bugcrowd

DEFENSIVE VULNERABILITY PRICING MODEL

How to Budget for Your Crowdsourced Security Program

Table of Contents

Background

The following guidance is taken from nearly 1,000 programs and 7 years of experience running successful programs for organizations in more than 50 industries in over 30 countries. Bugcrowd's customers include some of the largest and most mature security organizations in the world.

This guidance is specifically focused on defensive vulnerability pricing for web and mobile applications, APIs, thick clients, and embedded devices.

Bugcrowd's crowdsourced security programs are both public and private and may be ongoing, or time boxed. Regardless, the defensive vulnerability pricing model applies.

What's a Bug Really Worth?

"What is a bug worth?" "What should my organization budget for a successful program?" We receive these questions time and again from prospects and customers looking at crowdsourced security programs.

These are great questions, and frankly, the answers to them will continue to evolve as the market matures. Yet, the key to success remains the same: to run a successful bounty program you must attract the right researchers with the appropriate incentive.

With over 7 years of experience helping hundreds of companies run both public and private crowdsourced security programs, Bugcrowd has collected vulnerability data from many different types of programs and organizations. From that data, we are pleased to provide you with a very practical assessment of the current market rate for vulnerabilities.

Our hope is that these insights will set you up for success in running a crowdsourced security program, and help you work with the researcher community more effectively.

Security Maturity

Your baseline budget is based on the security maturity of your organization.

Organizational security maturity is a foundational element for determining how to reward a vulnerability. An organization with a more mature security program has security-focused processes in place, and thus, vulnerabilities require more time and effort to find.

While there are many different factors that dictate the maturity of your security practice and in truth, it's a broad spectrum, the below table produced by the **Enterprise Strategy Group** is a simple way to identify the current maturity state of an organization at a high level.

	Basic	Progressing	Advanced
Philosophy	"Cybersecurity is a necessary evil."	"Cybersecurity must be more integrated into the business."	"Cybersecurity is part of the culture."
People	CISO reports to IT; Small security team with minimal skills	CISO reports to COO or other non-IT manager; Larger security team with some autonomy from IT	CISO reports to CEO and is active with the board; Large, well-organized staff with good work environment
Process	Informal and ad-hoc; Subservient to IT	Better coordination with IT, but processes are rare, informal, manual and dependent upon individual contributors	Documented and formal with an eye toward scale and automation
Technology	Elementary security technologies with simple configurations; Decentralized security organizations with limited coordination across functions; Focus on preventions and regulatory compliance	More advanced use of security technologies and adoption of new tools for incident detections and security analytics	Building enterprise security technology architecture; Focus on incident prevention, detection and response; Adding elements of identity management and data security to deal with cloud and mobile computing security

Source: Enterprise Strategy Group

Its important to note that there might be a range of targets with differing levels of maturity in the organization. This maturity model can be applied on an organizational, or a target basis.

Priority of a Bug

Vulnerability types that deserve the highest priority rating

Once a baseline maturity has been established, a context for the organization's pricing structure can be established. To select the specific payout, however, we need a priority for the vulnerability. The priority of a bug is important to the reward process because higher priority issues deserve higher rewards — they require more time, effort and skill to identify. To obtain priority, we must evaluate the vulnerability for technical and business impact.

When Bugcrowd receives a reproducible vulnerability, it's evaluated by Application Security Engineers for technical impact and a baseline priority is assigned. This submission is then sent to the customer for evaluation of business impact and the final priority may be adjusted in context of the application, external security controls, or the organization's processes.

We have found that using a 1 through 5 priority scale works well for this process, with a "P1" label assigned to the most critical vulnerabilities and a "P5" label assigned to vulnerabilities that are acceptable risks. Below, we have provided a baseline priority matrix, which serves as a guide for Bugcrowd's application security engineers. This priority matrix is often adjusted by customers, but is a useful starting point.

Priority	Impact	Vulnerability Types
P1 - Critical	Vulnerabilities that cause a privilege escalation from unprivileged to admin or allow for remote execution, financial theft, etc.	• Remote Code Execution • Vertical Authentication Bypass • XML External Entities Injection with significant impact • SQL Injection with significant impact
P2 - High	Vulnerabilities that affect the security of the platform including the processes it supports	• Lateral authentication bypass • Stored XXS with significant impact • CSRF with significant impact • Direct object reference with significant impact • Internal SSRF
P3 - Medium	Vulnerabilities that affect multiple users and require little or no user interaction to trigger	• Reflective XXS with impact • Direct object reference • URL redirect • CSRF with impact
P4 - Low	Vulnerabilities that affect singular users and require interaction or significant prerequisites to trigger (MitM) to trigger	• SSL misconfigurations with little impact • SPF configuration problems • XSS with limited impact • CSRF with limited impact
P5 - Acceptable Risk	Non-exploitable vulnerabilities in functionality. Vulnerabilities that are by design or are deemed acceptable business risk to the customer	• Debug information • Use of CAPTCHAs • Code obfuscation • Rate limiting, etc.

Impact and vulnerability types by priority

The Market Rate for Bugs

Annual budget, reward ranges, and price per bug

Since 2012, Bugcrowd has gathered vulnerability data from nearly 1,000 programs. Using this data we have created a standard defensive vulnerability pricing model that delivers results for organizations.

Baseline Amounts

Security maturity and submission priority are the most prominent variables when determining the value of a bug, and give enough information to make a baseline recommendation for organizations.

It is not uncommon for organizations to start out with lower reward ranges and increase them over time. Lower reward ranges can bring initial success, however the reward range is what allows organizations to compete for talent within the market – and for sustained success, we suggest starting with the ranges below.

	Security Maturity Model			
	P4	P3	P2	P1
Untested Web Apps	$125	$400-$600	$1,000-$1,200	$2,100-$2,500
Moderately Tested Web Apps Untested APIs / Mobile Apps	$150	$600-$750	$1,500-$1,800	$3,100-$3,500
Well Tested Web Apps Moderately Tested: APIs / Mobile Apps Thick Clients / Binaries / Embedded Devices	$200	$750-$1,000	$1,800-$2,100	$4,500-$5,000
Well-hardened Web Apps Well-tested APIs / Mobile Apps Moderately Secure Thick Clients / Binaries / Embedded Devices	$300	$1,000-$1,250	$2,100-$2,500	$7,500-$10,000
Well-hardened Web Apps, APIs, and Mobile Apps Moderately- to Highly-Secured Thick Clients / Binaries and/or Hardened Embedded Devices	$500	$1,500-$2,000	$4,500-$5,000	$10,000-20,000+

Baseline Vulnerability Budget

Additional Considerations

Target Criticality
If the organization handles mission critical, highly sensitive or valuable information (e.g. PII, PHI, financial data, etc.), it is prudent to consider increasing payouts in order to attract and retain talent quickly.

Target Accessibility
If your targets require significant setups to test, or other situational knowledge, it is useful to modify the pricing to account for the ramp time.

World Class Security Maturity
If an organization has an extremely advanced security maturity, and/or the desire to attract the best security talent, you should multiply all base payouts by 2. This will keep you on par with some of the most prominent brands currently running bounty programs in the market.

Marketing Your Program
If an organization has a desire to utilize the program for marketing their security capabilities, they may choose to increase payouts in order to provide the broader market with a clear indicator of security maturity.

Here to Help

Not sure what pricing model your organization should follow? We are happy to help! Feel free to reach out to us and we can give you a free, no obligation consultation as to what range your company should set for a bounty program and what to expect to spend over the next year. **Contact us** for a free consultation..