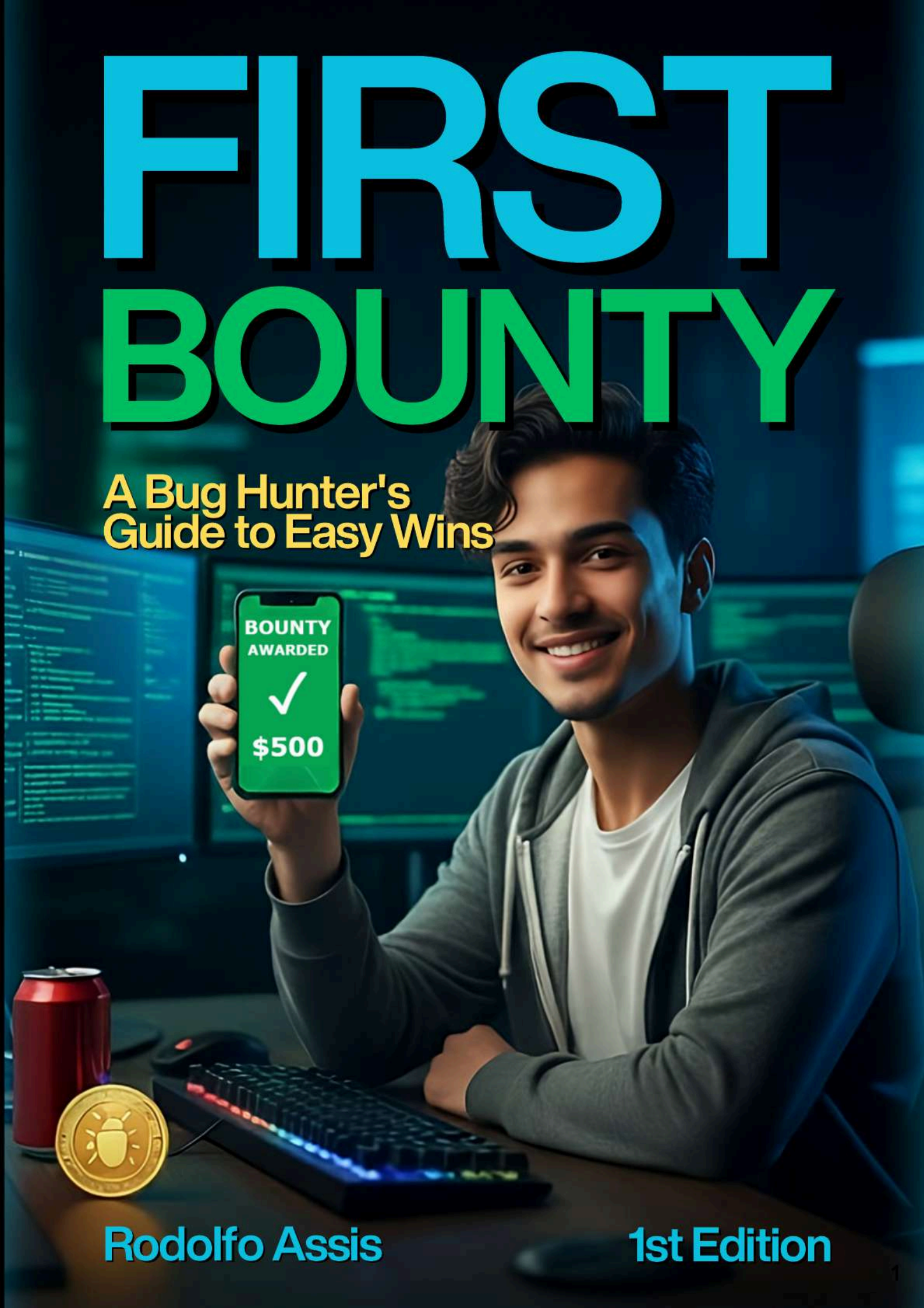
FIRST BOUNTY

A Bug Hunter's Guide to Easy Wins



Rodolfo Assis

1st Edition

About This Work

This work was created with the assistance of Artificial Intelligence (AI) technology. While thoroughly reviewed by the author, this publication may still contain minor errors or inaccuracies. It may contain estimated numbers instead of publicly credible ones (if available) and stories that may be fictional but are always based on real-world information. The technical content (especially code) is generally much more accurate but may also contain minor errors. However, in the author's opinion, this does not compromise the overall value and quality of the content presented.

If you find that you disagree with this assessment or are unsatisfied with the quality of this work, the author will gladly provide a full refund of your purchase. Your satisfaction is important, and feedback is always welcome.

Every revision of the edition you purchased is free for all purchasers. Updated versions will be emailed directly to the address used for payment. Alternatively, you will receive a notification email from the online ebook platform (such as Leanpub or Gumroad) where you purchased it once the revision becomes available.

If you discover any errors or have suggestions for improvements, please reach out via email or any other available communication channel. Your contributions help make this work better for all readers and are greatly appreciated.

About The Author

Rodolfo Assis (aka Brute Logic) is an independent cybersecurity researcher with 15+ years of experience in web application security. He is the creator of KNOXSS, the industry-leading automated XSS detection tool used by hundreds of security professionals worldwide, and has helped fix over 1,000 XSS vulnerabilities including discoveries in major companies like Oracle, Samsung, Uber, Apple, Amazon, and Microsoft.

Recognized as a Top 200 Global Cybersecurity Influencer by CheckPoint/Perimeter 81 and Top 20 by SourceDefense, Rodolfo has authored the widely-adopted Brute XSS Cheat Sheet series and is an international speaker at conferences including DEFCON and Ekoparty. He specializes in XSS testing and WAF evasion techniques, with a philosophy that "XSS is much more than just <script>alert(1)</script>."

Disclaimer

This work is provided for educational and informational purposes only. The techniques and information presented should be used ethically and legally, with proper authorization on systems you own or have explicit permission to test. The author is not responsible for any misuse of the information or any damages that may result from applying the techniques described. Use at your own risk and discretion.

First Bounty: A Bug Hunter's Guide to Easy Wins

By Rodolfo Assis

Introduction

Bug bounty hunting doesn't require genius-level technical skills or expensive tools. What it requires is systematic methodology, strategic target selection, and realistic expectations about timelines and competition.

This guide assumes you're prepared for a genuine learning curve. Bug bounty hunting requires developing systematic methodology, understanding business context, and building professional relationships - skills that take months to develop regardless of your technical background. Most readers will face weeks or months of testing without discoveries before achieving consistent results. This learning period is normal and necessary, not a sign of inadequate ability.

This guide cuts through the noise of "get rich quick" tutorials and social media success theater. Instead, it provides evidence-based strategies for finding your first legitimate security vulnerability and earning your first bounty payment.

Why This Guide Exists

The bug bounty landscape is saturated with contradictory advice. YouTube tutorials promise overnight success while Twitter feeds showcase $20,000 payouts without context. Meanwhile, beginners burn out after months of random testing with no clear methodology.

Most educational content falls into two categories: oversimplified tutorials that don't work on real applications, or advanced techniques requiring years of experience. This guide addresses the gap between theory and practice for hunters seeking their first success.

What You'll Actually Learn

This manual focuses on three vulnerability categories that consistently deliver results for systematic hunters:

IDOR (Insecure Direct Object References): Access control failures that appear in most web applications with user accounts. These vulnerabilities require minimal technical knowledge but systematic testing methodology.

Information Disclosure: Configuration files, debug pages, and exposed sensitive data that developers forget to secure. Discovery relies more on systematic reconnaissance than advanced exploitation.

Business Logic Flaws: Authentication bypasses, rate limiting issues, and workflow vulnerabilities that don't require complex technical exploitation but do require understanding business context.

These aren't the most glamorous vulnerability types, but they represent the majority of successful first bounties in documented cases.

What This Guide Is Not

This is not a comprehensive security reference. We don't cover advanced topics like SQL injection, remote code execution, or complex exploit chaining. These techniques require substantial background knowledge and rarely lead to first bounty success.

This is not a tool tutorial. While we recommend specific tools, the focus is on methodology and systematic approaches that work regardless of your technical setup.

This is not motivational content promising unrealistic outcomes. Success in bug bounty hunting requires sustained effort over months, not days or weeks.

Realistic Expectations

Based on analysis of documented beginner success stories and industry data:

Timeline: Most hunters require 6-8 months of consistent effort before their first accepted vulnerability report. Hunters dedicating 10+ hours weekly may achieve success faster, but 3-4 months remains optimistic even with intensive effort.

Competition: Popular programs receive hundreds of duplicate reports for common vulnerabilities within hours of launch. Success requires strategic program selection, not just technical capability.

Financial Reality: First bounties typically range from $100-500, not thousands. Sustainable income requires months of reputation building and access to private programs.

Learning Curve: Web application security has genuine complexity. Expect to invest substantial time understanding how modern applications work before finding meaningful vulnerabilities.

Your Strategic Advantage

While technical knowledge is learnable by anyone, strategic positioning creates sustainable competitive advantage:

Geographic Arbitrage: Programs from non-English speaking companies receive less attention from the predominantly English-speaking hunter community. This guide includes basic communication phrases for professional interaction in target markets.

ROI-Driven Methodology: Calculate time investment vs. bounty potential for each technique. IDOR testing averages 15 minutes per target with 8% success rate and $300 average payout - understanding these metrics guides efficient time allocation.

Relationship Building Systems: Professional communication templates and specific tactics for gaining security team attention and private program invitations. Most hunters ignore this crucial element.

Working Automation Scripts: Complete copy-paste scripts for reconnaissance and discovery, not just tool recommendations. Immediate competitive advantage through systematic efficiency.

Failure Pattern Analysis: Understanding why 90% of reports get rejected through specific examples of ineffective vs. successful reports for identical vulnerability types.

These strategic elements, combined with systematic technical methodology, create repeatable success patterns rather than depending on luck.

How to Use This Guide

Each chapter builds systematically toward your first bounty:

Foundation Building (Chapters 1-2): Establishes realistic mindset, tool setup, and strategic target selection before any technical testing.

Technical Methodology (Chapters 3-4): Provides step-by-step testing procedures for high-success-rate vulnerability types.

Competitive Positioning (Chapters 5-6): Covers automation, specialization, and professional development for sustainable long-term success.

The content assumes no prior security experience but expects commitment to systematic learning and persistent effort.

Success Definition

Your first bounty represents more than financial compensation. It validates that you can:

- Identify security vulnerabilities in production applications
- Document findings clearly enough for developer understanding
- Communicate professionally with security teams
- Execute systematic methodology that produces repeatable results

This validation opens pathways to security consulting, full-time security roles, and continued bug bounty success.

The techniques in this guide come from analysis of documented successful discoveries, industry research, and strategic frameworks proven effective in competitive environments.

Ready to begin? Your first bounty is a systematic process away.

Table of Contents

Chapter 1: The Hunter's Foundation

Building Sustainable Skills and Realistic Expectations

- Why Easy Wins Require Systematic Methodology
- Honest Timeline Expectations: The 6-8 Month Reality
- Tool Setup That Professionals Actually Use
- Your First 30 Days: Structured Learning Roadmap
- Motivation Framework: Process Over Outcomes
- Q&A: "Am I Technical Enough?" and Other Foundation Concerns

Chapter 2: Strategic Target Selection

Competitive Positioning Through Smart Program Choice

- The Blue Ocean Strategy: Programs Others Avoid
- Geographic Targeting: Non-English Market Opportunities
- Timing Optimization: The 48-Hour Launch Window
- Program Intelligence: Research Before Testing
- Red Flags: When to Move On vs. Persist
- Pipeline Management: Sustainable Target Flow
- Q&A: "Why Can't I Find Bugs Anywhere?" and Selection Strategy

Chapter 3: The Big Three Easy Wins

Systematic Methodology for High-Success Vulnerabilities

- IDOR Discovery: The 15-Minute Testing Framework
- Information Disclosure: Automated Discovery Workflows
- Business Logic Flaws: Context-Aware Testing Methodology
- Business Impact Assessment for Each Vulnerability Type
- Success Rate Expectations and Realistic Timelines
- Q&A: "Should I Focus on One Vulnerability Type?" and Specialization Decisions

Chapter 4: Business Logic and Simple Flaws

Non-Technical Vulnerabilities with High Impact Potential

- Authentication Bypass Patterns and Systematic Testing
- Rate Limiting: Identification and Exploitation Workflows
- File Upload Vulnerabilities: Beyond Basic Extension Checks
- Session Management: Gap Detection and Testing Procedures

- Price Manipulation and Workflow Bypass Techniques
- Q&A: "How Do I Know If Something Is Actually a Vulnerability?" and Impact Assessment

Chapter 5: Low-Hanging Fruit Hunting

Automation and Specialization for Competitive Advantage

- Reconnaissance Automation: Time-Saving Discovery Scripts
- OSINT Methodology: Context and Attack Surface Expansion
- Default Credentials: Systematic Testing Beyond Common Lists
- API Discovery: Beyond Basic Directory Enumeration
- Forgotten Assets: Subdomain Archaeology and Integration Points
- Specialization Pathways: Geographic, Technology, and Industry Focus
- Q&A: "How Do I Differentiate Myself?" and Competitive Positioning

Chapter 6: From Bug to Bounty

Professional Development and Long-Term Success

- Report Writing: Templates and Communication Strategies
- Security Team Relationships: Professional Interaction Guidelines
- Reputation Building: Private Program Access Strategies
- Career Progression: Beyond Bug Bounties to Security Roles
- Sustainability Planning: Avoiding Burnout and Financial Instability
- Q&A: "What's Next After My First Bounty?" and Career Development

---

Estimated Reading Time: 3-4 hours
Practical Application Time: 6-8 months for first bounty
Long-term Value: Foundation for cybersecurity career development