Bug Bounty Roadmap:

PART 1

Mastering the Basics!

Before embarking on your bug bounty journey, it's essential to establish a solid grasp of the foundational elements that underpin the world of cybersecurity. This section lays the groundwork for your exploration, ensuring you have the necessary knowledge to navigate the intricate web of networks, systems, and programming languages.

To effectively engage in bug bounty hunting and ethical hacking, a firm grasp of the fundamental building blocks is crucial. Begin your journey by acquainting yourself with the following key concepts:

Understanding Network, Web, and Communication Basics

Network Basics:

Acquire a basic understanding of networking principles, an essential knowledge for anyone delving into the realm of computers. Explore resources such as

- Networking Basics: What You Need to Know (CISCO)
- The Fundamentals of Networking (IBM)
- Basics of Computer Networking (Geeks for Geeks)
- Computer Networking Complete Course – Basic to Advanced (9 Hours YouTube Course)
- Fundamentals of computer networking (Microsoft)

Web:

For an overview of the web, you should give a read to any two of these. These will not only refresh your web basic fundamentals but also prepare you for what's coming ahead.

- Web – Basic Concepts (Tutorials Point)
- Web Fundamentals (Google Developers)
- Web Basics and Overview (Kent State University)

Communication Protocols:

In order to learn something, you must learn how it works and how data is exchanged within or between computers. In our case how an application works and what its flow is we need to learn how it communicates with you. For that purpose, I believe you must go through the following list to understand Network Protocols and their uses.

- Communication protocols (Wikipedia)
- Official Internet Protocol Standards (RFC Editor)
- MDN Web Docs Glossary: Definitions of Web-related terms (Mozilla)
- HTTP (Mozilla)
- HTTP Related Protocols (W3.org)
- Types of Network Protocols and Their Uses (W3 Schools)

Database:

You must learn about Database basics and understand it as this is one of the crucial parts of what you'll gonna be attacking as a hacker in many cases.

- Basics Of DBMS (Toppr)
- Database basics (Oracle)
- Database Basics: Concepts & Examples for Beginners (Lido)

Choose an Operating System:

According to Eric Steven Raymond, "**The single most important step any newbie can take toward acquiring hacker skills is to get a copy of Linux or one of the BSD-Unixes, install it on a personal machine, and run it. Trying to learn to hack on a Microsoft Windows machine or under any other closed-source system is like trying to learn to dance while wearing a body cast.**"

Whichever OS you choose, ensure to familiarize yourself with essential commands through cheat sheets like this below:

http://linuxcommand.org/

Essential Windows CMD Commands You Should Know

ASSOC	Displays or modifies file extension associations
ATTRIB	Displays or changes file attributes
BREAK	Sets or clears extended CTRL+C checking
BCDBOOT	Used to copy critical files to the system partition and to crate a new system BCD store
BCDEDIT	Sets properties in boot database to control boot loading
CACLS	Shows or changes access control lists (ACLs) of files
CALL	Calls a batch program from another
CD	Shows the name of or changes to a current directory
CHCP	Displays or sets the active code page number
CHDIR	Displays the name of or changes to the current directory
CHKDSK	Checks a disk and displays a status report
CHKNTFS	Displays or modifies the checking of disk at boot time
CHOICE	Batch file command that allows users to select from a set of options
CIPHER	Displays or alters the encryption of directories (files) on NTFS partitions
CLIP	Redirects output off another command to the Windows clipboard
CLS	Clears the screen
CMD	Starts a new instance of the Windows command interpreter
CMDKEY	Creates, lists, and deletes stored user names and passwords or credentials
COLOR	Sets the default console colors
COMP	Compares the contents of two files or sets of files byte-by-byte
COMPACT	Displays or alters the compression of files on NTSF partitions
CONVERT	Converts FAT volumes to NTFS. You cannot convert the current drive
COPY	Copies one of more files to another location
DATE	Displays or sets the date
DEFRAG	Disk defragment accessory
DEL	Deletes one or more files
DIR	Displays a list of files and sub-directories in a directory
DISKCOMP	Compares the contents of two floppy disks
DISKCOPY	Copies the contents of one floppy disk to another
DISKPART	Displays or configures Disk Partition properties
DOSKEY	Edits command lines, recalls Windows commands, and creates macros
DRIVERQUERY	Displays current device driver status and properties
ECHO	Displays messages, or turns commands echoing on or off
ENDLOCAL	Ends localization of environment changes in a batch file
ERASE	Deletes one of more files
EXIT	Quits and closes the command shell
EXPAND	Expands compressed files
FC	Compares two files or sets of files, and displays the differences between them
FIND	Searches for a text string in a file or files
FINDSTR	Searches for strings in files
FOR	Runs a specified command for each item in a set
FORFILES	Selects files in a folder for batch processing
FORMAT	Formats a disc for use with Windows
FSUTIL	Displays or configures the file system properties
FTYPE	Displays or modifies file types used in file extensions associations
GOTO	Directs the Windows command interpreter to a labeled line in a batch program
GPRESULT	Displays Group Policy Information for machine or user.
GRAFTABL	Enables Windows to display an extended character set in graphics mode
HELP	Provides help information for Windows commands
ICACLS	Display, modify, backup, or restore ACLs for files and directories
IF	Performs conditional processing in batch programs.
IPCONFIG	Displays all current TCP/IP network configuration values
LABEL	Creates, changes, or deletes the volume label of a disk
MD	Creates a directory
MKDIR	Creates a directory
MKLINK	Creates Symbolic Links and Hard Links
MODE	Configures a system device
MORE	Displays output one screen at a time
MOVE	Moves one or more files from one directory to another directory
NETSTAT	Returns a list of currently open ports and related IP addresses
OPENFILES	Queries, displays, or disconnects open files or files opened by network users
PATH	Displays or sets a search path for executable files
PATHPING	Advanced version of ping, used if there are multiple routers between your PC and the device you're testing
PAUSE	Suspends processing of a batch file
PING	Sends a series of test packets to the specified address
POPD	Restores the previous value of the current directory saved by PUSHD
POWERCFG	Manages and tracks energy utilization and power consumption
PRINT	Prints a text file
PROMPT	Changes the Windows command prompt
PUSHD	Saves the current directory then changes it
RD	Removes a directory
RECIMG	Configures the custom Windows recovery image
RECOVER	Recovers readable information from a bad or defective disk
REM	Designates comments (remarks) in batch files
REN	Renames a file or files
RENAME	Renames a file or files
REPLACE	Replaces files
RMDIR	Removes a directory
ROBOCOPY	Advanced utility to copy files and directory trees
SET	Displays, sets, or removes environment variables for current session
SETLOCAL	Begins localization of environment changes in a batch file
SETX	Sets environment variables
SFC	Finds corrupt/missing files, replaces them with cached copies
SC	Displays or configures services (background processes)
SCHTASKS	Schedules commands and programs to run on a computer
SHIFT	Shifts the position of replaceable parameters in batch files
SHUTDOWN	Allows proper local or remote shutdown of machine
SORT	Sorts input
START	Starts a separate window to run a specified programs or command
SUBST	Associates a path with a drive letter
SYSTEMINFO	Displays machine specific properties and configuration
TAKEOWN	Allows an administrator to take ownership of a file
TASKLIST	Displays all currently running tasks including services
TASKKILL	Kill running process or applications
TIME	Displays or sets the system time
TIMEOUT	Pauses the command processor for the specified number of seconds
TITLE	Sets the window title for a CMD.EXE session
TRACERT	Returns information about each step in the route between your PC and the target
TREE	Graphically displays the directory structure of a drove or path
TYPE	Displays the contents of a text file
VER	Displays the Windows version
VERIFY	Tells Windows whether or verify that your files are written correctly to a disk
VOL	Displays a disk volume label and serial number
VSSADMIN	Volume Shadow Copy Service administration tool
WHERE	Displays the locations of files that match a search pattern
WMIC	Displays WMI information inside interactive command shell
XCOPY	Copies files and directory trees

- http://linuxcommand.org/

- https://helpdeskgeek.com/help-desk/21-cmd-commands-all-windows-users-should-know/

Coding Proficiency: The Path to Mastery:

While becoming a proficient programmer might not be mandatory, having a solid understanding of programming languages is undeniably beneficial in the realm of bug bounty hunting.

I personally suffered for two years in bug bounties because in many cases I couldn't really understand what the particular code meant, couldn't exploit an issue properly, or couldn't even code in general, and I'm, still trying my best to catch up to speed so I'll suggest you guys not to skip these parts.

Strengthen your coding skills with the following languages:

HTML:

- HTML Tutorial (W3 Schools)
- Learn HTML (Code Academy)

PHP:

- PHP Tutorial (W3 Schools)
- Learn PHP (Code Academy)

JavaScript:

- Learn JavaScript – Full Course for Beginners (freecodecamp.org)
- Learn JavaScript (Code Academy)
- Build anything you want with JavaScript (learnjavascript.today)

SQL (Structured Query Language):

- SQL Tutorial – Full Database Course for Beginners (freeCodeCamp.org)
- SQL Tutorial (W3 Schools)
- Learn SQL (Code Academy)

Java:

- Learn Java (Code Academy)
- Java | How to start learning Java (Geeks for Geeks)

- Learn Java Online (learnjavaonline.org)
- Java Beginner Course – Get Started Coding with Java! (freeCodeCamp.org)

C/C++

- C/C++ Full Course Playlist (freeCodeCamp.org)
- Learncpp (Learncpp.com)
- Learn C++ (Code Academy)

What You'll learn from these is not just Programming languages but the proper way of web and systems to communicate that you gonna test or build. I'm also a student in Programming so sharing the resources I'm currently following.

Embrace Automation:

"Never send a human to do a machine's job"

To truly excel in the world of bug bounty hunting, mastering automation is essential. Automation empowers you to work faster, more efficiently, and continuously while reducing repetitive tasks.

Have a look at the slides below and read an awesome article on "**Conference notes: Automation for Bug Hunters (Bug Bounty Talks)**"

Strengthen your automation capabilities with these languages, If you can grasp hold on to one or more of the following languages you can easily & very happily automate your work and earn in a better way.

Python:

- Real Python Tutorials (Real Python)
- Hacking with Python – 7 Best online courses for ethical hacking (By AIMEE O'DRISCOLL)
- The Python Tutorial (Python.org)

Bash:

- 10 Best Linux Shell Scripting Tutorials for Beginners (Quick Code)
- Learn Shell (learnshell.org)
- Shell Scripting Tutorial (Tutorials Point)

Golang:

- Learn GO (go.dev)
- Learn Golang (Code Academy)
- Go Tutorial (Tutorials Point)

Ruby:

- Learn Ruby (Code Academy)
- Learn Ruby Online (learnrubyonline.org)

PART 2

Learning About Vulnerabilities

This part is all about building your skills, learning about how to identify weaknesses, and arming yourself with the tools to become a bug bounty hunter. Choosing the right path to start in Bug Bounty is very important. Your choice should align with your interests and aspirations. While some opt for the Web Application route due to its approachable nature, others may delve into the realm of Mobile. Here, I'll be focusing on Web and Mobile paths, reflecting my own area of expertise.

The Web Application Security Path:

The Web Application path is a popular starting point due to its accessible nature. Begin by understanding the intricacies of web applications and the vulnerabilities they can harbor. Resources like:

- OWASP Top Ten Project
- Web Application Security Basics (Mozilla)
- PortSwigger Web Security Academy

Equip you with the foundational knowledge and insights needed to navigate this domain.

The Mobile Application Security Path:

For those intrigued by the mobile landscape, the Mobile path beckons. Immerse yourself in the world of mobile application security, uncovering potential vulnerabilities that lurk within. Key resources such as:

- OWASP Mobile Security Project
- Android Security Documentation
- iOS Security Documentation

Will serve as your guiding beacons, leading you through the intricate mobile security landscape.

Key Resources:

The Platforms below should be your first stop toward learning about security.

- **HackerOne** Hacker101
- **Bugcrowd** University
- **Intigriti** Hackademy

These platforms offer a wealth of resources and lectures that can significantly enhance your learning journey. They provide invaluable insights, often surpassing what I might share here.

Exploring Web Application Security: Building Your Foundation

In this phase, we're delving into the exciting world of exploring Web Application Security.

Recommended Books and Guides: Building Your Expertise

To fortify your understanding of Web Application Penetration Testing and Security, delve into the following essential resources:

- Mastering Modern Web Penetration Testing
- The Hacker's Underground Handbook
- Web Hacking 101
- The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws
- The Tangled Web: A Guide to Securing Modern Web Applications
- OWASP Testing Project

These resources offer comprehensive insights into the intricacies of web application penetration testing and security assessment.

Embrace OWASP:

Make it a priority to familiarize yourself with the OWASP Testing Guide and OWASP Top 10 Vulnerabilities. These invaluable references provide guidance and understanding:

- OWASP Testing Guide
- OWASP Top 10 for 2023
- OWASP Top 10 for 2017

These resources provide a solid foundation for comprehending common vulnerabilities and security practices.

Exploring Common Web Application Vulnerabilities

This is a crucial phase of your bug bounty journey, where we learn about common web application vulnerabilities that you're likely to encounter while hunting for bugs. In this section, my focus is on providing you with valuable resources to understand and learn about these vulnerabilities effectively.

Cross-Site Request Forgery (CSRF)

Cross-Site Request Forgery is a potent attack that exploits the trust a web application has in the authenticated user's browser. By coercing the user into unknowingly performing actions they didn't intend, the attacker can manipulate the application's functionalities and wreak havoc.

Delve Deeper with These Resources

- Imperva: Understanding CSRF – Cross-Site Request Forgery
- OWASP Cross-Site Request Forgery (CSRF)
- Netsparker Blog: Demystifying CSRF – Cross-Site Request Forgery

Uncover Real-World Scenarios:

- CSRF Account Takeover famebit by Hassan Khan
- Hacking PayPal Accounts with one click (Patched) by Yasser Ali
- Add tweet to collection CSRF by vijay kumar
- Facebookmarketingdevelopers.com: Proxies, CSRF Quandry and API Fun by phwd
- How i Hacked your Beats account ? Apple Bug Bounty by @aaditya_purani

- Story of a weird CSRF bug by Sudhanshu Rajbhar
- Bumble Full account takeover using CSRF by Mahmoud G
- Uber CSRF Account Takeover by Ron Chan
- Messenger.com CSRF that show you the steps when you check for CSRF by Jack Whitton

Cross-Site Scripting (XSS)

Cross-Site Scripting, commonly known as XSS, empowers malicious actors to inject client-side scripts into web pages, potentially compromising the security of other users who view those pages. These scripts can execute in a victim's browser, leading to unauthorized actions, data theft, or the spread of malware.

Resources for Deepening Your Knowledge:

- OWASP Cross-site Scripting (XSS)
- PortSwigger Web Security: Cross-site Scripting
- Excess XSS: A Comprehensive Tutorial on Cross-Site Scripting

Practical Examples and Proof of Concepts:

- AirBnb Bug Bounty: Turning Self-XSS into Good-XSS #2 by geekboy
- Uber Self XSS to Global XSS
- How I found a $5,000 Google Maps XSS (by fiddling with Protobuf) by Marin MoulinierFollow
- XSSI, Client Side Brute Force
- postMessage XSS Bypass
- XSS in Uber via Cookie by zhchbin
- Stealing contact form data on www.hackerone.com using Marketo Forms XSS with postMessage frame-jumping and jQuery-JSONP by frans
- XSS due to improper regex in third party js Uber 7k XSS
- XSS in TinyMCE 2.4.0 by Jelmer de Hen
- Pass uncoded URL in IE11 to cause XSS
- Twitter XSS by stopping redirection and javascript scheme by Sergey Bobrov

- Years ago Google xss
- xss in Yahoo Mail Again, worth $10000 by Klikki Oy
- Google Account Takeover
- God-like XSS, Log-in, Log-out, Log-in in Uber by Jack Whitton
- Three Stored XSS in Facebook by Nirgoldshlager
- Using a Braun Shaver to Bypass XSS Audit and WAF by Frans Rosen
- An XSS on Facebook via PNGs & Wonky Content Types by Jack Whitton
- Stored XSS in *.ebay.com by Jack Whitton
- Complicated, Best Report of Google XSS by Ramzes
- Tricky Html Injection and Possible XSS in sms-be-vip.twitter.com by secgeek
- Command Injection in Google Console by Venkat S
- Stored XSS on developer.uber.com via admin account compromise in Uber by James Kettle (albinowax)
- Yahoo Mail stored XSS by Klikki Oy
- Abusing XSS Filter: One ^ leads to XSS(CVE-2016-3212) by Masato Kinugawa
- Youtube XSS by fransrosen
- Google XSS subdomain Clickjacking

SQL Injection

SQL injection is a common attack vector that uses malicious SQL code for backend database manipulation to access information that was not intended to be displayed.

Resources for Deepening Your Knowledge:

For a comprehensive grasp of SQL Injection, these resources are your go-to:

- OWASP SQL Injection
- PortSwigger Web Security: SQL Injection
- W3Schools: SQL Injection

Real-Life Scenarios: Proof of Concepts

- SQL Injection Vulnerability nutanix by Muhammad Khizer Javed

- Multiple vulnerabilities in a WordPress plugin at drive.uber.com by Abood Nour (syndr0me)
- GitHub Enterprise SQL Injection by Orange
- SQL injection in WordPress Plugin Huge IT Video Gallery in Uber by glc
- SQL Injection on sctrack.email.uber.com.cn by Orange Tsai

Remote Code Execution (RCE)

Remote Code Execution (RCE) is a formidable technique that grants attackers the power to execute their own code on a victim's system. Imagine the potential havoc if a malevolent actor gains control over a machine, enabling them to manipulate it at will.

Resources for Deepening Your Knowledge:

To truly comprehend and master RCE, these references will serve as your compass:

- Netsparker: Remote Code Execution and Evaluation
- Wikipedia: Arbitrary Code Execution

Practical Examples and Proof of Concepts:

- How we broke PHP, hacked Pornhub and earned $20,000 by Ruslan Habalov
- WordPress SOME bug in plupload.flash.swf leading to RCE in Automatic by Cure53 (cure53)
- Read-Only user can execute arbitraty shell commands on AirOS by 93c08539 (93c08539)
- Remote Code Execution by impage upload! by Raz0r (ru_raz0r)
- Popping a shell on the Oculus developer portal by Bitquark
- Crazy! PornHub RCE AGAIN!!! How I hacked Pornhub for fun and profit – 10,000$ by 5haked
- PayPal Node.js code injection (RCE) by Michael Stepankin
- eBay PHP Parameter Injection lead to RCE
- Yahoo Acqusition RCE
- Command Injection Vulnerability in Hostinger by @alberto__segura
- RCE in Imgur by Command Line

- RCE in git.imgur.com by abusing out dated software by Orange Tsai
- RCE in Disclosure
- Remote Code Execution by struct2 Yahoo Server
- Command Injection in Yahoo Acquisition
- Paypal RCE
- $50k RCE in JetBrains IDE
- JDWP Remote Code Execution in PayPal by Milan A Solanki
- XXE in OpenID: one bug to rule them all, or how I found a Remote Code Execution flaw affecting Facebook's servers by Reginaldo Silva
- How I Hacked Facebook, and Found Someone's Backdoor Script by Orange Tsai
- How I Chained 4 vulnerabilities on GitHub Enterprise, From SSRF Execution Chain to RCE! by Orange Tsai
- uber.com may RCE by Flask Jinja2 Template Injection by Orange Tsai
- Yahoo Bug Bounty – *.login.yahoo.com Remote Code Execution by Orange Tsai (in Chinese)
- Google App Engine RCE by Ezequiel Pereira
- Exploting ImageMagick to get RCE on HackerOne by c666a323be94d57
- Trello bug bounty: Access server's files using ImageTragick by Florian Courtial
- 40k fb RCE
- Yahoo Bleed 1
- Yahoo Bleed 2
- Microsoft Apache Solr RCE Velocity Template By Muhammad Khizer Javed

Insecure Direct Object Reference (IDOR)

In IDOR an application provides **direct** access to **objects** based on the user-supplied input. As a result of this vulnerability, attackers can bypass authorization and access resources in the system directly.

Guiding Lights: References for Clarity

Embark on a journey to understand and combat IDOR with these invaluable references:

- Bugcrowd: How to Find IDOR (Insecure Direct Object Reference) Vulnerabilities
- OWASP: Testing for Insecure Direct Object References (OTG-AUTHZ-004)
- Secjuice: IDOR – Insecure Direct Object Reference Definition

Real-World Glimpses: Proof of Concepts

Dive into real-world demonstrations of IDOR's potential impact:

- DOB disclosed using "Facebook Graph API Reverse Engineering" by Raja Sekar Durairaj
- Change the description of a video without publish_actions permission in Facebook by phwd
- Leak of all project names and all user names , even across applications on Harvest by Edgar Boda-Majer (eboda)
- Changing paymentProfileUuid when booking a trip allows free rides at Uber by Matthew Temmy (temmyscript)
- View private tweet
- Delete FB Video
- Facebook Page Takeover by Manipulating the Parameter by arunsureshkumar
- Classic IDOR endpoints in Twitter
- Mass Assignment, Response to Request Injection, Admin Escalation by sean
- Change any user's password in Uber by mongo
- Microsoft-careers.com Remote Password Reset by Yaaser Ali
- How I could change your eBay password by Yaaser Ali
- Duo Security Researchers Uncover Bypass of PayPal's Two-Factor Authentication by Duo Labs
- Hacking Facebook.com/thanks Posting on behalf of your friends! by Anand Prakash
- How I got access to millions of [redacted] accounts
- All Vimeo Private videos disclosure via Authorization Bypass with Excellent Technical Description by Enguerran Gillier (opnsec)

- Urgent: attacker can access every data source on Bime by Jobert Abma (jobert)
- Downloading password protected / restricted videos on Vimeo by Gazza (gazza)
- Get organization info base on uuid in Uber by Severus (severus)
- How I Exposed your Primary Facebook Email Address (Bug worth $4500) by Roy Castillo

HTTP Request Smuggling

HTTP request smuggling is a technique for interfering with the way a web site processes sequences of HTTP requests that are received from one or more users. Request smuggling vulnerabilities are often critical in nature, allowing an attacker to bypass security controls, gain unauthorized access to sensitive data, and directly compromise other application users.

Resources for Deepening Your Knowledge:

For a comprehensive grasp of Request Smuggling these resources are your go-to:

- HTTP request smuggling PortSwigger
- What Is HTTP Request Smuggling?
- HTTP Request Smuggling – The Ultimate Guide

Real-Life Scenarios: Proof of Concepts

- Finding My First Bug: HTTP Request Smuggling
- HTTP Request Smuggling on api.flocktory.com Leads to XSS on Customer Sites
- HTTP Request Smuggling on business.apple.com and Others.

Web Cache Deception

Web Cache Deception (WCD) is an attack in which an attacker deceives a caching proxy into improperly storing private information sent over the internet and gaining unauthorized access to that cached data. It was proposed by Omer Gil, a security researcher in 2017.

Resources for Deepening Your Knowledge:

For a comprehensive grasp of WCD these resources are your go-to:

- Web Cache Deception

- Path confusion: Web cache deception threatens user information online
- What is Web Cache Deception?
- Cache Poisoning and Cache Deception

Real-Life Scenarios: Proof of Concepts

- ChatGPT Account Takeover – Nagli
- How I Made $16,500 Hacking CDN Caching Servers — Part 1 by bombon
- Web Cache Deception Attack on a private bug bounty program by Snoopy
- Web Cache Deception Attack leads to user info disclosure by Kunal pandey

Unrestricted File Upload

As in the name unrestricted file upload allows user to upload malicious file to a system to further exploit to for Code execution. Think of Unrestricted File Upload as an unlocked gate allowing unauthorized files to infiltrate an application. This vulnerability lays the foundation for attackers to upload and manipulate files, potentially gaining unauthorized control over a system.

Illuminating Your Path: References for Understanding

Navigate this vulnerability's landscape with the help of these guiding references:

- Netsparker: Unrestricted File Upload Vulnerability
- OWASP: Unrestricted File Upload
- Hacking Articles: 5 Ways File Upload Vulnerability Exploitation

Journey into the Wild: Real-world Examples

- File Upload XSS in image uploading of App in mopub by vijay kumar
- File Upload XSS in image uploading of App in mopub in Twitter by vijay kumar (vijay_kumar1110)
- Unrestricted File Upload to RCE by Muhammad Khizer Javed

XML External Entity Attack (XXE)

XXE is an attack against an application that parses XML input. This attack occurs when XML input containing a reference to an external entity is processed by a weakly configured XML parser.

Guiding Light: Resources for XXE

Embark on your journey of understanding XXE attacks with these guiding references:

- PortSwigger: XXE
- OWASP: XML External Entity (XXE) Prevention Cheat Sheet
- InfoSec by Phonexicum: Demystifying XXE

Real-world Examples:

- XXE through SAML
- XXE in Uber to read local files
- XXE by SVG in community.lithium.com
- How we got read access on Google's production servers by detectify
- Blind OOB XXE At UBER 26+ Domains Hacked by Raghav Bisht

Local File Inclusion (LFI)

The File Inclusion vulnerability allows an attacker to include a file, usually exploiting a "dynamic file inclusion" mechanisms implemented in the target application. The vulnerability occurs due to the use of user-supplied input without proper validation.

Guiding Light: Resources for LFI

Navigate the LFI terrain armed with these guiding references:

- OWASP: Testing for Local File Inclusion
- Netsparker: Local File Inclusion Vulnerability
- Medium: Local File Inclusion Web Application Penetration Testing

Real-world Examples:

- SSRF to LFI
- Disclosure Local File Inclusion by Symlink
- Facebook Symlink Local File Inclusion
- Gitlab Symlink Local File Inclusion
- Gitlab Symlink Local File Inclusion Part II
- One Cloud-based Local File Inclusion = Many Companies affected

- LFI by video conversion, excited about this trick!

Subdomain Takeover

A subdomain takeover occurs when an attacker gains control over a subdomain of a target domain. Typically, this happens when the subdomain has a canonical name (CNAME) in the Domain Name System (DNS), but no host is providing content for it.

Guiding Lights: Sources of Wisdom

Navigate the intricate landscape of Subdomain Takeover armed with these enlightening references:

- Security Breached: Unveiling the Subdomain Takeover Vulnerability
- Subdomain Takeover Basics: Insights from 0xpatrik
- Can I Take Over .xyz? Exploring Subdomain Takeover with EdOverflow

Real-world Examples:

- Hijacking tons of Instapage expired users Domains & Subdomains by geekboy
- Reading Emails in Uber Subdomains
- Slack Bug Journey – by David Vieira-Kurz
- Subdomain takeover and chain it to perform authentication bypass by Arne Swinnen
- UBER Wildcard Subdomain Takeover by Muhammad Khizer Javed
- Lamborghini Subdomain Takeover Through Expired Cloudfront Distribution by Muhammad Khizer Javed
- Subdomain Takeover via Unsecured S3 Bucket Connected to the Website by Muhammad khizer Javed

Server Side Request Forgery (SSRF)

Server-side request forgery (also known as SSRF) is a web security vulnerability that allows an attacker to induce the server-side application to make requests to an unintended location. In a typical SSRF attack, the attacker might cause the server to make a connection to internal-only services within the organization's infrastructure. In other cases, they may be able to force the server to connect to arbitrary external systems, potentially leaking sensitive data such as authorization credentials.

Guiding Beacons: Resources for SSRF

Learn from these SSRF with these guiding references:

- SSRF: Types and Ways to Exploit It – Part 1
- OWASP's Guide to Server Side Request Forgery
- Deciphering the SSRF Vulnerability
- Unmasking the Veil: What is Server Side Request Forgery (SSRF)?

Casting Light: Real-world Examples

Peer through the shadows with these real-world Examples of SSRF's potent capabilities:

- SSRF to pivot internal network
- SSRF to LFI
- SSRF to query google internal server
- SSRF tips from BugBountyHQ of Images
- SSRF to RCE
- XXE at Twitter
- Blog post: Cracking the Lens: Targeting HTTP's Hidden Attack-Surface

Deserialization

- Java Deserialization in manager.paypal.com by Michael Stepankin
- Instagram's Million Dollar Bug by Wesley Wineberg
- (Ruby Cookie Deserialization RCE on facebooksearch.algolia.com by Michiel Prins (michiel)
- Java deserialization by meals

Race Condition:

- Race conditions on Facebook, DigitalOcean and others (fixed) by Josip Franjković
- Race Conditions in Popular reports feature in HackerOne by Fábio Pires (shmoo)

Business Logic Flaws:

- How I Could Steal Money from Instagram, Google and Microsoft by Arne Swinnen
- How I could have removed all your Facebook notes

- Uber Ride for Free by anand praka
- Uber Eat for Free by

Authentication Bypass:

- OneLogin authentication bypass on WordPress sites via XMLRPC in Uber by Jouko Pynnönen (jouko)
- 2FA PayPal Bypass by henryhoggard
- SAML Bug in Github worth 15000
- Authentication bypass on Airbnb via OAuth tokens theft
- Uber Login CSRF + Open Redirect -> Account Takeover at Uber
- [[http://c0rni3sm.blogspot.hk/2017/08/accidentally-typo-to-bypass.html?m=1](Administrative](http://c0rni3sm.blogspot.hk/2017/08/accidentally-typo-to-bypass.html?m=1](Administrative) Panel Access) by c0rni3sm
- Uber Bug Bounty: Gaining Access To An Internal Chat System by mishre
- User Account Takeover via Signup by Muhammad Khizer Javed

HTTP Header Injection:

- Adblock Plus and (a little) more in Google
- $10k host header by Ezequiel Pereira

Email Related:

- Slack Yammer Takeover by using TicketTrick by Inti De Ceukelaire
- How I could have mass uploaded from every Flickr account!

Information Disclosure

- Hacking SMS API Service Provider of a Company |Android App Static Security Analysis By Muhammad Khizer Javed
- Vine User Private information disclosure
- The feature works as intended, but what's in the source? By zseano

Some other real world examples:

- Payment Flaw in Yahoo

- Bypassing Google Email Domain Check to Deliver Spam Email on Google's Behalf
- When Server Side Request Forgery combine with Cross Site Scripting
- A list of FB writeup collected by phwd by phwd
- NoSQL Injection by websecurify
- CORS in action
- CORS in Fb messenger
- Web App Methodologies
- The road to hell is paved with SAML Assertions, Microsoft Vulnerability
- Study this if you like to learn Mongo SQL Injection by cirw
- Mongo DB Injection again by websecrify
- w3af speech about modern vulnerability by w3af
- Web cache attack that lead to account takeover
- A talk to teach you how to use SAML Raider
- XSS Checklist when you have no idea how to exploit the bug
- CTF write up, Great for Bug Bounty
- It turns out every site uses jquery mobile with Open Redirect is vulnerable to XSS by sirdarckcat
- Bypass CSP by using google-analytics
- Payment Issue with Paypal
- Browser Exploitation in Chinese
- XSS bypass filter
- Markup Impropose Sanitization
- Breaking XSS mitigations via Script Gadget
- X41 Browser Security White Paper
- Improper Input Validation | Add Custom Text and URLs In SMS send by Snapchat By Muhammad Khizer Javed
- Exploiting Insecure Firebase Database! By Muhammad Khizer Javed

- Using Inspect Element to Bypass Security restrictions By Muhammad Khizer Javed

Cloud Security Resources:

As bug bounty hunting evolves, so does the landscape of potential vulnerabilities. With the rapid adoption of cloud technologies, understanding cloud security is becoming increasingly important for bug bounty hunters. Cloud platforms introduce unique attack surfaces and potential weaknesses that skilled hunters can exploit. Here are some valuable resources to help you navigate the world of cloud security and enhance your bug bounty capabilities:

Cloud Fundamentals and Introduction

- Introduction to Cloud Security
- SANS – Penetration Testing in the Cloud

AWS Penetration Testing

- AWS Official Guide for Permissions and Penetration Testing
- Penetration Testing AWS Services
- Rhinosecuritylabs AWS Resources – A comprehensive collection covering S3 bucket vulnerabilities, IAM, AWS privilege escalation, and more.
- Cloud Shadow Admin Threat & Permissions – CyberArk

Azure Penetration Testing

- An Introduction to Pen Testing Azure

Write-ups

- Abusing AWS Metadata Service using SSRF Vulnerabilities
- HackerOne Report: SSRF Exploitation on AWS

List Of Some Common Vulnerabilities:

These are some common issues you should understand and learn more about. Here's a list of attack topics you should explore by reading blogs and reports:

- SQL Injection Attack
- Hibernate Query Language Injection

- Direct OS Code Injection
- XML Entity Injection
- Broken Authentication and Session Management
- Cross-Site Scripting (XSS)
- Insecure Direct Object References
- Missing Function Level Access Control
- Cross-Site Request Forgery (CSRF)
- Using Components with Known Vulnerabilities
- Unvalidated Redirects and Forwards
- ClickJacking Attacks
- DNS Cache Poisoning
- Symlinking
- Remote Code Execution Attacks
- Remote File inclusion
- Local file inclusion
- Denial oF Service Attack
- PHPwn
- NAT Pinning
- XSHM
- HTTP Parameter Pollution
- Tabnabbing
- LDAP injection
- Log Injection
- Path Traversal
- Reflected DOM Injection
- Repudiation Attack

- Resource Injection
- Server-Side Includes (SSI) Injection
- Session fixation
- Session hijacking attack
- Session Prediction
- Setting Manipulation
- Special Element Injection
- SMTP injection
- Traffic flood
- XPATH Injection

For more detailed information and examples, you can explore additional write-ups at Pentester.land Writeups & Awesome Bug Bounty. These will help you gain a better understanding of these concepts and how they can be exploited.

Exploring Mobile Application Security: Building Your Foundation

In this phase, we're delving into the exciting world of exploring Mobile Application Security.

Here's a Great **"Android Application Penetration Testing Checklist"** that you should definitely check out.

Recommended Books and Guides: Building Your Expertise

To fortify your understanding of Mobile Application Penetration Testing and Security, delve into the following essential resources:

- OWASP Mobile Application Security
- Mobile Application Penetration Testing By Vijay Kumar Velu
- Mobile Application Penetration Testing Learn Mobile Application Hacking for iOS and Android Devices
- Mobile Application Penetration Testing Professional Learning Paths INE
- Android App Reverse Engineering 101

- Mobile Systems and Smartphone Security

These resources offer comprehensive insights into the intricacies of mobile application penetration testing and security assessment.

Embrace OWASP:

Make it a priority to familiarize yourself with the OWASP Testing Guide and OWASP Top 10 Vulnerabilities. These invaluable references provide guidance and understanding:

- OWASP Mobile Application Security Testing Guide (MASTG)
- OWASP Mobile Top 10

These resources provide a solid foundation for comprehending common vulnerabilities and security practices.

Exploring Common Mobile Application Vulnerabilities

This is a crucial phase of your bug bounty journey, where we learn about common mobile application vulnerabilities that you're likely to encounter while hunting for bugs. In this section, my focus is on providing you with valuable resources to understand and learn about these vulnerabilities effectively.

Hardcoded Credentials:

Developers sometimes embed sensitive credentials in the app's code, risking the exposure of private API keys and secrets.

Real-Life Scenarios: Proof of Concepts

- Disclosure of all uploads to Cloudinary via hardcoded api secret in Android app
- Hard-Coded credentials in the Android app

WebView Vulnerabilities:

Security risks associated with improper configuration or usage of WebView, enabling attackers to execute malicious code within the app.

Real-Life Scenarios: Proof of Concepts

- Android security checklist: WebView
- Vulnerabilities in exported activity WebView
- com.basecamp.bc3 Webview Javascript Injection and JS bridge takeover

Insecure Deeplinks

Real-Life Scenarios: Proof of Concepts

- Account takeover intercepting magic link for Arrive app
- [Grab Android/iOS] Insecure deeplink leads to sensitive information disclosure
- [Zomato Order] Insecure deeplink leads to sensitive information disclosure

Remote Code Execution (RCE) / Arbitrary Code Execution (ACE)

Insecure loading of dynamic code allows attackers to execute arbitrary commands, potentially leading to unauthorized access or control of the app.

- RCE in TinyCards for Android Report on HackerOne
- Why dynamic code loading could be dangerous for your apps: a Google example. Reference: Oversecured Blog
- Persistent Arbitrary Code Execution HackerOne Report, CVE-2020-8913
- TikTok Vulnerabilities Reference

Memory Corruption:

Exploiting memory vulnerabilities to manipulate app behavior or inject malicious code, potentially compromising user data.

- Exploiting memory corruption vulnerabilities on Android.

Cryptography in Mobile Apps:

Mistakes in implementing cryptographic techniques may expose sensitive data, jeopardizing user privacy.

- Use cryptography in mobile apps the right way.

SQL Injection:

Lack of input validation in SQL queries can lead to injection attacks, enabling attackers to manipulate the app's database.

- SQL Injection found in NextCloud Android App Content Provider

Session Theft:

Attacks that target user sessions, potentially allowing unauthorized access to user accounts.

- [Zomato Android/iOS] Theft of user session.

File Theft and Manipulation:

Weaknesses in handling files may enable attackers to steal or manipulate sensitive user data.

- Android security checklist: Theft of arbitrary files.

Insecure WebResourceResponse Configurations:

Misconfigurations in WebResourceResponse may expose apps to attacks that manipulate responses and compromise user security.

- Android: Exploring vulnerabilities in WebResourceResponse

Vulnerable to Local File Steal, JavaScript Injection, Open Redirect:

Apps may be vulnerable to a combination of attacks including local file theft, JavaScript injection, and open redirects.

- Twitter lite(Android): Vulnerable to local file steal, Javascript injection, Open redirect

Token Leakage Due to Stolen Files:

Stolen tokens from insecure storage may lead to unauthorized access to user accounts.

- [IRCCloud Android] Theft of arbitrary files leading to token leakage Share

Bypasses:

Methods that allow attackers to bypass security mechanisms, potentially gaining unauthorized access to the app.

- Accidental $70k Google Pixel Lock Screen Bypass. Reference
- Golden Techniques to Bypass Host Validations
- Two-factor authentication bypass on Grab Android App
- Extremly simple way to bypass Nextcloud-Client PIN/Fingerprint lock

Cross-Site Scripting (XSS):

Injection of malicious scripts into web content, leading to unauthorized actions or data theft.

- [Android] HTML Injection in BatterySaveArticleRenderer WebView

Privilege Escalation:

Discovering vulnerabilities that allow attackers to elevate their privileges, potentially gaining unauthorized access to sensitive app functionalities.

- Discovering vendor-specific vulnerabilities in Android.

Intent Spoofing:

Manipulating app intents to perform unauthorized actions or access restricted components.

- Reference: Report on HackerOne

Access of Not Exported Content Providers:

Gaining unauthorized access to content providers that are not properly exported, potentially exposing sensitive data.

- Multiple critical vulnerabilities in Odnoklassniki Android application

Access Protected Components via Intent:

Exploiting intents to access protected app components without proper authorization.

- Access of Android protected components via embedded intent
- Fragment Injection

Javascript Injection:

Injection of malicious JavaScript code into app components, enabling attackers to manipulate app behavior.

- Vulnerable to JavaScript injection. (WXS) (Javascript injection)!

Cross-Site Request Forgery (CSRF):

Tricking users into performing unintended actions, potentially compromising their accounts or data.

- Periscope android app deeplink leads to CSRF in follow action.

Case Sensitive Account Collisions:

Exploiting case sensitivity in account identifiers to perform unauthorized actions or account takeovers.

- Vine – overwrite account associated with email via android application.

Intercept Broadcasts:

Intercepting broadcasts to gain unauthorized access to sensitive information or execute actions.

- Possible to intercept broadcasts about file uploads.
- Vulnerable Exported Broadcast Receiver
- View Every Network Request Response's Information

Stay updated with HackerOne Public Bug reports by regularly following HackerOne Public Reports, where you can learn a lot from real-world bug reports.

Blogs & YouTube Channels Worth Following!

Blogs and YouTube channels created by seasoned hackers and security enthusiasts serve as invaluable resources for those seeking to delve deeper into the world of vulnerabilities, exploits, and defensive techniques. By following these trusted sources, you gain access to real-world examples, detailed breakdowns of attack vectors, and practical demonstrations. In this section, we've curated a list of recommended blogs and YouTube channels that provide a wealth of knowledge, enabling you to enhance your skill set and stay ahead in the dynamic field of bug bounty hunting.

Blogs:

- IT Security Guard
- Brute Logic
- Klikki
- Philippe Harewood
- Sean Melia
- Respect XSS
- Graceful Security
- Jack Whitton
- Tisiphone
- Nahamsec

- Bitquark
- Arne Swinnen
- Bug Bounty POC
- Arbaz Hussain
- Shawar Khan
- Detectify Blog
- Security Wall
- HackerOne Blog
- SecurityTube
- Hack Asia
- Mukarram Khalid
- Jubaer Alnazi White Hat
- Hackaday
- Packet Storm Security
- Black Hat
- Metasploit
- SecTools
- Detectify Labs
- Security Idiots
- HackerNoon
- SQLi Basic
- Vulnerability Lab
- KnowOnix
- Coding Karma
- remonsec

YouTube Channels:

- security idiots
- Black Hat
- Hisham Mir
- Muhammad Khizer Javed
- Frans Rosén
- HackerOne
- josue Fernandez
- Bugcrowd
- intigriti
- Web Development Tutorials
- Jan Wikholm
- Penetration Testing in Linux
- Farah Hawa
- LiveOverflow
- The Cyber Mentor
- David Bombal
- Bug Bounty Reports Explained
- PhD Security
- NahamSec
- NetworkChuck
- STÖK
- remonsec

Groups to Join!

You can also join Slack & Discord communities for hackers to connect, share insights, and learn from fellow bug bounty hunters:

1. BugBounty World

2. BugBounty Forum
3. SecurityNewbs
4. BugCrowd Discord
5. Hacker101 Discord

These resources, blogs, and YouTube channels are excellent ways to expand your knowledge and stay informed about the latest trends, techniques, and experiences in the world of bug bounty hunting and cybersecurity.

PART 3

Bug Bounty Tools & Scripts: Your Arsenal for Successful Hunting

Bug Bounty Hunting is a career that is known for the heavy use of security tools. These tools help us find vulnerabilities in software, web, and mobile applications and are an integral part of bounty hunting. Below is a list of security tools for bug bounty hunters.

Tools you should definitely know about:

- **BurpSuite: Burp Suite is a software security application used for penetration testing of web applications.**
- **ZAP: OWASP ZAP is an open-source web application security scanner.**
- **Caido: A lightweight web security auditing toolkit.**

Below is an awesome list by Kamil Vavra. I would love it if you could go and give this repository a star.

Recon

Subdomain Enumeration

- **Sublist3r – Fast subdomains enumeration tool for penetration testers**
- **Amass – In-depth Attack Surface Mapping and Asset Discovery**
- **massdns – A high-performance DNS stub resolver for bulk lookups and reconnaissance (subdomain enumeration)**
- **Findomain – The fastest and cross-platform subdomain enumerator, do not waste your time.**

- **Sudomy – Sudomy is a subdomain enumeration tool to collect subdomains and analyzing domains performing automated reconnaissance (recon) for bug hunting / pentesting**
- **chaos-client – Go client to communicate with Chaos DNS API.**
- **domained – Multi Tool Subdomain Enumeration**
- **bugcrowd-levelup-subdomain-enumeration – This repository contains all the material from the talk "Esoteric sub-domain enumeration techniques" given at Bugcrowd LevelUp 2017 virtual conference**
- **shuffledns – shuffleDNS is a wrapper around massdns written in go that allows you to enumerate valid subdomains using active bruteforce as well as resolve subdomains with wildcard handling and easy input-output...**
- **censys-subdomain-finder – Perform subdomain enumeration using the certificate transparency logs from Censys.**
- **Turbolist3r – Subdomain enumeration tool with analysis features for discovered domains**
- **censys-enumeration – A script to extract subdomains/emails for a given domain using SSL/TLS certificate dataset on Censys**
- **tugarecon – Fast subdomains enumeration tool for penetration testers.**
- **as3nt – Another Subdomain ENumeration Tool**
- **Subra – A Web-UI for subdomain enumeration (subfinder)**
- **Substr3am – Passive reconnaissance/enumeration of interesting targets by watching for SSL certificates being issued**
- **domain – enumall.py Setup script for Regon-ng**
- **altdns – Generates permutations, alterations and mutations of subdomains and then resolves them**
- **brutesubs – An automation framework for running multiple open sourced subdomain bruteforcing tools (in parallel) using your own wordlists via Docker Compose**
- **dns-parallel-prober – his is a parallelised domain name prober to find as many subdomains of a given domain as fast as possible.**

- dnscan – dnscan is a python wordlist-based DNS subdomain scanner.
- knock – Knockpy is a python tool designed to enumerate subdomains on a target domain through a wordlist.
- hakrevdns – Small, fast tool for performing reverse DNS lookups en masse.
- dnsx – Dnsx is a fast and multi-purpose DNS toolkit allow to run multiple DNS queries of your choice with a list of user-supplied resolvers.
- subfinder – Subfinder is a subdomain discovery tool that discovers valid subdomains for websites.
- assetfinder – Find domains and subdomains related to a given domain
- crtndstry – Yet another subdomain finder
- VHostScan – A virtual host scanner that performs reverse lookups
- scilla – Information Gathering tool – DNS / Subdomains / Ports / Directories enumeration
- sub3suite – A research-grade suite of tools for subdomain enumeration, intelligence gathering and attack surface mapping.
- cero – Scrape domain names from SSL certificates of arbitrary hosts

Port Scanning

- masscan – TCP port scanner, spews SYN packets asynchronously, scanning entire Internet in under 5 minutes.
- RustScan – The Modern Port Scanner
- naabu – A fast port scanner written in go with focus on reliability and simplicity.
- nmap – Nmap – the Network Mapper. Github mirror of official SVN repository.
- sandmap – Nmap on steroids. Simple CLI with the ability to run pure Nmap engine, 31 modules with 459 scan profiles.
- ScanCannon – Combines the speed of masscan with the reliability and detailed enumeration of nmap

Screenshots

- EyeWitness – EyeWitness is designed to take screenshots of websites, provide some server header info, and identify default credentials if possible.

- aquatone – Aquatone is a tool for visual inspection of websites across a large amount of hosts and is convenient for quickly gaining an overview of HTTP-based attack surface.
- screenshoteer – Make website screenshots and mobile emulations from the command line.
- gowitness – gowitness – a golang, web screenshot utility using Chrome Headless
- WitnessMe – Web Inventory tool, takes screenshots of webpages using Pyppeteer (headless Chrome/Chromium) and provides some extra bells & whistles to make life easier.
- eyeballer – Convolutional neural network for analyzing pentest screenshots
- scrying – A tool for collecting RDP, web and VNC screenshots all in one place
- Depix – Recovers passwords from pixelized screenshots
- httpscreenshot – HTTPScreenshot is a tool for grabbing screenshots and HTML of large numbers of websites.

Technologies

- wappalyzer – Identify technology on websites.
- webanalyze – Port of Wappalyzer (uncovers technologies used on websites) to automate mass scanning.
- python-builtwith – BuiltWith API client
- whatweb – Next generation web scanner
- retire.js – scanner detecting the use of JavaScript libraries with known vulnerabilities
- httpx – httpx is a fast and multi-purpose HTTP toolkit allows to run multiple probers using retryablehttp library, it is designed to maintain the result reliability with increased threads.
- fingerprintx – fingerprintx is a standalone utility for service discovery on open ports that works well with other popular bug bounty command line tools.

Content Discovery

- gobuster – Directory/File, DNS and VHost busting tool written in Go

- recursebuster – rapid content discovery tool for recursively querying webservers, handy in pentesting and web application assessments
- feroxbuster – A fast, simple, recursive content discovery tool written in Rust.
- dirsearch – Web path scanner
- dirsearch – A Go implementation of dirsearch.
- filebuster – An extremely fast and flexible web fuzzer
- dirstalk – Modern alternative to dirbuster/dirb
- dirbuster-ng – dirbuster-ng is C CLI implementation of the Java dirbuster tool
- gospider – Gospider – Fast web spider written in Go
- hakrawler – Simple, fast web crawler designed for easy, quick discovery of endpoints and assets within a web application
- crawley – fast, feature-rich unix-way web scraper/crawler written in Golang.

Links

- LinkFinder – A python script that finds endpoints in JavaScript files
- JS-Scan – a .js scanner, built in php. designed to scrape urls and other info
- LinksDumper – Extract (links/possible endpoints) from responses & filter them via decoding/sorting
- GoLinkFinder – A fast and minimal JS endpoint extractor
- BurpJSLinkFinder – Burp Extension for a passive scanning JS files for endpoint links.
- urlgrab – A golang utility to spider through a website searching for additional links.
- waybackurls – Fetch all the URLs that the Wayback Machine knows about for a domain
- gau – Fetch known URLs from AlienVault's Open Threat Exchange, the Wayback Machine, and Common Crawl.
- getJS – A tool to fastly get all javascript sources/files
- linx – Reveals invisible links within JavaScript files

Parameters

- **parameth – This tool can be used to brute discover GET and POST parameters**
- **param-miner – This extension identifies hidden, unlinked parameters. It's particularly useful for finding web cache poisoning vulnerabilities.**
- **ParamPamPam – This tool for brute discover GET and POST parameters.**
- **Arjun – HTTP parameter discovery suite.**
- **ParamSpider – Mining parameters from dark corners of Web Archives.**
- **x8 – Hidden parameters discovery suite written in Rust.**

Fuzzing

- **wfuzz – Web application fuzzer**
- **ffuf – Fast web fuzzer written in Go**
- **fuzzdb – Dictionary of attack patterns and primitives for black-box application fault injection and resource discovery.**
- **IntruderPayloads – A collection of Burpsuite Intruder payloads, BurpBounty payloads, fuzz lists, malicious file uploads and web pentesting methodologies and checklists.**
- **fuzz.txt – Potentially dangerous files**
- **fuzzilli – A JavaScript Engine Fuzzer**
- **fuzzapi – Fuzzapi is a tool used for REST API pentesting and uses API_Fuzzer gem**
- **qsfuzz – qsfuzz (Query String Fuzz) allows you to build your own rules to fuzz query strings and easily identify vulnerabilities.**
- **vaf – very advanced (web) fuzzer written in Nim.**

Cloud Security Tools

- **SkyArk – Privilege Escalation and Data Collection for AWS**
- **Pacu – AWS Exploitation Framework**
- **AWS Privilege Escalation Testing Script**
- **AWS Exploitation Framework – RhinoSecurityLabs**

Exploitation

List of tools that will be helpful during exploitation.

Command Injection

- commix – Automated All-in-One OS command injection and exploitation tool.

CORS Misconfiguration

- Corsy – CORS Misconfiguration Scanner
- CORStest – A simple CORS misconfiguration scanner
- cors-scanner – A multi-threaded scanner that helps identify CORS flaws/misconfigurations
- CorsMe – Cross Origin Resource Sharing MisConfiguration Scanner

CRLF Injection

- CRLFsuite – A fast tool specially designed to scan CRLF injection
- crlfuzz – A fast tool to scan CRLF vulnerability written in Go
- CRLF-Injection-Scanner – Command line tool for testing CRLF injection on a list of domains.
- Injectus – CRLF and open redirect fuzzer

CSRF Injection

- XSRFProbe -The Prime Cross Site Request Forgery (CSRF) Audit and Exploitation Toolkit.

Directory Traversal

- dotdotpwn – DotDotPwn – The Directory Traversal Fuzzer
- FDsploit – File Inclusion & Directory Traversal fuzzing, enumeration & exploitation tool.
- off-by-slash – Burp extension to detect alias traversal via NGINX misconfiguration at scale.
- liffier – tired of manually add dot-dot-slash to your possible path traversal? this short snippet will increment ../ on the URL.

File Inclusion

- liffy – Local file inclusion exploitation tool
- Burp-LFI-tests – Fuzzing for LFI using Burpsuite
- LFI-Enum – Scripts to execute enumeration via LFI
- LFISuite – Totally Automatic LFI Exploiter (+ Reverse Shell) and Scanner
- LFI-files – Wordlist to bruteforce for LFI

GraphQL Injection

- inql – InQL – A Burp Extension for GraphQL Security Testing
- GraphQLmap – GraphQLmap is a scripting engine to interact with a graphql endpoint for pentesting purposes.
- shapeshifter – GraphQL security testing tool
- graphql_beautifier – Burp Suite extension to help make Graphql request more readable
- clairvoyance – Obtain GraphQL API schema despite disabled introspection!

Header Injection

- headi – Customisable and automated HTTP header injection.

Insecure Deserialization

- ysoserial – A proof-of-concept tool for generating payloads that exploit unsafe Java object deserialization.
- GadgetProbe – Probe endpoints consuming Java serialized objects to identify classes, libraries, and library versions on remote Java classpaths.
- ysoserial.net – Deserialization payload generator for a variety of .NET formatters
- phpggc – PHPGGC is a library of PHP unserialize() payloads along with a tool to generate them, from command line or programmatically.

Insecure Direct Object References

- Autorize – Automatic authorization enforcement detection extension for burp suite written in Jython developed by Barak Tawily

Open Redirect

- Oralyzer – Open Redirection Analyzer
- Injectus – CRLF and open redirect fuzzer
- dom-red – Small script to check a list of domains against open redirect vulnerability
- OpenRedireX – A Fuzzer for OpenRedirect issues

Race Condition

- razzer – A Kernel fuzzer focusing on race bugs
- racepwn – Race Condition framework
- requests-racer – Small Python library that makes it easy to exploit race conditions in web apps with Requests.
- turbo-intruder – Turbo Intruder is a Burp Suite extension for sending large numbers of HTTP requests and analyzing the results.
- race-the-web – Tests for race conditions in web applications. Includes a RESTful API to integrate into a continuous integration pipeline.

Request Smuggling

- http-request-smuggling – HTTP Request Smuggling Detection Tool
- smuggler – Smuggler – An HTTP Request Smuggling / Desync testing tool written in Python 3
- h2csmuggler – HTTP Request Smuggling over HTTP/2 Cleartext (h2c)
- tiscripts – These scripts I use to create Request Smuggling Desync payloads for CLTE and TECL style attacks.

Server Side Request Forgery

- SSRFmap – Automatic SSRF fuzzer and exploitation tool
- Gopherus – This tool generates gopher link for exploiting SSRF and gaining RCE in various servers
- ground-control – A collection of scripts that run on my web server. Mainly for debugging SSRF, blind XSS, and XXE vulnerabilities.
- SSRFire – An automated SSRF finder. Just give the domain name and your server and chill! 😉 Also has options to find XSS and open redirects

- **httprebind – Automatic tool for DNS rebinding-based SSRF attacks**
- **ssrf-sheriff – A simple SSRF-testing sheriff written in Go**
- **B-XSSRF – Toolkit to detect and keep track on Blind XSS, XXE & SSRF**
- **extended-ssrf-search – Smart ssrf scanner using different methods like parameter brute forcing in post and get...**
- **gaussrf – Fetch known URLs from AlienVault's Open Threat Exchange, the Wayback Machine, and Common Crawl and Filter Urls With OpenRedirection or SSRF Parameters.**
- **ssrfDetector – Server-side request forgery detector**
- **grafana-ssrf – Authenticated SSRF in Grafana**
- **sentrySSRF – Tool to searching sentry config on page or in javascript files and check blind SSRF**
- **lorsrf – Bruteforcing on Hidden parameters to find SSRF vulnerability using GET and POST Methods**
- **singularity – A DNS rebinding attack framework.**
- **whonow – A "malicious" DNS server for executing DNS Rebinding attacks on the fly (public instance running on rebind.network:53)**
- **dns-rebind-toolkit – A front-end JavaScript toolkit for creating DNS rebinding attacks.**
- **dref – DNS Rebinding Exploitation Framework**
- **rbndr – Simple DNS Rebinding Service**
- **httprebind – Automatic tool for DNS rebinding-based SSRF attacks**
- **dnsFookup – DNS rebinding toolkit**

SQL Injection

- **sqlmap – Automatic SQL injection and database takeover tool**
- **NoSQLMap – Automated NoSQL database enumeration and web application exploitation tool.**
- **SQLiScanner – Automatic SQL injection with Charles and sqlmap api**

- SleuthQL – Python3 Burp History parsing tool to discover potential SQL injection points. To be used in tandem with SQLmap.
- mssqlproxy – mssqlproxy is a toolkit aimed to perform lateral movement in restricted environments through a compromised Microsoft SQL Server via socket reuse
- sqli-hunter – SQLi-Hunter is a simple HTTP / HTTPS proxy server and a SQLMAP API wrapper that makes digging SQLi easy.
- waybackSqliScanner – Gather urls from wayback machine then test each GET parameter for sql injection.
- ESC – Evil SQL Client (ESC) is an interactive .NET SQL console client with enhanced SQL Server discovery, access, and data exfiltration features.
- mssqli-duet – SQL injection script for MSSQL that extracts domain users from an Active Directory environment based on RID bruteforcing
- burp-to-sqlmap – Performing SQLInjection test on Burp Suite Bulk Requests using SQLMap
- BurpSQLTruncSanner – Messy BurpSuite plugin for SQL Truncation vulnerabilities.
- andor – Blind SQL Injection Tool with Golang
- Blinder – A python library to automate time-based blind SQL injection
- sqliv – massive SQL injection vulnerability scanner
- nosqli – NoSql Injection CLI tool, for finding vulnerable websites using MongoDB.

XSS Injection

- XSStrike – Most advanced XSS scanner.
- xssor2 – XSS'OR – Hack with JavaScript.
- xsscrapy – XSS spider – 66/66 wavsep XSS detected
- sleepy-puppy – Sleepy Puppy XSS Payload Management Framework
- ezXSS – ezXSS is an easy way for penetration testers and bug bounty hunters to test (blind) Cross Site Scripting.

- xsshunter – The XSS Hunter service – a portable version of XSSHunter.com
- dalfox – DalFox(Finder Of XSS) / Parameter Analysis and XSS Scanning tool based on golang
- xsser – Cross Site "Scripter" (aka XSSer) is an automatic -framework- to detect, exploit and report XSS vulnerabilities in web-based applications.
- XSpear – Powerfull XSS Scanning and Parameter analysis tool&gem
- weaponised-XSS-payloads – XSS payloads designed to turn alert(1) into P1
- tracy – A tool designed to assist with finding all sinks and sources of a web application and display these results in a digestible manner.
- ground-control – A collection of scripts that run on my web server. Mainly for debugging SSRF, blind XSS, and XXE vulnerabilities.
- xssValidator – This is a burp intruder extender that is designed for automation and validation of XSS vulnerabilities.
- JSShell – An interactive multi-user web JS shell
- bXSS – bXSS is a utility which can be used by bug hunters and organizations to identify Blind Cross-Site Scripting.
- docem – Uility to embed XXE and XSS payloads in docx,odt,pptx,etc (OXML_XEE on steroids)
- XSS-Radar – XSS Radar is a tool that detects parameters and fuzzes them for cross-site scripting vulnerabilities.
- BruteXSS – BruteXSS is a tool written in python simply to find XSS vulnerabilities in web application.
- findom-xss – A fast DOM based XSS vulnerability scanner with simplicity.
- domdig – DOM XSS scanner for Single Page Applications
- femida – Automated blind-xss search for Burp Suite
- B-XSSRF – Toolkit to detect and keep track on Blind XSS, XXE & SSRF
- domxssscanner – DOMXSS Scanner is an online tool to scan source code for DOM based XSS vulnerabilities
- xsshunter_client – Correlated injection proxy tool for XSS Hunter

- extended-xss-search – A better version of my xssfinder tool – scans for different types of xss on a list of urls.
- XSSCon – XSSCon: Simple XSS Scanner tool
- BitBlinder – BurpSuite extension to inject custom cross-site scripting payloads on every form/request submitted to detect blind XSS vulnerabilities
- XSSOauthPersistence – Maintaining account persistence via XSS and Oauth
- shadow-workers – Shadow Workers is a free and open source C2 and proxy designed for penetration testers to help in the exploitation of XSS and malicious Service Workers (SW)
- rexsser – This is a burp plugin that extracts keywords from response using regexes and test for reflected XSS on the target scope.
- vaya-ciego-nen – Detect, manage and exploit Blind Cross-site scripting (XSS) vulnerabilities.
- dom-based-xss-finder – Chrome extension that finds DOM based XSS vulnerabilities
- xss2png – PNG IDAT chunks XSS payload generator
- XSSwagger – A simple Swagger-ui scanner that can detect old versions vulnerable to various XSS attacks

XXE Injection

- ground-control – A collection of scripts that run on my web server. Mainly for debugging SSRF, blind XSS, and XXE vulnerabilities.
- dtd-finder – List DTDs and generate XXE payloads using those local DTDs.
- docem – Uility to embed XXE and XSS payloads in docx,odt,pptx,etc (OXML_XEE on steroids)
- xxeserv – A mini webserver with FTP support for XXE payloads
- xxexploiter – Tool to help exploit XXE vulnerabilities
- B-XSSRF – Toolkit to detect and keep track on Blind XSS, XXE & SSRF
- XXEinjector – Tool for automatic exploitation of XXE vulnerability using direct and different out of band methods.

- oxml_xxe – A tool for embedding XXE/XML exploits into different filetypes
- metahttp – A bash script that automates the scanning of a target network for HTTP resources through XXE

Miscellaneous

Passwords

- thc-hydra – Hydra is a parallelized login cracker which supports numerous protocols to attack.
- DefaultCreds-cheat-sheet – One place for all the default credentials to assist the Blue/Red teamers activities on finding devices with default password
- changeme – A default credential scanner.
- BruteX – Automatically brute force all services running on a target.
- patator – Patator is a multi-purpose brute-forcer, with a modular design and a flexible usage.

Secrets

- git-secrets – Prevents you from committing secrets and credentials into git repositories
- gitleaks – Scan git repos (or files) for secrets using regex and entropy
- truffleHog – Searches through git repositories for high entropy strings and secrets, digging deep into commit history
- gitGraber – gitGraber: monitor GitHub to search and find sensitive data in real time for different online services
- talisman – By hooking into the pre-push hook provided by Git, Talisman validates the outgoing changeset for things that look suspicious – such as authorization tokens and private keys.
- GitGot – Semi-automated, feedback-driven tool to rapidly search through troves of public data on GitHub for sensitive secrets.
- git-all-secrets – A tool to capture all the git secrets by leveraging multiple open source git searching tools

- github-search – Tools to perform basic search on GitHub.
- git-vuln-finder – Finding potential software vulnerabilities from git commit messages
- commit-stream – #OSINT tool for finding Github repositories by extracting commit logs in real time from the Github event API
- gitrob – Reconnaissance tool for GitHub organizations
- repo-supervisor – Scan your code for security misconfiguration, search for passwords and secrets.
- GitMiner – Tool for advanced mining for content on Github
- shhgit – Ah shhgit! Find GitHub secrets in real time
- detect-secrets – An enterprise friendly way of detecting and preventing secrets in code.
- rusty-hog – A suite of secret scanners built in Rust for performance. Based on TruffleHog
- whispers – Identify hardcoded secrets and dangerous behaviours
- yar – Yar is a tool for plunderin' organizations, users and/or repositories.
- dufflebag – Search exposed EBS volumes for secrets
- secret-bridge – Monitors Github for leaked secrets
- earlybird – EarlyBird is a sensitive data detection tool capable of scanning source code repositories for clear text password violations, PII, outdated cryptography methods, key files and more.
- Trufflehog-Chrome-Extension – Trufflehog-Chrome-Extension
- noseyparker – Nosey Parker is a command-line program that finds secrets and sensitive information in textual data and Git history.

Git

- GitTools – A repository with 3 tools for pwn'ing websites with .git repositories available
- gitjacker – Leak git repositories from misconfigured websites
- git-dumper – A tool to dump a git repository from a website

- GitHunter – A tool for searching a Git repository for interesting content
- dvcs-ripper – Rip web accessible (distributed) version control systems: SVN/GIT/HG...
- Gato (Github Attack TOolkit) – GitHub Self-Hosted Runner Enumeration and Attack Tool

Buckets

- S3Scanner – Scan for open AWS S3 buckets and dump the contents
- AWSBucketDump – Security Tool to Look For Interesting Files in S3 Buckets
- CloudScraper – CloudScraper: Tool to enumerate targets in search of cloud resources. S3 Buckets, Azure Blobs, Digital Ocean Storage Space.
- s3viewer – Publicly Open Amazon AWS S3 Bucket Viewer
- festin – FestIn – S3 Bucket Weakness Discovery
- s3reverse – The format of various s3 buckets is convert in one format. for bugbounty and security testing.
- mass-s3-bucket-tester – This tests a list of s3 buckets to see if they have dir listings enabled or if they are uploadable
- S3BucketList – Firefox plugin that lists Amazon S3 Buckets found in requests
- dirlstr – Finds Directory Listings or open S3 buckets from a list of URLs
- Burp-AnonymousCloud – Burp extension that performs a passive scan to identify cloud buckets and then test them for publicly accessible vulnerabilities
- kicks3 – S3 bucket finder from html,js and bucket misconfiguration testing tool
- 2tearsinabucket – Enumerate s3 buckets for a specific target.
- s3_objects_check – Whitebox evaluation of effective S3 object permissions, to identify publicly accessible files.
- s3tk – A security toolkit for Amazon S3
- CloudBrute – Awesome cloud enumerator
- s3cario – This tool will get the CNAME first if it's a valid Amazon s3 bucket and if it's not, it will try to check if the domain is a bucket name.

- S3Cruze – All-in-one AWS S3 bucket tool for pentesters.

CMS

- wpscan – WPScan is a free, for non-commercial use, black box WordPress security scanner
- WPSpider – A centralized dashboard for running and scheduling WordPress scans powered by wpscan utility.
- wprecon – WordPress Recon
- CMSmap – CMSmap is a python open source CMS scanner that automates the process of detecting security flaws of the most popular CMSs.
- joomscan – OWASP Joomla Vulnerability Scanner Project
- pyfiscan – Free web-application vulnerability and version scanner

JSON Web Token

- jwt_tool – A toolkit for testing, tweaking and cracking JSON Web Tokens
- c-jwt-cracker – JWT brute force cracker written in C
- jwt-heartbreaker – The Burp extension to check JWT (JSON Web Tokens) for using keys from known from public sources
- jwtear – Modular command-line tool to parse, create and manipulate JWT tokens for hackers
- jwt-key-id-injector – Simple python script to check against hypothetical JWT vulnerability.
- jwt-hack – jwt-hack is tool for hacking / security testing to JWT.
- jwt-cracker – Simple HS256 JWT token brute force cracker

postMessage

- postMessage-tracker – A Chrome Extension to track postMessage usage (url, domain and stack) both by logging using CORS and also visually as an extension-icon
- PostMessage_Fuzz_Tool – #BugBounty #BugBounty Tools #WebDeveloper Tool

Subdomain Takeover

- subjack – Subdomain Takeover tool written in Go
- SubOver – A Powerful Subdomain Takeover Tool
- autoSubTakeover – A tool used to check if a CNAME resolves to the scope address. If the CNAME resolves to a non-scope address it might be worth checking out if subdomain takeover is possible.
- NSBrute – Python utility to takeover domains vulnerable to AWS NS Takeover
- can-i-take-over-xyz – "Can I take over XYZ?" — a list of services and how to claim (sub)domains with dangling DNS records.
- cnames – take a list of resolved subdomains and output any corresponding CNAMES en masse.
- subHijack – Hijacking forgotten & misconfigured subdomains
- tko-subs – A tool that can help detect and takeover subdomains with dead DNS records
- HostileSubBruteforcer – This app will bruteforce for exisiting subdomains and provide information if the 3rd party host has been properly setup.
- second-order – Second-order subdomain takeover scanner
- takeover – A tool for testing subdomain takeover possibilities at a mass scale.
- dnsReaper – DNS Reaper is yet another sub-domain takeover tool, but with an emphasis on accuracy, speed and the number of signatures in our arsenal!

Vulnerability Scanners

- nuclei – Nuclei is a fast tool for configurable targeted scanning based on templates offering massive extensibility and ease of use.
- Sn1per – Automated pentest framework for offensive security experts
- metasploit-framework – Metasploit Framework
- nikto – Nikto web server scanner
- arachni – Web Application Security Scanner Framework
- jaeles – The Swiss Army knife for automated Web Application Testing
- retire.js – scanner detecting the use of JavaScript libraries with known vulnerabilities

- Osmedeus – Fully automated offensive security framework for reconnaissance and vulnerability scanning
- getsploit – Command line utility for searching and downloading exploits
- flan – A pretty sweet vulnerability scanner
- Findsploit – Find exploits in local and online databases instantly
- BlackWidow – A Python based web application scanner to gather OSINT and fuzz for OWASP vulnerabilities on a target website.
- backslash-powered-scanner – Finds unknown classes of injection vulnerabilities
- Eagle – Multithreaded Plugin based vulnerability scanner for mass detection of web-based applications vulnerabilities
- cariddi – Take a list of domains, crawl urls and scan for endpoints, secrets, api keys, file extensions, tokens and more...
- OWASP ZAP – World's most popular free web security tools and is actively maintained by a dedicated international team of volunteers
- SSTImap – SSTImap is a penetration testing software that can check websites for Code Injection and Server-Side Template Injection vulnerabilities and exploit them, giving access to the operating system itself.

Uncategorized

- JSONBee – A ready to use JSONP endpoints/payloads to help bypass content security policy (CSP) of different websites.
- CyberChef – The Cyber Swiss Army Knife – a web app for encryption, encoding, compression and data analysis
- –
- bountyplz – Automated security reporting from markdown templates (HackerOne and Bugcrowd are currently the platforms supported)
- PayloadsAllTheThings – A list of useful payloads and bypass for Web Application Security and Pentest/CTF

- bounty-targets-data – This repo contains hourly-updated data dumps of bug bounty platform scopes (like Hackerone/Bugcrowd/Intigriti/etc) that are eligible for reports
- android-security-awesome – A collection of android security related resources
- awesome-mobile-security – An effort to build a single place for all useful android and iOS security related stuff.
- awesome-vulnerable-apps – Awesome Vulnerable Applications
- XFFenum – X-Forwarded-For [403 forbidden] enumeration
- httpx – httpx is a fast and multi-purpose HTTP toolkit allow to run multiple probers using retryablehttp library, it is designed to maintain the result reliability with increased threads.
- csprecon – Discover new target domains using Content Security Policy

---

Continual Learning and Practice

Bug bounty hunting requires continual learning and practice. As you progress, you'll find each bug bounty program has its unique challenges and rewards. Learn from your experiences and always strive to improve your skills.

As you start your journey to become a bug bounty hunter, you'll find that practicing and honing your skills is a crucial step. Capture The Flag (CTF) challenges provide an excellent platform to exercise your abilities by simulating real-world vulnerabilities. Engaging in these challenges exposes you to diverse technologies required to breach applications and systems effectively.

Learning and Practicing Resources:

To aid your Bug Bounty Hunting journey, here's a curated list of reputable CTF platforms and learning resources:

- PentesterLab: PentesterLab is an excellent resource for learning about web application security and ways how it can be subverted.
- Hacker101: This platform offers a collection of web security challenges with a focus on practical skills. It covers a wide range of topics, making it suitable for both beginners and seasoned professionals. Hacker 101

- **Hack The Box: With a vibrant community, Hack The Box provides a diverse set of realistic challenges that encompass various skill levels. It's a great platform to enhance your penetration testing skills. Hack the Box**
- **OverTheWire Wargames: This platform offers a series of war games designed to teach and test various security concepts. It covers networking, cryptography, and more. OverTheWire Wargames**
- **Pwnable.tw: If you're interested in binary exploitation and reverse engineering, Pwnable.tw offers challenges that require you to analyze and exploit vulnerable binaries. Pwnable.tw**
- **VulnHub: VulnHub provides a collection of vulnerable virtual machines that allow you to practice exploiting real-world scenarios in controlled environments. VulnHub**
- **"Hack Yourself First" by Troy Hunt: This resource offers practical lessons to help you understand how common security vulnerabilities can be exploited and how to prevent them. Hack Yourself First**
- **Hacksplaining: Hacksplaining offers interactive lessons that break down complex security topics, providing clear explanations and practical demonstrations. Hacksplaining**
- **Penetration Testing Practice Labs: Aman Hardikar's collection of practice labs covers various security concepts and challenges, enabling you to test your skills. Practice Labs**
- **Bug Bounty Hunter: This platform provides a set of challenges that mimic real-world bug bounty scenarios, helping you refine your skills for actual bug hunting. Bug Bounty Hunter**
- **PortSwigger Web Security: PortSwigger offers comprehensive web security training, including hands-on labs and exercises to enhance your web application security skills. PortSwigger Web Security**
- **TryHackMe: TryHackMe offers a variety of virtual rooms and challenges to help you learn and practice penetration testing techniques. TryHackMe**
- **CTFTime: CTFTime is a platform that provides information about upcoming CTF events, allowing you to participate and challenge yourself against the best. CTFTime**

- **Gin and Juice Shop: This is a deliberately vulnerable web application that helps you practice your security testing skills in a realistic setting. Gin and Juice Shop**
- **OWASP Juice Shop: OWASP Juice Shop is another vulnerable web application designed to educate and train security professionals on web security. OWASP Juice Shop**

Cloud CTFs:

- **AWS CTF Challenges – Flaws.Cloud**
- **Azure CTF Challenges – brokenazure.cloud**
- **Google Cloud CTF Challanges – thunder-ctf.cloud**
- **Kubernetes Goat – Kubernetes Goat is "Vulnerable by Design" Kubernetes Cluster. Designed to be an intentionally vulnerable cluster environment to learn and practice Kubernetes security.**
- **CloudGoat – CloudGoat is Rhino Security Labs' "Vulnerable by Design" AWS deployment tool**
- **CdkGoat – Vulnerable AWS CDK Infra – CdkGoat is Bridgecrew's "Vulnerable by Design" AWS CDK repository.**
- **Cfngoat – Vulnerable Cloudformation Template – Cfngoat is Bridgecrew's "Vulnerable by Design" Cloudformation repository.**
- **TerraGoat – Vulnerable Terraform Infra – TerraGoat is Bridgecrew's "Vulnerable by Design" Terraform repository.**
- **caponeme – Capital One Breach – Repository demonstrating the Capital One breach on your AWS account**
- **WrongSecrets – WrongSecrets is "Vulnerable by Design" to show how to not handle secrets in Docker, Kubernetes and in the cloud (AWS/GCP/Azure).**
- **AWSGoat – A Damn Vulnerable AWS Infrastructure**
- **AzureGoat – A Damn Vulnerable Azure Infrastructure**
- **IAM Vulnerable – Use Terraform to create your own vulnerable by design AWS IAM privilege escalation playground.**
- **Sadcloud – A tool for standing up (and tearing down!) purposefully insecure cloud infrastructure**

Mobile CTFs

- **Allsafe – Allsafe is an intentionally vulnerable application that contains various vulnerabilities.**
- **InsecureBankv2 – Vulnerable Android application for developers and security enthusiasts to learn about Android insecurities.**
- **Vulnerable Kext – A WIP "Vulnerable by Design" kext for iOS/macOS to play & learn *OS kernel exploitation.**
- **InjuredAndroid – A vulnerable Android application that shows simple examples of vulnerabilities in a ctf style.**
- **Damn Vulnerable Bank – Damn Vulnerable Bank is designed to be an intentionally vulnerable android application.**
- **InsecureShop – An Intentionally designed Vulnerable Android Application built in Kotlin.**
- **AndroGoat – AndroGoat is purposely developed open source vulnerable/insecure app using Kotlin.**
- **DIVA Android – Damn Insecure and vulnerable App for Android.**
- **OVAA – Oversecured Vulnerable Android App.**
- **Vuldroid – Android Application covering various static and dynamic vulnerabilities.**
- **Android Security Testing – hpAndro1337 Application made in Kotlin with multiple vulnerabilities and a CTF.**

Certifications: Your Learning Path

While hands-on experience and self-study are vital components of becoming a successful Cybersecurity Researcher & a Bug Bounty Hunter, certifications play a significant role in enhancing your skills and credibility as well as they help you get a better job in the future. Here are a few certifications that you might consider pursuing as a beginner:

- **CompTIA Security+**
- **Google Cybersecurity Certificate**

- Certified Ethical Hacker (CEH)
- GIAC Security Essentials Certification (GSEC)
- eLearnSecurity Certified Professional Penetration Tester (eCPPT)
- Offensive Security Certified Professional (OSCP)
- $(ISC)^2$ CC – Certified in Cybersecurity

PART 4

Selecting a Target, Testing, and Writing Effective Reports

In this phase, we'll delve into the critical process of selecting a target, getting started with testing, and ultimately crafting impactful bug reports. Let's dive right in!

Hey so Now the Final Phase I have in my mind is for People who have gone through all the good important stuff and now are testing.. so I'll like to give my advice about a few things and then will sum up this blog.

Selecting and Approaching a Target

Choosing the right target is a pivotal decision that sets the stage for your Bug Bounty Hunting endeavors. Honestly, your selection should be based on your mood, experience, and skill level. You can opt for a target with an expansive scope, encompassing multiple websites, subdomains, and mobile apps. Alternatively, you may prefer to focus on a single domain or app with intricate features for in-depth testing.

List of Bug Bounty Platforms:

To identify suitable programs, Bug Bounty Platforms like those below offer directories of programs.

- **Bugcrowd**
- **HackerOne**
- **Intigriti**
- **Yes We Hack**
- **Hacken Proof**
- **Open Bug Bounty**

Individual giants like Google, Facebook, and Apple run their own bug bounty programs like many other companies.

When approaching a target, careful reconnaissance is key. Conduct a thorough review of domain history, links, IPs, and Wayback Info to gain insights. Maintain detailed notes of your activities. Initiate your testing process by testing a specific functionality or workflow within the application. Begin by searching for low-hanging fruits and surface-level bugs, documenting their existence. Tools like Burp Suite or OWASP Zap are invaluable for observing workflows and requests.

Creating multiple accounts allows you to test user-to-user interactions. If not provided, request additional accounts, as it's a common practice. Engage with the app's flow, testing and probing for unusual behavior. While encountering anomalies doesn't always indicate a report-worthy bug, persistent exploration could unveil a security impact. Familiarize yourself with major security vulnerabilities and their corresponding methods. Web application flow comprehension is crucial; delve into API documentation for enhanced understanding. If you encounter challenges, make detailed notes for future reference.

These are great resources that will help you more about approaching & testing the targets

- **HOW TO APPROACH A NEW BOUNTY TARGET? 5 THINGS YOU MUST TEST FOR!**
- **Sticking With It: How To Choose a Target & Stay Motivated**

Reporting a Vulnerability

After investing considerable time in learning, practicing, and successfully identifying vulnerabilities, the report-writing phase emerges as a crucial step. Crafting an effective report demands precision and clarity to ensure your findings are properly communicated to the security team. A well-structured report expedites the review process and enhances collaboration. Consider the following guidelines:

1. **Thoroughness: Detail each step required to reproduce the bug. Eliminate ambiguity by providing comprehensive information.**
2. **Simplicity: Avoid unnecessary complexity. While technical details are important, excessive intricacy can hinder comprehension.**
3. **Impact Communication: Clearly convey the vulnerability's potential impact. If the impact exceeds initial assumptions, support your claims with evidence.**

4. **Courtesy: Remember, your report reaches a human audience. Be polite, patient, and respectful in your communication.**
5. **Media Elements: Use screenshots, videos, or other media to bolster your report. Visual aids can significantly enhance clarity.**

Here are resources that offer detailed insights into writing effective bug reports:

- **Bugcrowd's Guide to Successful Bug Submissions**
- **HackerOne's Quality Reports Guide**
- **Bug Bounty Guide: Writing Reports**
- **Intigriti's Guide: How to Write a Good Report**

Remember, your bug report reflects your professionalism and commitment. A well-crafted report enhances the efficiency of the triage process and maximizes your chances of a successful submission. Stay patient, be persistent, and continue refining your skills as you progress on your Bug Bounty Hunting journey. You're making a valuable contribution to cybersecurity, one report at a time.

Final Thoughts: A Bug Bounty Hunter's Perspective

With this final part, you've now gained insights into almost every critical aspect of bug bounty hunting. Your knowledge, skills, and dedication will undoubtedly propel you toward success in the exciting and ever-evolving world of Bug Bounty Hunting & Ethical Hacking.

As someone exploring security, keeping up with the latest can be tough. To those just starting, remember the power of learning on your own. You can achieve anything with the passion to take that first step. I'm still learning and want to share my knowledge to help others learn too.

Remember, you might not be perfect, but you're already better than most.

For both Bug Bounty Hunters and Cybersecurity Researchers, passion is the key. I hope this article has motivated you to start something positive. Thank you for reading. This is what I can share for now, but I promise to update this article with more helpful insights for more readers as much as I can.