt, so the best approach is to build iteratively. Play around with parameters and approaches, and gather plenty of data to become informed. As long as organizations don't offend the security community or their boards, all steps taken are valuable.

Be accessible

It is also important to give clear guidance regarding communication within dedicated channels. This could be a security@companyname.com email address to begin with, but it is crucial to avoid single points of failure. Establishing multiple channels, safeguards, and responsible parties can prevent an unchecked inbox or an overactive spam filter from creating blind spots and associated risks.

Factor in respect

Finally, and perhaps most importantly, a VDP should define clear disclosure standards based on good faith. These create the baseline for the strength of an organization's relationship with hackers and should align incentives to ensure that both parties benefit from interactions. Owners should receive as much detail of a vulnerability as possible, along with a good faith commitment from the hacker to stick to the agreed method of disclosure. Hackers should expect to get prompt responses to their submissions and a commitment to their appropriate recognition.

Managing a VDP

Those willing to implement best practices in vulnerability disclosure can both set a standard among peers while differentiating themselves from their competitors. Here are some steps that can make VDPs work best for organizations, partners, and the security community.

Align expectations

Hackers should feel legally protected and know exactly how to report a bug and what to expect throughout the process. Don't be afraid to overcommunicate.

Provide clear legal guidance

Use standardized terms and clear examples to encourage good faith interactions and authorize conduct under the CFAA by providing explicit consent for system access.

Ground interactions in good faith

Allow for the accidental overreach of scope by hackers done in good faith. Ensure policies prioritize relationships and industry norms over strict interpretations of guidelines.

Remediate efficiently

Prioritize end users and the vulnerability finder by getting to work resolving a bug and validating a fix quickly.

Start a dialogue

VDPs are a two-way street, and there are long-term benefits to working on clear communication and appropriate incentives to cement good relationships with hackers.

Troubleshoot the process

Remove single points of failure in communication channels, seek feedback from hackers, and commit to flexibility in a VDP's philosophy and operation.

Take an integrated approach

VDPs are just one of a number of overlapping tools and procedures that make up an organization's security posture. Ensure all processes and products are configured to move in the same direction.

Know your limits

Unless managed with the expectation of growth, VDPs can become overwhelming. Work with security teams and VDP providers to configure a manageable solution.

Combining VDPs with bug bounty and pen testing

Bug bounty programs—which some call “VDPs with rewards”—allow organizations to direct targeted, rigorous testing at business-critical assets. Similarly, pen test programs enable organizations to focus on compliance-related assets or those in which a structured methodology would improve how security posture is communicated to partners, investors, and customers. Vulnerabilities found through these programs qualify for financial rewards, so most organizations limit their scope for budgetary reasons, and they may also impose limited testing windows. While economical, this creates gaps in coverage and wrongfully assumes that all potential vulnerabilities can and will be surfaced through an exclusive (often private) crowd of hackers.

79%

of organizations with a VDP run it alongside bug bounties and/or pen tests.

NIST 800-53 r5 codified the idea that a public bug bounty program is actually a subset of a VDP and is specifically a VDP where monetary rewards are optionally offered as thanks to the finder.

Each program has its strengths and limitations. Pen testing has been recognized and accepted by the auditing community, which makes it useful for assets where compliance is of particular importance.

However, the limits in scope and partners involved mean pen tests can become rigid and less effective over time. VDP programs add a much-needed, yet economical, tool for catching vulnerabilities surfaced by anyone, anywhere. But when is the right time to implement a VDP?

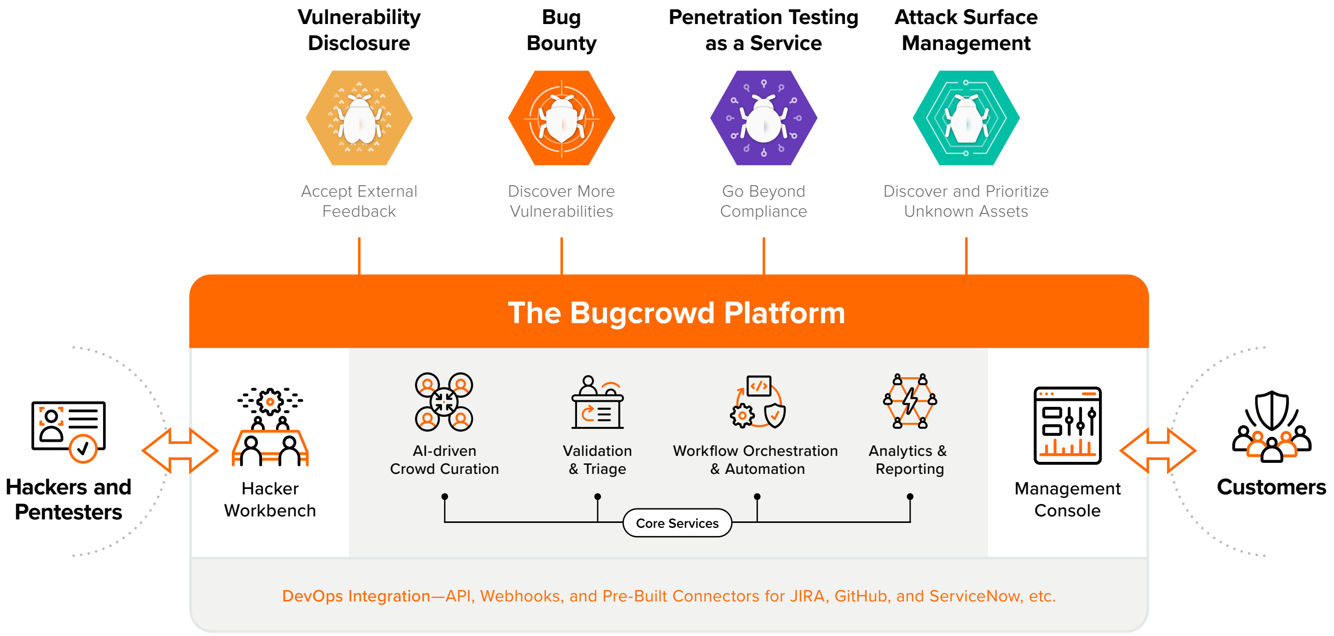
99%

of organizations already run or would consider running a pay-per-finding bug bounty alongside their VDPs for targeted testing on priority assets.

The market has tied itself in knots trying to create a linear maturity model for when and how to “progress” between a VDP, Bug Bounty, and/or Pen Test. However, each should be viewed as providing complementary benefits, with adoption driven by individual goals and resources rather than maturity. Think of a VDP as the first building block in external testing. While an agreed-upon sequence might make for tidier budgeting, it also goes against the organic, adaptive, and sometimes unruly nature of security. Every organization is different.

The Bugcrowd Security Knowledge Platform

VDPs aren't the only way to leverage the power of the Crowd. The multi-solution Bugcrowd Platform brings the right crowd into all your workflows at the right time, allowing you to run bug bounties, pen tests, VDPs, and more at scale and in an integrated, coordinated way.


Vulnerability Disclosure
Accept External Feedback
Bug Bounty
Discover More Vulnerabilities
Penetration Testing as a Service
Go Beyond Compliance
Attack Surface Management
Discover and Prioritize Unknown Assets
The Bugcrowd Platform
Hackers and Pentesters
Hacker Workbench
AI-driven Crowd Curation
Validation & Triage
Workflow Orchestration & Automation
Analytics & Reporting
Core Services
Management Console
Customers
DevOps Integration—API, Webhooks, and Pre-Built Connectors for JIRA, GitHub, and ServiceNow, etc.

✓ **Best Security ROI from The Crowd**

We match you with the right trusted hackers for your needs and environment across hundreds of dimensions using AI.

✓ **Instant focus on critical issues**

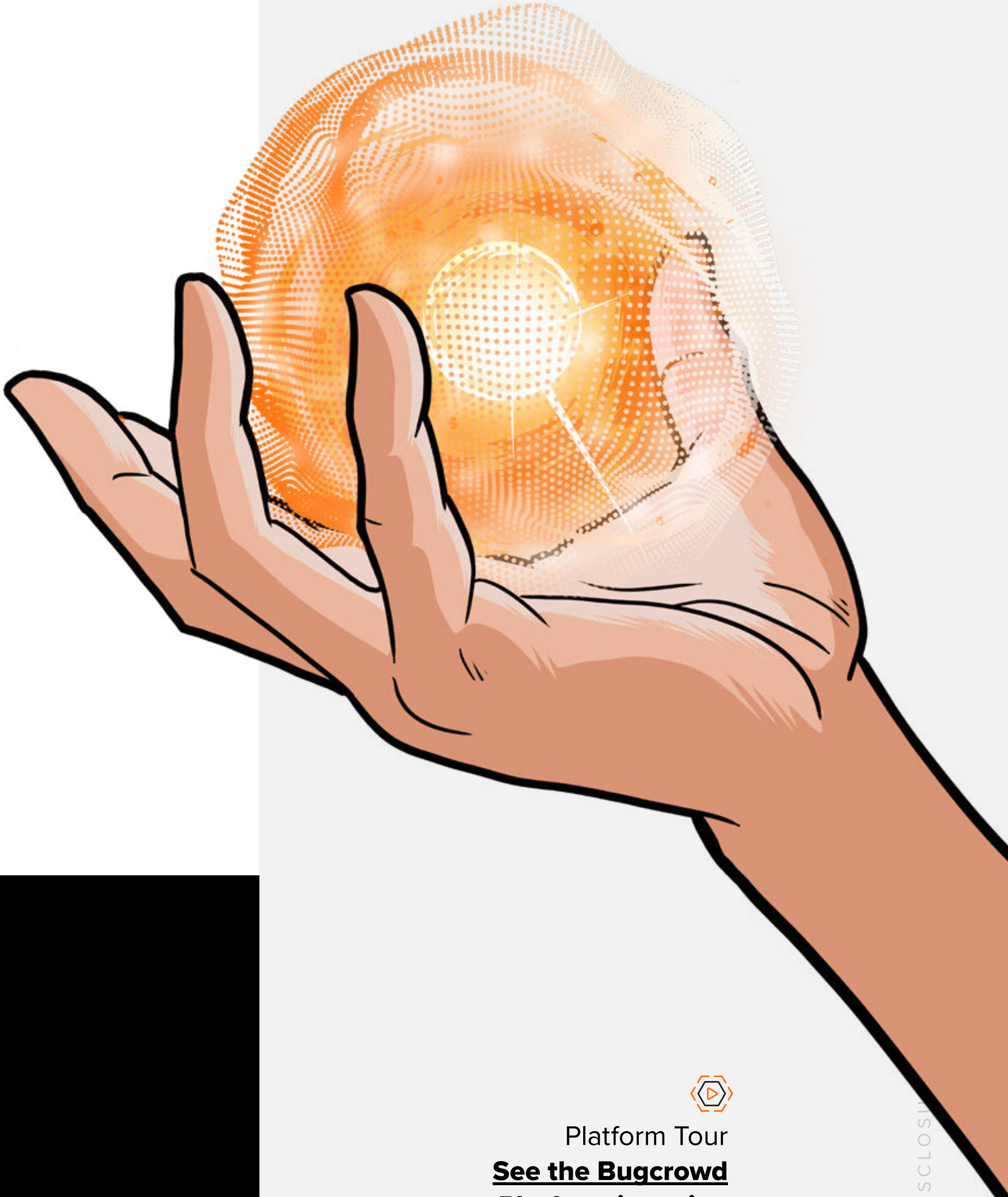
Working as an extension of the platform, our global security engineer team rapidly validates and triages submissions, with P1s (critical vulnerabilities) often handled within hours.

✓ **Continuous, resilient security for DevOps**

The platform integrates workflows with existing tools and processes to ensure that applications and APIs are continuously tested before they ship.

✓ **Contextual intelligence for best results**

We apply accumulated knowledge from over a decade of experience crafting thousands of customer solutions to your goals for better outcomes.



Unleash Human Creativity for Proactive Security

Try Bugcrowd

Platform Tour

See the Bugcrowd Platform in action

Data Sheet

Vulnerability Disclosure Program

bugcrowd