

BUGCROWD'S

VULNERABILITY RATING TAXONOMY

Bugcrowd is proud of the VRT, a valuable resource for both researchers and customers to better understand the technical rating we use to classify vulnerabilities. This report details how and why we created the VRT, and a usage guide to accompany the taxonomy itself.



THE METHODOLOGY

At the beginning of 2016, we released the Bugcrowd Vulnerability Rating Taxonomy (VRT) in an effort to further bolster transparency and communication, as well as to contribute valuable and actionable content to the bug bounty community.

Bugcrowd's VRT is a resource outlining Bugcrowd's baseline severity rating, including certain edge cases, for vulnerabilities that we see often. To arrive at this baseline rating, Bugcrowd's security engineers started with generally accepted industry impact and further considered the average acceptance rate, average priority, and commonly requested program-specific exclusions (based on business use cases) across all of Bugcrowd's programs.

Implications For Bug Hunters

Bugcrowd's VRT is an invaluable resource for bug hunters as it outlines the types of issues that are normally seen and accepted by bug bounty programs. We hope that being transparent about the typical severity level for various bug types will help bug bounty participants save valuable time and effort in their quest to make bounty targets more secure. The VRT can also help researchers identify which types of high-value bugs they have overlooked, and when to provide exploitation information (POC info) in a report where it might impact priority.

Interested in becoming a Bugcrowd researcher? [Join the crowd.](#)

Implications For Customers

The VRT helps customers gain a more comprehensive understanding of bug bounties. The following information in this document will help our customers understand the impact of a given vulnerability, assist any adjustments to a bounty scope, and provides insight to write a clear bounty brief. During remediation, the VRT will help business units across the board in communicating the severity of identified security issues. For more information on our severity rating and worth of a bug, [read our recently launched guide "What's A Bug Worth."](#)

USAGE GUIDE:

The VRT is intended to provide valuable information for bug bounty stakeholders. It is important that we identify the ways in which we use it successfully, and what considerations should be kept in mind.

The Severity Rating is a Baseline

The recommended severity, from P1 to P5, is a baseline. That having been said, while this severity rating might apply without context, it's possible that application complexity, bounty brief restrictions, or unusual impact could result in a different rating. As a customer, it's important to weigh the VRT alongside your internal application security ratings.

For bug hunters, if you think a bug's impact warrants reporting despite the VRT's guidelines, or that the customer has misunderstood the threat scenario, we encourage you to submit the issue regardless and use the [Bugcrowd Crowdcontrol](#) commenting system to clearly communicate your reasoning.

Low Severity Does Not Imply Insignificance

For customers, it's important to recognize that the base severity rating does not equate to "industry accepted impact." This rating is defined by our Security Operations Team and our VRT is a living document - see the following point about the "VRT Council." Your internal teams or engineers might assess certain bugs - especially those designated P4 or P5 within the VRT - differently. As a bug hunter, it's important to not discount lower severity bugs, as many bug hunters have used such bugs within "exploit chains" consisting of two or three bugs resulting in creative, valid, and high-impact submissions.

Importance of a VRT Council

Bugcrowd reviews proposed changes to the VRT every week at an operations meeting with "VRT Council." We use this one-hour meeting to discuss new vulnerabilities, edge cases for existing vulnerabilities, technical severity level adjustments, and to share general bug validation knowledge. When the team comes to a consensus regarding each proposed change, it is committed to the master version. Members of the Security Operations team look forward to this

meeting each week, as examining some of the most difficult to validate bugs serves as a unique learning exercise.

[This specific document will be updated on an ongoing basis.](#)

Communication is King

Having cut-and-dry baseline ratings, as defined by our VRT, make rating bugs a faster and less difficult process. We have to remember, however, that strong communication is the most powerful tool for anyone running or participating in a bug bounty.

Both sides of the bug bounty equation must exist in balance. When in doubt, ask dumb questions, be verbose, and more generally, behave in a way that allows you and your bounty opposite to foster a respectful relationship. As a customer, keep in mind that every bug takes time and effort to find. As a bounty hunter, try to remember that every bug's impact is ultimately determined by the customer's environment and use cases.

One Size Doesn't Fit All

While this taxonomy maps bugs to the OWASP Top Ten and the OWASP Mobile Top Ten to add more contextual information, additional meta-data could include CWE or WASC, among others. As always, the program owner retains all rights to choose final bug prioritization levels.

Priority	OWASP Top Ten + Bugcrowd Extras	Specific Vulnerability Name	Variant or Affected Function
P1	Server Security Misconfiguration	Using Default Credentials	
	Server-Side Injection	File Inclusion	Local
	Server-Side Injection	Remote Code Execution (RCE)	
	Server-Side Injection	SQL Injection	
	Server-Side Injection	XML External Entity Injection (XXE)	
	Broken Authentication and Session Management	Authentication Bypass	
	Sensitive Data Exposure	Critically Sensitive Data	Password Disclosure
	Sensitive Data Exposure	Critically Sensitive Data	Private API Keys
	Insecure OS/Firmware	Command Injection	
	Insecure OS/Firmware	Hardcoded Password	Privileged User
	Broken Cryptography	Cryptographic Flaw	Incorrect Usage
P2	Server Security Misconfiguration	Misconfigured DNS	High Impact Subdomain Takeover
	Server Security Misconfiguration	OAuth Misconfiguration	Account Takeover
	Cross-Site Scripting (XSS)	Stored	Non-Privileged User to Anyone
	Broken Access Control (BAC)	Server-Side Request Forgery (SSRF)	Internal
	Cross-Site Request Forgery (CSRF)	Application-Wide	
	Application-Level Denial-of-Service (DoS)	Critical Impact and/or Easy Difficulty	
	Insecure OS/Firmware	Hardcoded Password	Non-Privileged User
P3	Server Security Misconfiguration	Misconfigured DNS	Basic Subdomain Takeover
	Server Security Misconfiguration	Mail Server Misconfiguration	Email Spoofing on Email Domain
	Server-Side Injection	HTTP Response Manipulation	Response Splitting (CRLF)
	Server-Side Injection	Content Spoofing	iframe Injection
	Broken Authentication and Session Management	Second Factor Authentication (2FA) Bypass	
	Broken Authentication and Session Management	Weak Login Function	HTTPS not Available or HTTP by Default
	Broken Authentication and Session Management	Session Fixation	Remote Attack Vector
	Sensitive Data Exposure	EXIF Geolocation Data Not Stripped From Uploaded Images	Automatic User Enumeration
	Cross-Site Scripting (XSS)	Stored	Privileged User to Privilege Elevation

Priority

OWASP Top Ten + Bugcrowd Extras

Specific Vulnerability Name

Variant or Affected Function

P3
CONTINUED

Cross-Site Scripting (XSS)

Stored

CSRF/URL-Based

Cross-Site Scripting (XSS)

Reflected

Non-Self

Application-Level Denial-of-Service (DoS)

High Impact and/or Medium Difficulty

Client-Side Injection

Binary Planting

Default Folder Privilege Escalation

P4

Server Security Misconfiguration

Misconfigured DNS

Zone Transfer

Server Security Misconfiguration

Database Management System (DBMS) Misconfiguration

Excessively Privileged User / DBA

Server Security Misconfiguration

Lack of Password Confirmation

Delete Account

Server Security Misconfiguration

No Rate Limiting on Form

Registration

Server Security Misconfiguration

No Rate Limiting on Form

Login

Server Security Misconfiguration

No Rate Limiting on Form

Email-Triggering

Server Security Misconfiguration

No Rate Limiting on Form

SMS-Triggering

Server Security Misconfiguration

Missing Secure or HTTPOnly Cookie Flag

Session Token

Server Security Misconfiguration

Clickjacking

Sensitive Click-Based Action

Server Security Misconfiguration

CAPTCHA

Implementation Vulnerability

Server Security Misconfiguration

Lack of Security Headers

Cache-Control for a Sensitive Page

Server Security Misconfiguration

Web Application Firewall (WAF) Bypass

Direct Server Access

Server-Side Injection

Content Spoofing

External Authentication Injection

Server-Side Injection

Content Spoofing

Email HTML Injection

Broken Authentication and Session Management

Cleartext Transmission of Session Token

Broken Authentication and Session Management

Weak Login Function

Other Plaintext Protocol with no Secure Alternative

Broken Authentication and Session Management

Weak Login Function

LAN Only

Broken Authentication and Session Management

Weak Login Function

HTTP and HTTPS Available

Broken Authentication and Session Management

Failure to Invalidate Session

On Logout (Client and Server-Side)

Broken Authentication and Session Management

Failure to Invalidate Session

On Password Reset and/or Change

Broken Authentication and Session Management

Weak Registration Implementation

Over HTTP

Sensitive Data Exposure

EXIF Geolocation Data Not Stripped From Uploaded Images

Manual User Enumeration

Sensitive Data Exposure

Visible Detailed Error/Debug Page

Detailed Server Configuration

Sensitive Data Exposure

Token Leakage via Referer

Untrusted 3rd Party

Sensitive Data Exposure

Token Leakage via Referer

Over HTTP

Priority

P4
CONTINUED

OWASP Top Ten + Bugcrowd Extras

Specific Vulnerability Name

Variant or Affected Function

Sensitive Data Exposure

Sensitive Token in URL

User Facing

Sensitive Data Exposure

Weak Password Reset Implementation

Password Reset Token Sent Over HTTP

Cross-Site Scripting (XSS)

Stored

Privileged User to No Privilege Elevation

Cross-Site Scripting (XSS)

Flash-Based

Cross-Site Scripting (XSS)

IE-Only

IE11

Cross-Site Scripting (XSS)

Referer

Cross-Site Scripting (XSS)

Universal (UXSS)

Cross-Site Scripting (XSS)

Off-Domain

Data URI

Broken Access Control (BAC)

Server-Side Request Forgery (SSRF)

External

Broken Access Control (BAC)

Username Enumeration

Data Leak

Unvalidated Redirects and Forwards

Open Redirect

GET-Based

Insufficient Security Configurability

No Password Policy

Insufficient Security Configurability

Weak Password Reset Implementation

Token is Not Invalidated After Use

Using Components with Known Vulnerabilities

Rosetta Flash

Insecure Data Storage

Sensitive Application Data Stored Unencrypted

On External Storage

Insecure Data Storage

Server-Side Credentials Storage

Plaintext

Insecure Data Transport

Executable Download

No Secure Integrity Check

Privacy Concerns

Unnecessary Data Collection

WiFi SSID+Password

Mobile Security Misconfiguration

Clipboard Enabled

On Sensitive Content

P5

Server Security Misconfiguration

Directory Listing Enabled

Non-Sensitive Data Exposure

Server Security Misconfiguration

Same-Site Scripting

Server Security Misconfiguration

Misconfigured DNS

Missing Certification Authority Authorization (CAA) Record

Server Security Misconfiguration

Mail Server Misconfiguration

Missing SPF on Non-Email Domain

Server Security Misconfiguration

Mail Server Misconfiguration

SPF Uses a Soft Fail

Server Security Misconfiguration

Mail Server Misconfiguration

SPF Includes More Than 10 Lookups

Server Security Misconfiguration

Mail Server Misconfiguration

Missing DKIM/DMARC

Server Security Misconfiguration

Lack of Password Confirmation

Change Email Address

Server Security Misconfiguration

Lack of Password Confirmation

Change Password

Server Security Misconfiguration

Lack of Password Confirmation

Manage 2FA

Priority

P5
CONTINUED

OWASP Top Ten + Bugcrowd Extras	Specific Vulnerability Name	Variant or Affected Function
Server Security Misconfiguration	Unsafe File Upload	No Antivirus
Server Security Misconfiguration	Unsafe File Upload	No Size Limit
Server Security Misconfiguration	Unsafe File Upload	File Extension Filter Bypass
Server Security Misconfiguration	Cookie Scoped to Parent Domain	
Server Security Misconfiguration	Missing Secure or HTTPOnly Cookie Flag	Non-Session Cookie
Server Security Misconfiguration	Clickjacking	Form Input
Server Security Misconfiguration	Clickjacking	Non-Sensitive Action
Server Security Misconfiguration	CAPTCHA	Brute Force
Server Security Misconfiguration	CAPTCHA	Missing
Server Security Misconfiguration	Exposed Admin Portal	To Internet
Server Security Misconfiguration	Missing DNSSEC	
Server Security Misconfiguration	Fingerprinting/Banner Disclosure	
Server Security Misconfiguration	Username Enumeration	Brute Force
Server Security Misconfiguration	Potentially Unsafe HTTP Method Enabled	OPTIONS
Server Security Misconfiguration	Potentially Unsafe HTTP Method Enabled	TRACE
Server Security Misconfiguration	Insecure SSL	Lack of Forward Secrecy
Server Security Misconfiguration	Insecure SSL	Insecure Cipher Suite
Server Security Misconfiguration	Insecure SSL	Certificate Error
Server Security Misconfiguration	Reflected File Download (RFD)	
Server Security Misconfiguration	Lack of Security Headers	X-Frame-Options
Server Security Misconfiguration	Lack of Security Headers	Cache-Control for a Non-Sensitive Page
Server Security Misconfiguration	Lack of Security Headers	X-XSS-Protection
Server Security Misconfiguration	Lack of Security Headers	Strict-Transport-Security
Server Security Misconfiguration	Lack of Security Headers	X-Content-Type-Options
Server Security Misconfiguration	Lack of Security Headers	Content-Security-Policy
Server Security Misconfiguration	Lack of Security Headers	Public-Key-Pins
Server Security Misconfiguration	Lack of Security Headers	X-Content-Security-Policy
Server Security Misconfiguration	Lack of Security Headers	X-Webkit-CSP
Server Security Misconfiguration	Lack of Security Headers	Content-Security-Policy-Report-Only
Server Security Misconfiguration	Bitsquatting	

Server-Side Injection	Parameter Pollution	Social Media Sharing Buttons
Server-Side Injection	Content Spoofing	Flash Based External Authentication Injection
Server-Side Injection	Content Spoofing	Text Injection
Server-Side Injection	Content Spoofing	Homograph/IDN-Based
Server-Side Injection	Content Spoofing	Right-to-Left Override (RTLO)
Broken Authentication and Session Management	Weak Login Function	Not Operational or Intended Public Access
Broken Authentication and Session Management	Session Fixation	Local Attack Vector
Broken Authentication and Session Management	Failure to Invalidate Session	On Logout (Server-Side Only)
Broken Authentication and Session Management	Failure to Invalidate Session	Concurrent Sessions On Logout
Broken Authentication and Session Management	Failure to Invalidate Session	On Email Change
Broken Authentication and Session Management	Failure to Invalidate Session	Long Timeout
Broken Authentication and Session Management	Concurrent Logins	
Sensitive Data Exposure	Visible Detailed Error/Debug Page	Full Path Disclosure
Sensitive Data Exposure	Visible Detailed Error/Debug Page	Descriptive Stack Trace
Sensitive Data Exposure	Disclosure of Known Public Information	
Sensitive Data Exposure	Token Leakage via Referer	Trusted 3rd Party
Sensitive Data Exposure	Sensitive Token in URL	In the Background
Sensitive Data Exposure	Sensitive Token in URL	On Password Reset
Sensitive Data Exposure	Non-Sensitive Token in URL	
Sensitive Data Exposure	Mixed Content (HTTPS Sourcing HTTP)	
Sensitive Data Exposure	Sensitive Data Hardcoded	OAuth Secret
Sensitive Data Exposure	Sensitive Data Hardcoded	File Paths
Sensitive Data Exposure	Internal IP Disclosure	
Sensitive Data Exposure	JSON Hijacking	
Cross-Site Scripting (XSS)	Stored	Self
Cross-Site Scripting (XSS)	Reflected	Self
Cross-Site Scripting (XSS)	Cookie-Based	
Cross-Site Scripting (XSS)	IE-Only	XSS Filter Disabled
Cross-Site Scripting (XSS)	IE-Only	Older Version (< IE11)
Cross-Site Scripting (XSS)	TRACE Method	
Broken Access Control (BAC)	Server-Side Request Forgery (SSRF)	DNS Query Only
Cross-Site Request Forgery (CSRF)	Action-Specific	Logout

Cross-Site Request Forgery (CSRF)	CSRF Token Not Unique Per Request	
Application-Level Denial-of-Service (DoS)	App Crash	Malformed Android Intents
Application-Level Denial-of-Service (DoS)	App Crash	Malformed iOS URL Schemes
Unvalidated Redirects and Forwards	Open Redirect	POST-Based
Unvalidated Redirects and Forwards	Open Redirect	Header-Based
Unvalidated Redirects and Forwards	Open Redirect	Flash-Based
Unvalidated Redirects and Forwards	Tabnabbing	
Unvalidated Redirects and Forwards	Lack of Security Speed Bump Page	
External Behavior	Browser Feature	Plaintext Password Field
External Behavior	Browser Feature	Save Password
External Behavior	Browser Feature	Autocomplete Enabled
External Behavior	Browser Feature	Autocorrect Enabled
External Behavior	Browser Feature	Aggressive Offline Caching
External Behavior	CSV Injection	
External Behavior	Captcha Bypass	Crowdsourcing
External Behavior	System Clipboard Leak	Shared Links
External Behavior	User Password Persisted in Memory	
Insufficient Security Configurability	Weak Password Policy	
Insufficient Security Configurability	Weak Password Reset Implementation	Token is Not Invalidated After Email Change
Insufficient Security Configurability	Weak Password Reset Implementation	Token is Not Invalidated After Password Change
Insufficient Security Configurability	Weak Password Reset Implementation	Token Has Long Timed Expiry
Insufficient Security Configurability	Weak Password Reset Implementation	Token is Not Invalidated After New Token is Requested
Insufficient Security Configurability	Lack of Verification Email	
Insufficient Security Configurability	Lack of Notification Email	
Insufficient Security Configurability	Weak Registration Implementation	Allows Disposable Email Addresses
Insufficient Security Configurability	Weak 2FA Implementation	Missing Failsafe
Using Components with Known Vulnerabilities	Outdated Software Version	
Using Components with Known Vulnerabilities	Captcha Bypass	OCR (Optical Character Recognition)
Insecure Data Storage	Sensitive Application Data Stored Unencrypted	On Internal Storage
Insecure Data Storage	Non-Sensitive Application Data Stored Unencrypted	
Insecure Data Storage	Screen Caching Enabled	
Lack of Binary Hardening	Lack of Exploit Mitigations	
Lack of Binary Hardening	Lack of Jailbreak Detection	
Lack of Binary Hardening	Lack of Obfuscation	
Lack of Binary Hardening	Runtime Instrumentation-Based	

Priority	OWASP Top Ten + Bugcrowd Extras	Specific Vulnerability Name	Variant or Affected Function
	Network Security Misconfiguration	Telnet Enabled	
	Mobile Security Misconfiguration	SSL Certificate Pinning	Absent
	Mobile Security Misconfiguration	SSL Certificate Pinning	Defeatable
	Mobile Security Misconfiguration	Tapjacking	
	Mobile Security Misconfiguration	Clipboard Enabled	On Non-Sensitive Content
	Client-Side Injection	Binary Planting	Non-Default Folder Privilege Escalation
	Client-Side Injection	Binary Planting	No Privilege Escalation
VARIES			
	Server Security Misconfiguration	Unsafe Cross-Origin Resource Sharing	
	Server Security Misconfiguration	Path Traversal	
	Server Security Misconfiguration	Directory Listing Enabled	
	Server Security Misconfiguration	Directory Listing Enabled	Sensitive Data Exposure
	Server Security Misconfiguration	SSL Attack (BREACH, POODLE etc.)	
	Server Security Misconfiguration	Misconfigured DNS	
	Server Security Misconfiguration	Mail Server Misconfiguration	
	Server Security Misconfiguration	Database Management System (DBMS) Misconfiguration	
	Server Security Misconfiguration	Lack of Password Confirmation	
	Server Security Misconfiguration	No Rate Limiting on Form	
	Server Security Misconfiguration	Unsafe File Upload	
	Server Security Misconfiguration	Missing Secure or HTTPOnly Cookie Flag	
	Server Security Misconfiguration	Clickjacking	
	Server Security Misconfiguration	OAuth Misconfiguration	
	Server Security Misconfiguration	OAuth Misconfiguration	Missing/Broken State Parameter
	Server Security Misconfiguration	OAuth Misconfiguration	Insecure Redirect URI
	Server Security Misconfiguration	CAPTCHA	
	Server Security Misconfiguration	Exposed Admin Portal	
	Server Security Misconfiguration	Username Enumeration	
	Server Security Misconfiguration	Potentially Unsafe HTTP Method Enabled	
	Server Security Misconfiguration	Insecure SSL	
	Server Security Misconfiguration	Lack of Security Headers	
Server Security Misconfiguration	Web Application Firewall (WAF) Bypass		
Server-Side Injection			
Server-Side Injection	File Inclusion		
Server-Side Injection	Parameter Pollution		
Server-Side Injection	Content Spoofing		

Priority

VARIES
CONTINUED

OWASP Top Ten + Bugcrowd Extras	Specific Vulnerability Name	Variant or Affected Function
Broken Authentication and Session Management		
Broken Authentication and Session Management	Privilege Escalation	
Broken Authentication and Session Management	Weak Login Function	
Broken Authentication and Session Management	Session Fixation	
Broken Authentication and Session Management	Failure to Invalidate Session	
Broken Authentication and Session Management	Weak Registration Implementation	
Sensitive Data Exposure		
Sensitive Data Exposure	Critically Sensitive Data	
Sensitive Data Exposure	EXIF Geolocation Data Not Stripped From Uploaded Images	
Sensitive Data Exposure	Visible Detailed Error/Debug Page	
Sensitive Data Exposure	Token Leakage via Referer	
Sensitive Data Exposure	Sensitive Token in URL	
Sensitive Data Exposure	Weak Password Reset Implementation	
Sensitive Data Exposure	Sensitive Data Hardcoded	
Sensitive Data Exposure	Cross Site Script Inclusion (XSSI)	
Cross-Site Scripting (XSS)		
Cross-Site Scripting (XSS)	Stored	
Cross-Site Scripting (XSS)	Reflected	
Cross-Site Scripting (XSS)	IE-Only	
Cross-Site Scripting (XSS)	Off-Domain	
Broken Access Control (BAC)		
Broken Access Control (BAC)	Insecure Direct Object References (IDOR)	
Broken Access Control (BAC)	Server-Side Request Forgery (SSRF)	
Broken Access Control (BAC)	Username Enumeration	
Broken Access Control (BAC)	Exposed Sensitive Android Intent	
Broken Access Control (BAC)	Exposed Sensitive iOS URL Scheme	
Cross-Site Request Forgery (CSRF)		
Cross-Site Request Forgery (CSRF)	Action-Specific	
Cross-Site Request Forgery (CSRF)	Action-Specific	Authenticated Action
Cross-Site Request Forgery (CSRF)	Action-Specific	Unauthenticated Action
Application-Level Denial-of-Service (DoS)		
Application-Level Denial-of-Service (DoS)	App Crash	
Unvalidated Redirects and Forwards		
Unvalidated Redirects and Forwards	Open Redirect	
External Behavior		

v1.5- Oct. 1, 2018



Priority

OWASP Top Ten + Bugcrowd Extras

Specific Vulnerability Name

Variant or Affected Function

VARIES
CONTINUED

External Behavior

Browser Feature

External Behavior

Captcha Bypass

External Behavior

System Clipboard Leak

Insufficient Security Configurability

Insufficient Security Configurability

Weak Password Reset Implementation

Insufficient Security Configurability

Weak Registration Implementation

Insufficient Security Configurability

Weak 2FA Implementation

Using Components with Known Vulnerabilities

Using Components with Known Vulnerabilities

Captcha Bypass

Insecure Data Storage

Insecure Data Storage

Sensitive Application Data Stored Unencrypted

Insecure Data Storage

Server-Side Credentials Storage

Lack of Binary Hardening

Insecure Data Transport

Insecure Data Transport

Cleartext Transmission of Sensitive Data

Insecure Data Transport

Executable Download

Insecure OS/Firmware

Insecure OS/Firmware

Hardcoded Password

Broken Cryptography

Broken Cryptography

Cryptographic Flaw

Privacy Concerns

Privacy Concerns

Unnecessary Data Collection

Network Security Misconfiguration

Mobile Security Misconfiguration

Mobile Security Misconfiguration

SSL Certificate Pinning

Mobile Security Misconfiguration

Clipboard Enabled

Client-Side Injection

Binary

Client-Side Injection

Binary Planting

v1.5- Oct. 1, 2018

BUGCROWD'S
VRT

A NOTE FROM OUR SECURITY OPERATIONS TEAM

We believe in growth and transparency for security and bug bounty communities and see the release of our VRT as a tool that may help align expectations between researchers and program owners across ALL programs. Much of our employees' expertise in validating and rating thousands of submissions across hundreds of managed bounties is distilled into this document, making it a key component of Bugcrowd's managed services. Our internal VRT is a living document that changes constantly in response to discussions at our VRT Council, so specific severity ratings and notes are frequently updated.

As our first and foremost goal is usability, the VRT is not exhaustive. We believe that foregoing extreme technical depth for usability in creating such a community resource is a worthwhile tradeoff. We're confident that a security engineer using our VRT as a guide can triage and run a successful bug bounty program.

Happy Hunting,

Bugcrowd Security Operations Team

Follow us at [@Bugcrowd](#) and continue the discussion on [our forum](#).

UPDATES

0.1 - February 5, 2016

Original

0.2 - March 23, 2016

Divided the Cross-Site Scripting (XSS) entries to provide additional granularity that captures priority variations for XSS within applications with multiple user privilege levels.

0.4 - November 18, 2016

Minor priority changes, minor additions and subtractions, and typo fixes. Switched to a formal versioning system.

1.0 - February 24, 2017

Major changes to taxonomy structure with the addition of top-level categorizations to provide flexibility for context-dependent severity ratings. With this update we also launched our web-based taxonomy at [bugcrowd.com/vrt](#). Read more about it on our blog [here](#).

1.1 - May 5, 2017

Substantial additions, some priority changes, minor subtractions, and typo fixes. With this update we also released the open source taxonomy which can be found at [github.com/bugcrowd/vulnerability-rating-taxonomy](#). Read more about it on our blog [here](#).

1.2 - August 4, 2017

This update includes priority changes (most notable changes GET-based open redirects now set as P4, as well as all existing weak password policies as P5 "informational"), a few additions, and some minor modifications to increase the clarity of the taxonomy and align it with the security industry. Read more about it on our blog [here](#).

1.3 - September 28, 2017

Addition of VRT to CVSS v3 mapping as well as Broken Access Control category, aligned with the OWASP top 10 2017 release candidate. Revisions of VRT entries were made to provide better transparency for researchers and consistent triaging guidance. Read more about it on our blog [here](#).

1.4 - April 13, 2018

This release includes new entries that address missing, but commonly reported classes of issues, the removal of a few entries, and updated entry names to reduce ambiguity. Additionally, minor baseline severity rating adjustments were made along with increased granularity to some categories to assist our ASEs with more precise triage guidance. To submit suggested changes, edits, or additions to the VRT, use our open source taxonomy found at [github.com/bugcrowd/vulnerability-rating-taxonomy](#).

1.5 - October 1, 2018 (Current Version)

The latest version includes improving transparency by adding multiple entries for commonly reported issues. Additionally, aligning the baseline severity rating to best reflect the market by increasing taxonomy granularity. And lastly, we added designated variants for vulnerabilities that require Flash including some cases of XSS or open redirects. Read more about it on our blog [here](#).