1. Strengthen Your Foundations (1–2 months)

1. Computer & Network Basics
   * Operating Systems: Familiarize yourself with Linux (Kali/Parrot) and Windows internals.
   * Networking: Understand TCP/IP, DNS, HTTP/S, cookies, sessions, and common ports.
   * Programming/Scripting: Have at least one language under your belt—Python is hugely helpful for automation.
2. Web Fundamentals
   * HTML, CSS, JavaScript basics
   * How forms work, URL parameters, headers, JSON, XML, API calls
3. Security Fundamentals
   * Read OWASP Top 10 inside out
   * Learn basic cryptography concepts (hashes, encryption vs. hashing)

2. Set Up Your Toolkit (1–2 weeks)

* Kali Linux / Parrot (VM or dual-boot)
* Burp Suite Community Edition (upgradable to Pro later)
* Browser plugins: FoxyProxy, Wappalyzer, HTTP Headers
* CLI tools: Nmap, Netcat, Gobuster/Dirbuster, cURL, sqlmap
* Scripting environment: Python + Requests/BeautifulSoup

3. Hands-On Learning & Labs (2–3 months)

1. PortSwigger Web Security Academy
   * Free, interactive labs covering SQLi, XSS, SSRF, auth flaws, etc.
2. TryHackMe / HackTheBox
   * Start with “Beginner” or “Web Fundamentals” pathways on TryHackMe
   * Move on to “Web” machines on HTB when comfortable
3. OWASP Juice Shop & DVWA
   * Deploy locally and practice exploiting every vulnerability
4. CTFs & Challenges
   * Sites: VulnHub, CTFtime, PentesterLab

4. Dive Into Bug Bounty Platforms (ongoing)

1. Beginner-Friendly
   * HackerOne Hacktivity: Start with public programs like GitHub, Shopify
   * Bugcrowd: University & Introductory programs
   * Intigriti: Good for EU-based targets
2. Open-Submission
   * OpenBugBounty, YesWeHack — no NDA needed, easy to start
3. Private Programs
   * As you rack up reports & reputation, you’ll get invited to private scopes (higher payouts)

Tip: Always read each program’s scope, rules, and “Triaging” guidelines before poking things.

5. Build Your Workflow & Reporting Skills (ongoing)

1. Recon & Enumeration
   * Subdomain enumeration (Amass, Sublist3r)
   * Endpoint discovery (Burp Intruder, Gobuster)
2. Exploitation Methodology
   * Follow consistent checklists: injection → auth → logic → XSS → CSRF → SSRF → RCE
3. Report Writing
   * Clear title, proof-of-concept steps, impact statement, remediation advice
   * Include screenshots, request/response snippets, short video for complex flows

6. Level Up: Advanced Topics (3–6 months+)

* API Security: OAuth, JWT, GraphQL
* Mobile App Testing: Frida, MobSF
* Client-Side Logic Flaws: WebAssembly, single-page apps
* Automated Scanning & Burp Extensions: Write your own Burp plugins in Python/Java
* Server-Side: Docker container escapes, SSRF to RCE on clouds

7. Community, Mentorship & Continuous Growth

* Twitter/X & Discord: Follow top hunters (e.g., @*abatchy*, @JonLyonSec)
* Blog & Tweet Your Write-Ups: Build reputation, get feedback
* Discord Servers: Bug Bounty Forum, PortSwigger Community
* Conferences & Webinars: watch/review talks from Black Hat, DEF CON, OWASP

8. Certifications & Career Path (optional but helpful)

* eJPT (eLearnSecurity Junior Pentester) – solid intro credential
* OSCP – if you want to pivot into formal penetration testing
* Offsec Web Expert (OSWE) – deep dive into web app hacking

Sample 6-Month Timeline

| Month | Focus |
| --- | --- |
| 1–2 | Foundations & toolkit setup |
| 3 | PortSwigger labs + TryHackMe beginner |
| 4 | OWASP Juice Shop + first CTFs |
| 5 | Submit 1–2 low-hanging bugs on H1/BC |
| 6 | Deep dive advanced topics + private scopes |

Key Tips for Beginners

* Start small: find low-severity bugs (info-leaks, XSS) to build confidence
* Stay organized: use a spreadsheet or a tool like Burp Collaborator for tracking
* Learn from write-ups: study disclosed reports on HackerOne Hacktivity
* Be patient & persistent: bug hunting is a marathon, not a sprint

With consistent practice, you’ll not only earn bounties but also develop rock-solid offensive security skills that carry over into any pentesting or security role.