CTF Toolkit

Tools and tricks for capture-the-flag competitions across web, crypto, forensics, reversing, pwn, and OSINT.

Setup

Standard setup: Kali or Parrot in a VM, plus a notes app (Obsidian, CherryTree). Keep wordlists in a known location:

```
/usr/share/wordlists/
rockyou.txt # passwords
dirb/common.txt # web fuzzing
seclists/ # category-organized lists (clone from GitHub)
```

Web

`burpsuite`	Intercept proxy — set browser to 127.0.0.1:8080
`ffuf -w list -u url/FUZZ`	Directory & param fuzzing
`gobuster dir -u url -w list`	Alt directory brute-forcer
`sqlmap -u url --batch --dbs`	Auto SQL injection
`nikto -h url`	Quick web vuln scan
`wpscan --url url`	WordPress audit
`jwt.io / jwt-tool`	Inspect / forge JWTs
`curl -X POST -d ...`	Manual request crafting
`view-source: + DevTools`	Always check HTML/JS for hidden clues

Crypto

`CyberChef (gchq.github.io)`	Swiss army knife for encoding/cipher
`xor / xortool`	XOR cipher tools
`hashid hash`	Identify hash algorithm
`hashcat -m mode -a 0 hash list`	GPU cracking
`john --wordlist=rockyou.txt`	CPU cracking
`openssl enc -d -aes-256-cbc ...`	Symmetric decrypt
`rsatool / RsaCtfTool`	RSA attack toolkit
`factordb.com`	Lookup factorization of small N
`SageMath`	Mathematical CTF problems

Forensics

`file artifact`	First step always: identify file
`strings -n 8 file`	Find printable ASCII
`exiftool image.jpg`	EXIF metadata
`binwalk -e file`	Extract embedded files
`foremost / scalpel`	Carve files from raw images
`zsteg / stegsolve`	PNG/BMP steganography
`steghide extract -sf file`	Extract from JPG/WAV

`volatility -f mem.raw imageinfo`	Memory forensics
`autopsy`	Disk forensics GUI
`pdfid / pdf-parser`	Inspect PDF objects
`oletools (oleid, olevba)`	Office macro analysis

Reverse Engineering

`file binary`	Architecture & format
`strings binary`	Look for cleartext flags
`ltrace / strace`	Trace lib calls / syscalls
`objdump -d binary`	Quick disassembly
`ghidra`	Free decompiler (NSA)
`cutter / iaito`	GUIs for radare2
`radare2 binary`	CLI RE framework
`IDA Free / Pro`	Industry-standard disassembler
`Hopper`	macOS-friendly disassembler
`dnSpy / ILSpy`	.NET decompilers
`jadx`	Android APK decompiler
`apktool d app.apk`	Decompile APK to smali

Pwn / Binary Exploitation

`checksec binary`	Show NX, ASLR, RELRO, canary
`pwntools (Python)`	Exploit dev framework
`gdb-peda / pwndbg / gef`	Enhanced GDB for pwn
`one_gadget libc.so.6`	Find one-shot RCE in libc
`ROPgadget --binary file`	Find ROP gadgets
`ropper / ropium`	ROP chain builder
`libc-database`	Identify libc by leaked offsets
`cyclic 200`	pwntools deBruijn pattern

OSINT & Recon

`Google dorks`	site:, intext:, filetype:, inurl:
`shodan.io`	Internet-facing devices
`censys.io`	Alternative to Shodan
`whois domain`	Registration info
`crt.sh`	Certificate transparency log search
`archive.org wayback`	Old versions of pages
`exiftool image`	GPS, camera, software metadata
`reverse image search`	TinEye, Yandex, Google Images
`osintframework.com`	Curated OSINT links

Quick Wins to Always Try

1. **View source** + check JavaScript for endpoints, comments, base64 strings.
2. **Robots.txt, sitemap.xml, .git/, .env** — exposed metadata files.
3. **Default credentials** — admin/admin, root/root, user/password.

4. **SQL injection** — single quote in any input field.
5. **LFI** — try ?file=../../../etc/passwd patterns.
6. **Cookies** — base64 / JWT decode every cookie.
7. **Headers** — check for X-Powered-By, Server, custom debug headers.
8. **Hidden parameters** — ffuf with seclists/Discovery/Web-Content/burp-parameter-names.txt.

Practice Platforms

• **picoCTF** — beginner-friendly, free, year-round.

• **HackTheBox** — realistic boxes, community.

• **TryHackMe** — guided rooms, structured paths.

• **OverTheWire** — wargames (start with Bandit).

• **CTFtime.org** — calendar of upcoming events + writeups.