Full Advanced Bug Bounty Guide ( )

: Beginner Pro Level Bug Hunter Step-by-Step
, , ,

. Bug Bounty

Bug Bounty ?

Bug Bounty (vulnerability)

?

-
- Responsible Disclosure
- Ethical Hacking
-

- HackerOne
- BugCrowd
- Intigriti
- YesWeHack

.

OS

- **Linux (Ubuntu/Kali/Parrot)**:
- **WSL (Windows Subsystem for Linux)**: Windows

Terminal Tools & Fonts

- `zsh`, `oh-my-zsh`, `powerlevel10k`
- Nerd Fonts: Hack Nerd Font

( )

```
sudo apt update && sudo apt install -y git curl wget python3-pip
# Golang (nuclei, httpx, subfinder )
wget https://go.dev/dl/go1.21.5.linux-amd64.tar.gz
sudo tar -C /usr/local -xzf go1.21.5.linux-amd64.tar.gz
```

```
export PATH=$PATH:/usr/local/go/bin

#
GO111MODULE=on go install -v github.com/projectdiscovery/subfinder/v2/cmd/subfinder@latest
GO111MODULE=on go install -v github.com/projectdiscovery/httpx/cmd/httpx@latest
GO111MODULE=on go install -v github.com/projectdiscovery/nuclei/v2/cmd/nuclei@latest
GO111MODULE=on go install -v github.com/tomnomnom/waybackurls@latest
GO111MODULE=on go install -v github.com/lc/gau/v2/cmd/gau@latest
GO111MODULE=on go install -v github.com/hakluke/hakrawler@latest
pip3 install jsfinder
```

. Step-by-Step Bug Bounty

```
gantt
    title Bug Bounty Timeline
    section Recon
    Subdomain Enumeration    :done,    des1, 2024-01-01, 2d
    Asset Discovery          :done,    des2, after des1, 1d
    section Scanning
    Vulnerability Scanning   :active,  des3, after des2, 2d
    section Exploitation
    Manual Testing           :         des4, after des3, 2d
    section Reporting
    Report Writing           :         des5, after des4, 1d
```

1. **Reconnaissance**:
2. **Enumeration**: , ,
3. **Vulnerability Scanning**:
4. **Exploitation**:
5. **Reporting**:

. Advanced Subdomain Enumeration

Passive Method

- `subfinder -d example.com -o subs.txt`
- CRT.sh, SecurityTrails, VirusTotal

Active Method

- `amass enum -d example.com`
- Permutation: `dnsgen`, `altdns`

ASN Enumeration

```
python3 asnmap.py -a <ASN> -o asn_domains.txt
```

Custom Bash Script (Automation)

```
#!/bin/bash
# subenum.sh
# Usage: ./subenum.sh example.com
DOMAIN=$1
subfinder -d $DOMAIN -o subs.txt
amass enum -d $DOMAIN -o amass.txt
dnsgen subs.txt -o perm.txt
cat subs.txt amass.txt perm.txt | sort -u > all_subs.txt
```

. Reconnaissance ( )

URLs, JS, Secrets, API, Params

- Wayback: `waybackurls example.com > urls.txt`
- GAU: `gau example.com > gau.txt`
- JS Finder: `jsfinder -u https://example.com -o js.txt`
- Hakrawler: `hakrawler -url https://example.com -depth 2 > hak.txt`

Secrets/Keys Extraction

```
grep -Eri 'api[_-]?key|secret|token' js.txt
```

. Vulnerability Assessment (OWASP Based)

XSS

- : `dalfox`, `kXSS`
- Payload: `<script>alert(1)</script>`
- Example: `dalfox url https://example.com/vuln?param=1`

SQLi

- : `sqlmap`
- Payload: `' OR 1=1--`
- Example: `sqlmap -u "https://example.com/item?id=1" --batch`

IDOR

- : ID

SSRF

- : `ssrfmap`, Burp Collaborator
- Payload: `http://burpcollaborator.net`

RCE

- : `nuclei`, `commix`
- Payload: `;id`

CSRF

- Burp Suite

. Exploitation Techniques

Manual Exploitation

- Burp Suite
- Custom Payload

Automated Tools

- `nuclei -l urls.txt -t cves/`
- `dalfox file urls.txt`

. Report Writing

Report Format (HackerOne/BugCrowd)

- **Title**: Vulnerability Name
- **Summary**:
- **Steps to Reproduce**:
- **Impact**:
- **PoC**: , Burp log

Example:

```
## Title: Stored XSS in Profile Section
## Summary:
Profile update     XSS
## Steps to Reproduce:
1. Login
2. Profile
3. Name     <script>alert(1)</script>
```

```
4. Save
## Impact:
Attacker arbitrary JS execute
## PoC:
[    /   ]
```

. Automation Scripts

Recon Script (bash)

```bash
#!/bin/bash
# recon.sh
DOMAIN=$1
subfinder -d $DOMAIN -o subs.txt
amass enum -d $DOMAIN -o amass.txt
cat subs.txt amass.txt | sort -u > all_subs.txt
for sub in $(cat all_subs.txt); do
  httpx -u $sub -o live.txt
  waybackurls $sub >> urls.txt
  gau $sub >> gau.txt
  nuclei -u $sub -o nuclei.txt
  hakrawler -url https://$sub -depth 2 >> hak.txt
  jsfinder -u https://$sub -o js_$sub.txt
  #
  echo "[+] Done: $sub"
done
```

Folder Structure

```
recon/
     subs.txt
     amass.txt
     all_subs.txt
     live.txt
     urls.txt
     gau.txt
     nuclei.txt
     hak.txt
     js_*.txt
```

Bonus Content

Wordlist Optimization

- Custom wordlist: `assetfinder`, `commonspeak2-wordlists`
- Fuzzing: `ffuf`, `wfuzz`

Passive vs Active Recon

- Passive: 3rd party sources, no direct interaction
- Active: Direct probing, brute force

Scope

- Program Policy
- `scope` keyword grep

Github Dork Automation

```
gitdorks_go -q 'api_key' -o dorks.txt
```

Info Disclosure Enumeration

- `.git`, `.env`, JS Analysis
- `git ls-remote https://example.com/.git`
- `curl https://example.com/.env`

. Best Resources

- HackerOne Public Reports
- BugCrowd University
- YouTube: LiveOverflow, NahamSec, Stök
- Blogs: ProjectDiscovery Blog, PortSwigger Web Security
- Tool Docs: Nuclei, Subfinder

Special Tips: - - - Automation Deep Enumeration Custom Script - - Practice is the key! Happy Hunting!